

# 微服务架构设计与实践

## 监控篇



孙玄@58集团

# 关于我

- ✓ 58集团技术  全球软件开发大会 主席
- ✓ 58集团高级  中国软件开发者大会 架构师
- ✓ 转转架构师  程序员 负责人
- ✓ 百度高级工程师
- ✓ 毕业于浙江大学
- ✓ 代表公司多次对外分享
- ✓ 企业内训&公开课



# 关于我

---

## 企业内训

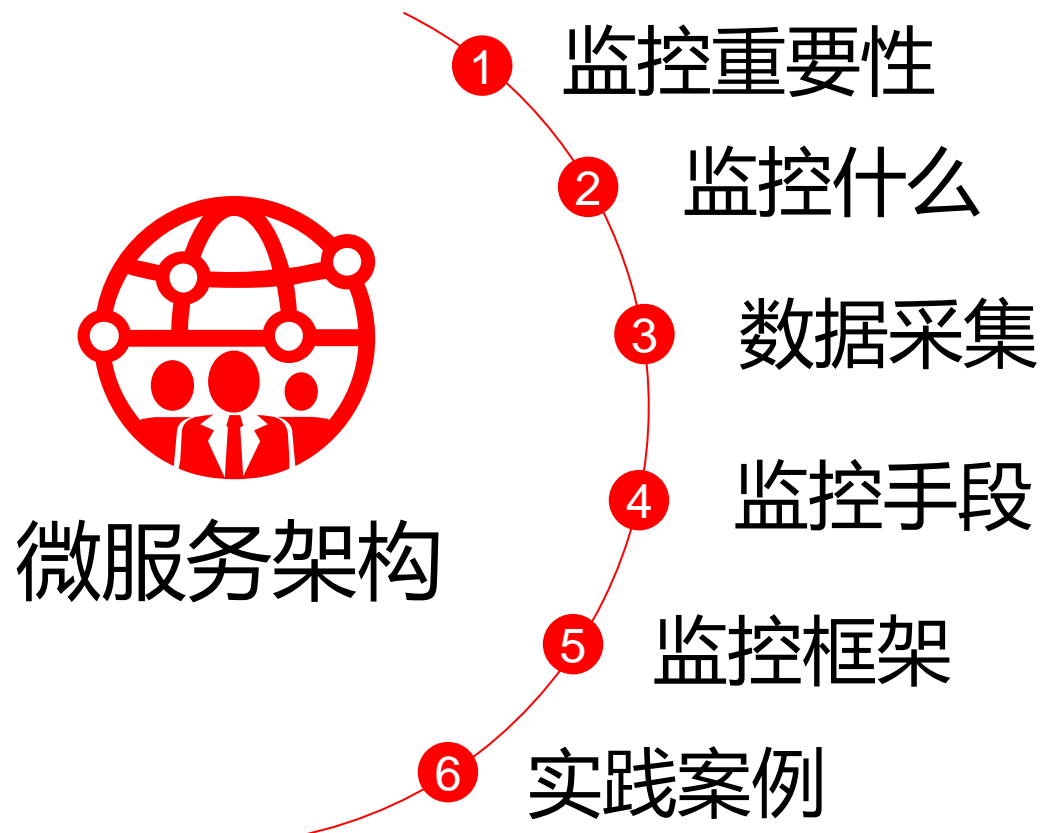
- ✓ 华为
- ✓ 中航信
- ✓ 平安
- ✓ 银联
- ✓ 华泰证券
- ✓ 思科

- ✓ 云南电力
- ✓ 深信服
- ✓ 新华社
- ✓ 民生银行
- ✓ 招商银行
- ✓ .....

## 公开课

- ✓ 北京
- ✓ 上海
- ✓ 深圳
- ✓ 广州
- ✓ 成都
- ✓ .....

# 分享要点



# 微服务监控重要性

---

**为什么需要监控**



# 监控什么

---

## 监控什么



# 监控数据采集

---

- **用户行为采集**

- **请求行为**

- 服务端日志收集
    - 日志打印

- **页面行为**

- 客户端埋点收集
      - JavaScript脚本埋点

# 监控数据采集

- **日志规范**

- 参考《日志规范@孙玄.pptx》

- **日志格式**

- 便于查看、分析问题
  - Key-Value 格式
  - 空格分割

```
20:34:14,087 INFO QRCodeScanController:252 - uid=28193990512390 logid=7279929494527705543 uri=/ershouchu/qr/getuserbyuid desc=getuserbyuid_start
20:34:14,090 INFO IMContentResult:27 - uid=28193990512390 logid=7279929494527705543 uri=/ershouchu/qr/getuserbyuid len=55 content={"respCode":0,"respData":{"
userStatus":0,"errCode":-1}}
20:34:15,735 INFO GetPostController:252 - uid=22790999711750 logid=4075759097141954615 uri=/ershouchu/getpostdetailinfo desc=parse_url_success infoid=2231952
8677407
```



# 微服务监控手段

## 监控手段



# 进程监控

- 微服务进程管理
  - 重启[进程挂掉]
  - 是否活着[alive]
  - CPP/Java
  - 工具
    - Linux daemontools
    - supervise

```
[im_work@im17 ~]$ ps -axu | grep router | grep -v "grep"
Warning: bad syntax, perhaps a bogus '-'? See /usr/share/doc/procps-3.2.7/FAQ
root    12733  0.0  0.0  3680   336 ?        S    Mar31   0:00 ./supervise router ./
root    27384 89.4  1.6 1665148 527608 ?      Ssl  Mar31 1922:48 ./router --conf=../conf/rs.conf
[im_work@im17 ~]$
```

# 进程监控

- Supervise开启

- start.sh

- ./supervise\_router ./ & **supervise\_router**

- run

- #!/bin/bash

- ./router --conf=../conf/rs.conf 1>/dev/null 2>&1

```
[im_work@im17 ~]$  
[im_work@im17 ~]$ ps -axu | grep router | grep -v "grep"  
Warning: bad syntax, perhaps a bogus '-'? See /usr/share/doc/procps-3.2.7/FAQ  
root    12733  0.0  0.0   3680   336 ?        S    Mar31   0:00 ./supervise_router ./  
root    27384 89.4  1.6 1665148 527608 ?        Ssl  Mar31 1922:48 ./router --conf=../conf/rs.conf  
[im_work@im17 ~]$
```

# 语义监控

- 微服务进程管理
  - 是否工作正常
  - 模拟发送请求
  - 预期结果对比

## 【服务器监控】接入层 ★

### 接口报警

发给 sunxuan

发件人: 接口报警

收件人:

时间: 2016年6月25日 (周六) 23:55

大小: 2 KB

10. / InfoForInfo

### 错误信息:

timeout [时间: 06-25 23:53:02:929]  
timeout [时间: 06-25 23:52:02:960]  
timeout [时间: 06-25 23:00:02:785]  
timeout [时间: 06-25 05:32:00:060]  
timeout [时间: 06-24 16:08:57:685]

```
ERROR 2015-03-31 22:59:12:101523 rs_monitor 3903:47117189707264 Init(src/monitor.cpp:46) - desc=conn_err ip=10.5.17.76 port=20023
ERROR 2015-03-31 22:59:51:058030 rs_monitor 14643:47774130169344 Init(src/monitor.cpp:46) - desc=conn_err ip=10.5.17.76 port=20023
ERROR 2015-03-31 23:01:09:677200 rs_monitor 1144:47036271257088 Init(src/monitor.cpp:46) - desc=conn_err ip=10.5.17.76 port=20023
ERROR 2015-03-31 23:01:48:523241 rs_monitor 8653:47052693046784 Init(src/monitor.cpp:46) - desc=conn_err ip=10.5.17.76 port=20023
ERROR 2015-03-31 23:02:55:317289 rs_monitor 26474:47549509731840 Init(src/monitor.cpp:46) - desc=conn_err ip=10.5.17.76 port=20023
ERROR 2015-03-31 23:03:35:390158 rs_monitor 2777:47325623535104 Init(src/monitor.cpp:46) - desc=conn_err ip=10.5.17.76 port=20023
ERROR 2015-03-31 23:04:43:884430 rs_monitor 20533:47465426945536 Init(src/monitor.cpp:46) - desc=conn_err ip=10.5.17.76 port=20023
ERROR 2015-03-31 23:05:22:824555 rs_monitor 29235:47329628468736 Init(src/monitor.cpp:46) - desc=conn_err ip=10.5.17.76 port=20023
ERROR 2015-03-31 23:17:50:549182 rs_monitor 21371:47561328674304 Init(src/monitor.cpp:46) - desc=conn_err ip=10.5.17.76 port=20023
ERROR 2015-03-31 23:27:53:471178 rs_monitor 7882:47021922223616 Init(src/monitor.cpp:46) - desc=conn_err ip=10.5.17.76 port=20023
```

- 微服务进程管理
  - 微服务是否正常
  - 阈值设定[1分钟高峰期错误量]
  - 实时统计
    - flume/kafka/spark
    - Redis

【服务器监控】

错误日志

★

接口报警

发给 sunxuan

发件人: 接口报警

收件人:

时间: 2016年6月26日 (周日) 19:01

大小: 2 KB

10\_

zzinfo

错误信息 :

阈值 5 实际 6 [时间: 06-26 18:59:53:110]

阈值 5 实际 8 [时间: 06-26 18:55:52:433]

```
ERROR 2015-03-31 22:59:12:101523 rs_monitor 3903:47117189707264 Init(src/monitor.cpp:46) - desc=conn_err ip=10.5.17.76 port=20023
ERROR 2015-03-31 22:59:51:058030 rs_monitor 14643:47774130169344 Init(src/monitor.cpp:46) - desc=conn_err ip=10.5.17.76 port=20023
ERROR 2015-03-31 23:01:09:677200 rs_monitor 1144:47036271257088 Init(src/monitor.cpp:46) - desc=conn_err ip=10.5.17.76 port=20023
ERROR 2015-03-31 23:01:48:523241 rs_monitor 8653:47052693046784 Init(src/monitor.cpp:46) - desc=conn_err ip=10.5.17.76 port=20023
ERROR 2015-03-31 23:02:55:317289 rs_monitor 26474:47549509731840 Init(src/monitor.cpp:46) - desc=conn_err ip=10.5.17.76 port=20023
ERROR 2015-03-31 23:03:35:390158 rs_monitor 2777:47325623535104 Init(src/monitor.cpp:46) - desc=conn_err ip=10.5.17.76 port=20023
ERROR 2015-03-31 23:04:43:884430 rs_monitor 20533:47465426945536 Init(src/monitor.cpp:46) - desc=conn_err ip=10.5.17.76 port=20023
ERROR 2015-03-31 23:05:22:824555 rs_monitor 29235:47329628468736 Init(src/monitor.cpp:46) - desc=conn_err ip=10.5.17.76 port=20023
ERROR 2015-03-31 23:17:50:549182 rs_monitor 21371:47561328674304 Init(src/monitor.cpp:46) - desc=conn_err ip=10.5.17.76 port=20023
ERROR 2015-03-31 23:27:53:471178 rs_monitor 7882:47021922223616 Init(src/monitor.cpp:46) - desc=conn_err ip=10.5.17.76 port=20023
```

# coredump监控

- 微服务进程管理
  - 及时发现微服务进程coredump
  - 任何语言

```
core.10140.20141109.161120 core.22066.20141014.205101 core.30071.20141106.163921 pusher.20140929.134646 pusher.20141022.210614 pusher.20141108.152426
core.10278.20141022.131301 core.22122.20141021.194503 core.30145.20141017.173951 pusher.20140930.074720 pusher.20141023.101544 pusher.20141109.161120
core.10838.20141020.101515 core.22201.20141113.142428 core.31001.20141023.160532 pusher.20140930.085740 pusher.20141023.111259 pusher.20141110.101540
core.1133.20141018.100555 core.22342.20141103.160810 core.31051.20141027.123650 pusher.20140930.094506 pusher.20141023.133401 pusher.20141110.142436
core.11646.20141022.210614 core.22764.20141027.165030 core.31240.20141117.170242 pusher.20140930.135822 pusher.20141023.150526 pusher.20141111.190637
core.11953.20141112.135041 core.23874.20140930.094506 core.31793.20141120.091714 pusher.20141001.175001 pusher.20141023.160532 pusher.20141112.084602
core.12313.20141020.185613 core.24030.20141105.182407 core.3190.20141114.172505 pusher.20141001.211455 pusher.20141024.141843 pusher.20141112.091041
```

# 机器资源监控

- 机器资源管理
  - 及时发现机器资源不足
    - 磁盘
    - 内存
    - CPU
    - 网络
    - ...

## 【服务器监控】硬盘 ★

### 接口报警

发给

发件人: 接口报警

收件人:

时间: 2016年6月26日 (周日) 20:34

大小: 2 KB

**IP: 10.**

### 错误信息 :

Used:85Percent Limit:85 [时间: 06-26 20:34:40:193]

Used:85Percent Limit:85 [时间: 06-26 20:33:37:108]

Used:85Percent Limit:85 [时间: 06-26 20:32:35:191]

Used:85Percent Limit:85 [时间: 06-26 20:31:40:768]

Used:85Percent Limit:85 [时间: 06-26 14:44:37:271]

# 机器资源监控

- 如何监控

- 磁盘

- df -h

```
[work(host/...@... bin]$ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda2        62G   7.5G   51G   13% /
tmpfs            63G    0    63G    0% /dev/shm
/dev/sda1        976M   57M   869M    7% /boot
/dev/sda4        2.6T   1.4T   1.1T   56% /opt
```

- 内存

- free -g

```
[work( bin]$ free -g
              total        used        free      shared    buffers     cached
Mem:           125          119           6           0           0          24
-/+ buffers/cache:           95           30
```

- CPU

- vmstat

```
[work( bin]$ vmstat 1
procs -----memory-----swap-----io-----system-- -----cpu-----
r  b   swpd   free   buff  cache   si   so    bi   bo    in   cs  us  sy  id  wa  st
4  0   2890052 5764188  63508 26308632    0    0    14   884    0    0   9   4  87   0   0
17 0   2890052 5700552  63536 26367584    0    0     0  6920 137996 185115 17   5  77   0   0
8  0   2890052 5642648  63572 26424876    0    0   128  8888 138440 185698 18   5  77   0   0
8  0   2890052 5569036  63584 26498476    0    0     0  5204 135008 184149 16   5  79   0   0
7  0   2890052 5511012  63608 26555776    0    0     0 10516 135384 184683 17   5  79   0   0
10 0   2890052 5445168  63624 26617952    0    0     0  7140 139666 187404 16   5  79   0   0
```

- .....

## Zabbix



# 数据波动监控

- 数据监控

核心数据变动情况 ★	
发件人	
发给	
===== (增幅) 超过10%的数据 =====	
有用户)	昨日: 4105 今日: 4725 增幅: 15.10%
	昨日: 1925 今日: 2212 增幅: 14.91%
===== (降幅) 超过10%的数据 =====	
	昨日: 114083 今日: 48937 降幅: -57.10%
	昨日: 0 今日: 0 降幅: -10.73%

# 数据波动监控

- 数据监控实施

日志采集-》日志缓存-》日志计算-》结果存储-》结果查询-》数据结果判定-》报警



# 微服务监控框架

- 监控框架设计

- 功能

- 统一调度监控

- 动态配置报警人

- 阈值设置

- 发送邮件、短信

- 业务接入

- 只需要编写发送“数据”部分

转转监控

配置项 <<

监控列表 ×

监控名称  查询

id	监控名称	命令路径	监控类型	thresho	状态	创建时间	操作
1	转转错误日志	aw_logs_moni	IP监控	2	正常	2016-05-26 18:50:32	修改 删除 暂停
2	硬盘	sk/new_disk_moni	IP监控	2	正常	2016-05-27 10:01:01	修改 删除 暂停
3	接入层	ry/new_entry.sh	URI监控	2	正常	2016-05-27 14:35:49	修改 删除 暂停
5	登录	onitor/entr/new_entry_sp	URI监控	2	正常	2016-05-27 14:47:00	修改 删除 暂停
6	搜索	new_es_moni	URI监控	2	正常	2016-05-27 14:47:00	修改 删除 暂停
7	运营接入层	new_e	URI监控	2	正常	2016-05-27 14:47:00	修改 删除 暂停
8	内存	_mem_m	IP监控	2	正常	2016-05-27 14:47:00	修改 删除 暂停
11	push监控	sh/new_logs_mon	IP监控	2	正常	2016-06-23 19:14:32	修改 删除 暂停

20 Page 1 of 1

# 微服务监控框架

---

- **监控框架设计**

- 支持功能

- 进程

- 语义

- 错误日志

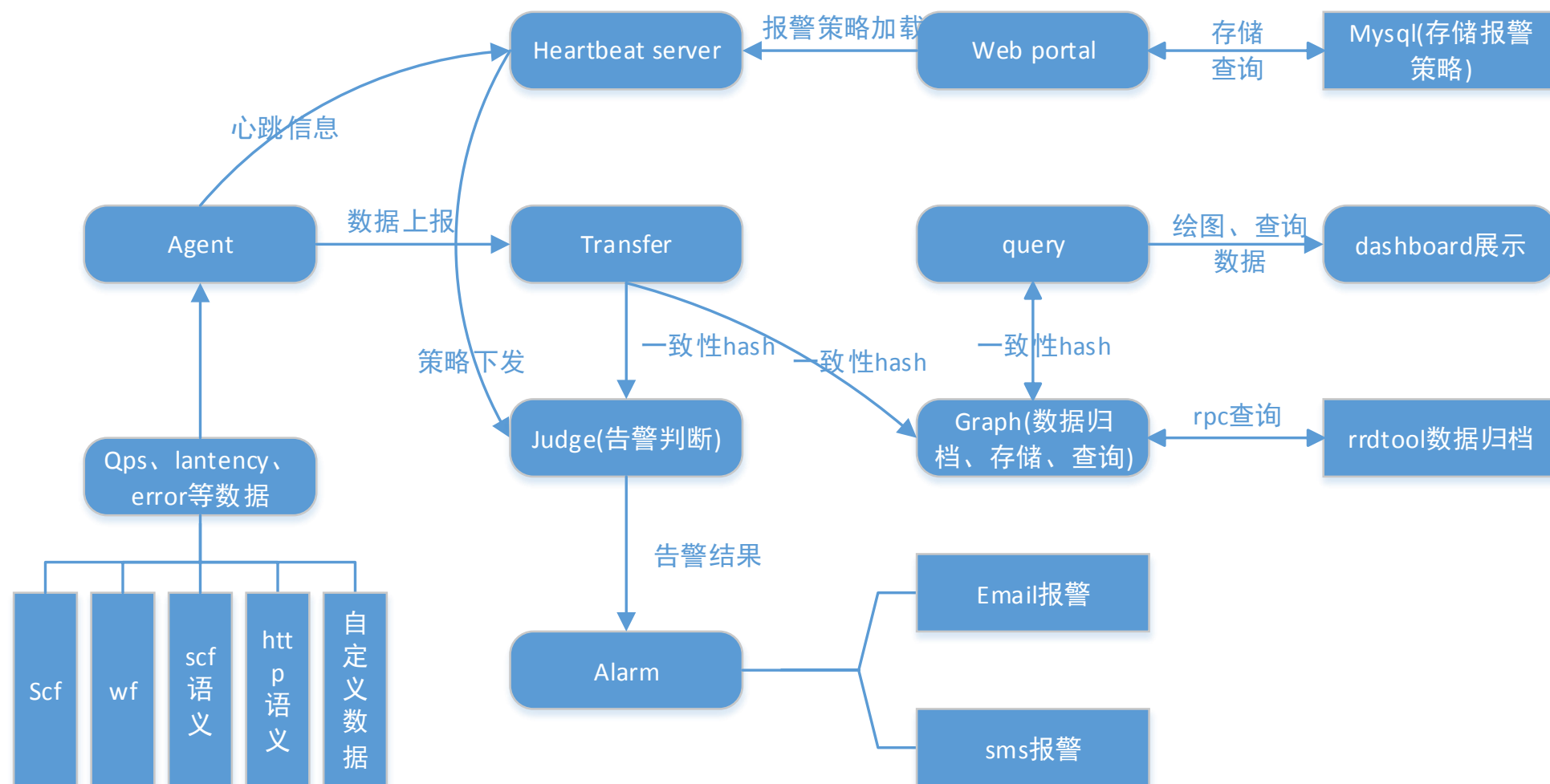
- QPS

- 平均延迟

- 自定义监控

# 微服务监控框架

## • 总体架构



# 监控痛点

---

- **短信通道**

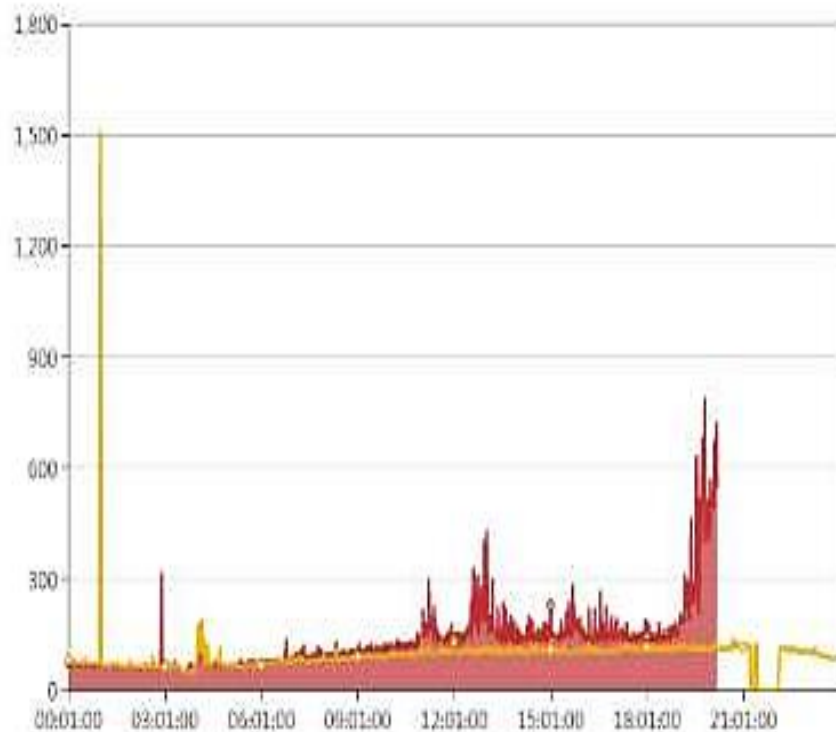
- 延迟、丢失
- 内容size限制，报警内容受限
  - 短信、邮件
  - 微信

- **可视化监控**

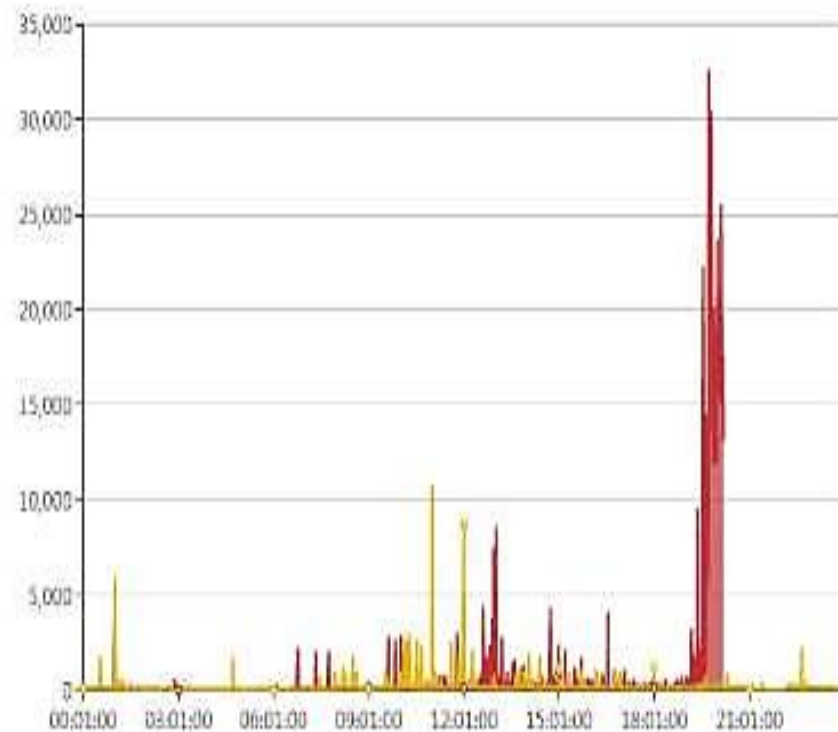
- 关键指标投影屏幕
- 更早、更及时发现问题
- 掌握主动权

# 可视化监控

请求平均耗时



请求异常条数



# 可视化监控

日志监控系统

退出 您好: 学

首页

日志监控

语义监控

脚本监控

触发项

业务监控

前端错误监控

我的收藏

关于系统

监控项列表

监控描述

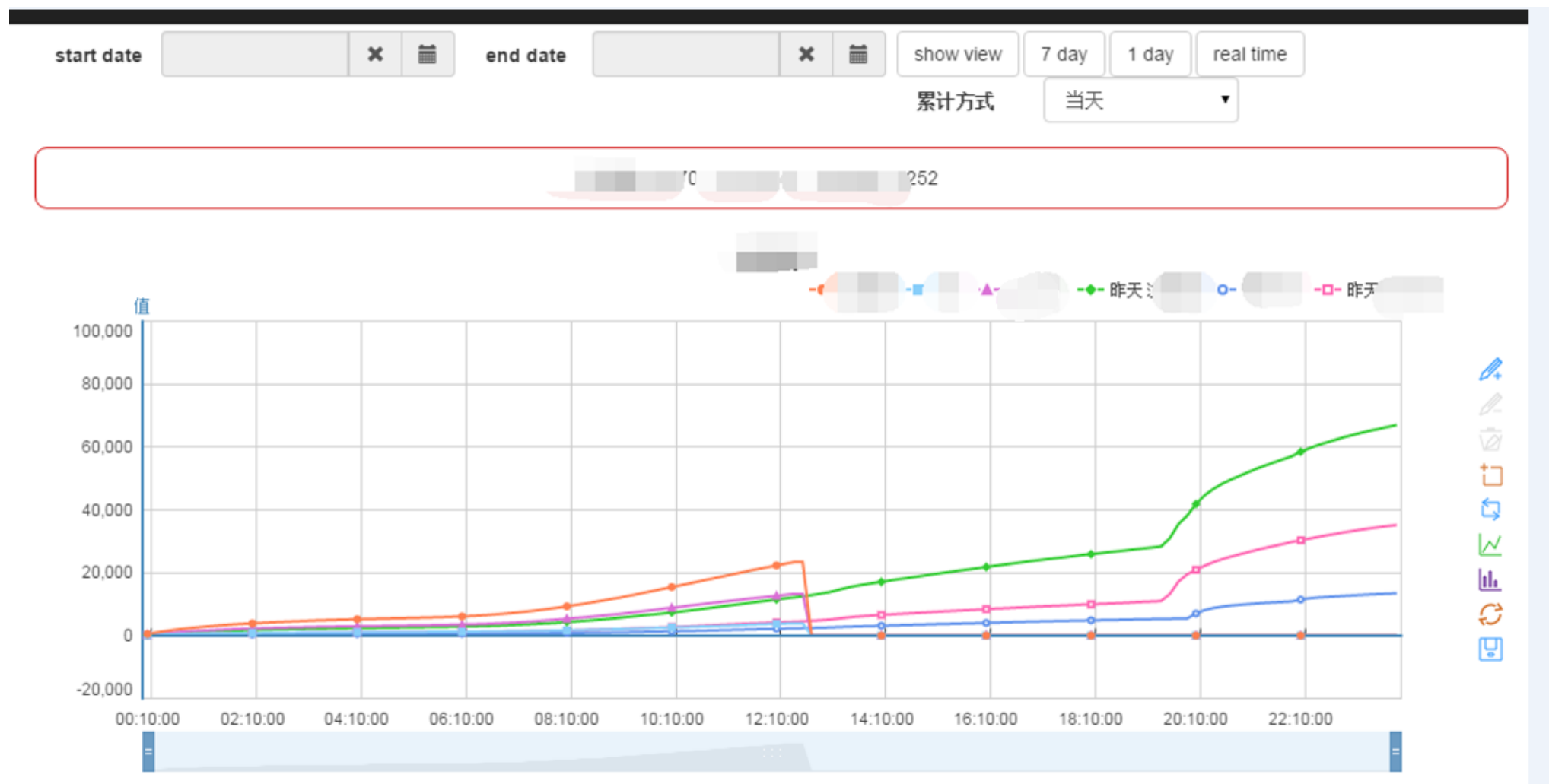
集群

查询

集群名称	日志行	监控项名称	监控字段	统计算法	创建人	操作
logic	log	注册数		去重	li	<div>图形</div> <div>收藏</div>
logic	log	登录数		去重	,	<div>图形</div> <div>收藏</div>
nfologic	log	发帖用户数		去重	yi	<div>图形</div> <div>收藏</div>
logic		创建订单数	d	计数		<div>图形</div> <div>收藏</div>
logic		创建订单用户数	d	去重		<div>图形</div> <div>收藏</div>
logic		支付成功数	d	计数		<div>图形</div> <div>收藏</div>
logic		确认收货数	ld	计数		<div>图形</div> <div>收藏</div>
logic		使用红包金额(分)	mount	求和		<div>图形</div> <div>收藏</div>



# 可视化监控



# 微服务架构监控实践

---

- 进程监控
- 语义监控
- 错误日志监控
- coredump监控
- 机器资源监控
- 数据波动

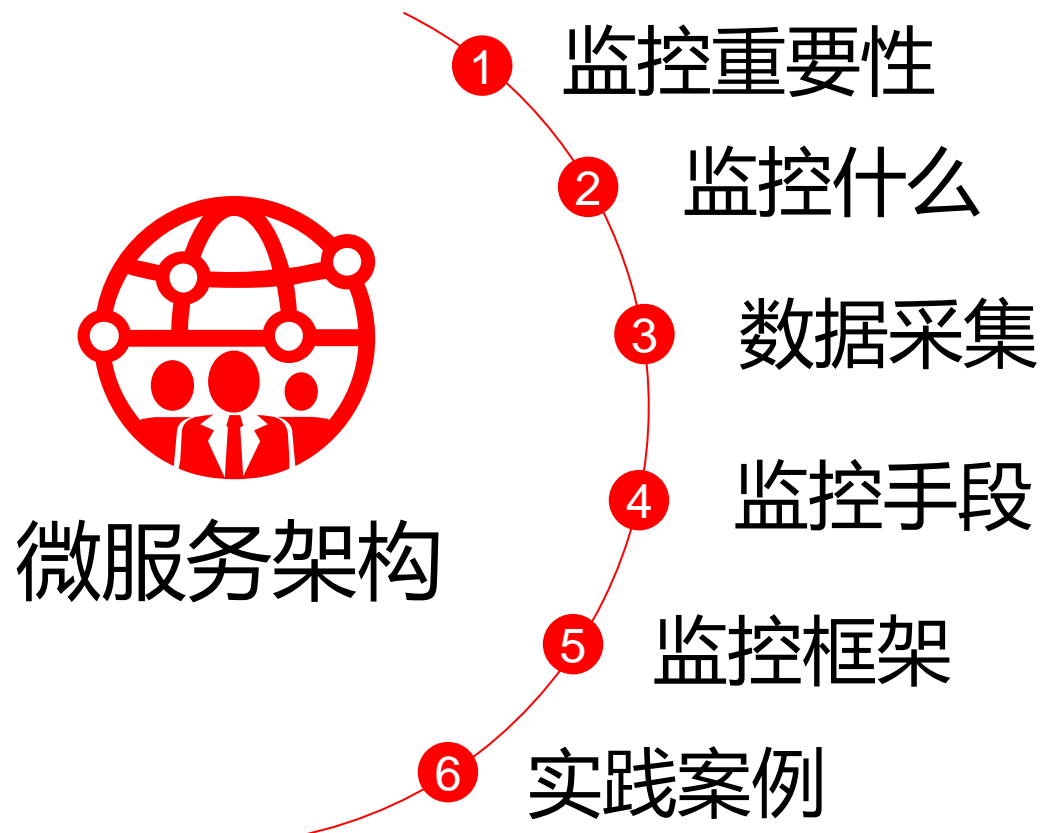
# 我们实践案例

---

- 实践案例二

- MongoDB监控
- 《MongoDB实战\_MongoDB集群监控@孙玄.pptx》

# 要点回顾



欢迎关注本人公众号 “架构之美”



# Thanks!

让生活更简单

