**原创；微信公众号：千里行走；**

**受限图片大小限制，有些图片不是很清晰，可以到微信公众号查看；**

前置阅读：[cdn-1：cdn使用的基础架构集成方案与成本控制方案](http://mp.weixin.qq.com/s?__biz=Mzg4MDEzMDM4MA==&mid=2247483792&idx=1&sn=c84b28a83b0d8926ed2952150cc5c54e&chksm=cf78a1b3f80f28a5d79330ccab8acfdaa003975dc2b27541c6f9e3651af7d4387855845653be&scene=21#wechat_redirect)

我们对服务端上传文件到oss做了收敛，将oss的写操作(目前只有文件上传)封装到基础服务中，那么就涉及到了对oss写操作的监控问题。

**目录：**

**(1).文件上传服务监控之线上效果图**

1.监控总览

2.oss bucket上传次数统计

3.oss bucket上传文件大小统计

4.oss bucket上传文件时间统计

5.grafana的dashboard-json

**(2).文件上传服务监控之主要代码**

1.自研框架注解方式集成

2.prometheus组件主要代码

3.主体逻辑

**(3).总结**

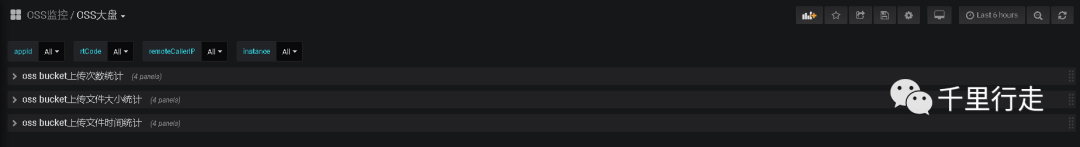
**(1).文件上传服务监控之线上效果图**

主要监控维度：

文件上传次数统计情况，文件上传大小统计情况，文件上传耗时统计情况。

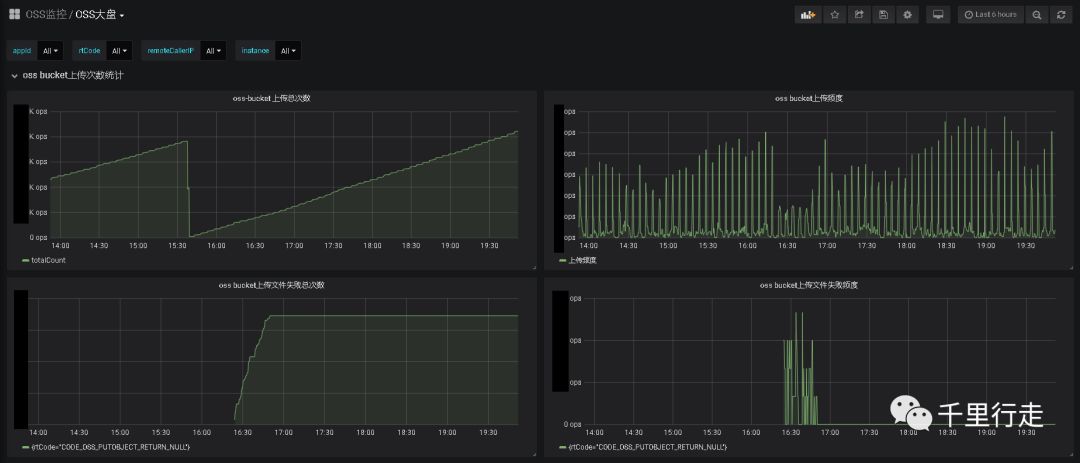
每个监控维度都有细分指标，如p90(fileSize), p99(timeCost)，详细见后图。

**1.监控总览**



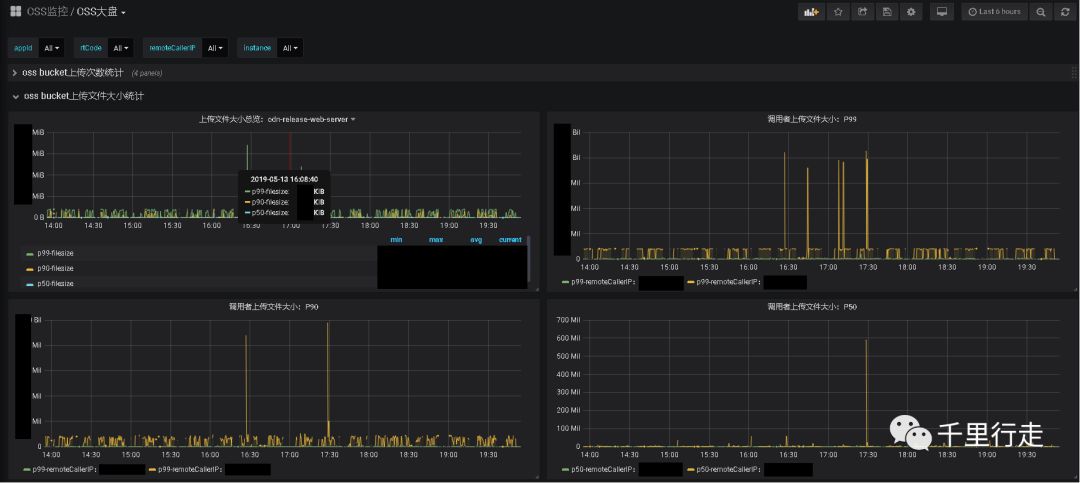
**2.oss bucket上传次数统计**

具体数据脱敏。



**3.oss bucket上传文件大小统计**

具体数据脱敏。



**4.oss bucket上传文件时间统计**

具体数据脱敏。



**5.grafana的dashboard-json**

配置参考：

https://github.com/hepyu/grafana-action/blob/master/dashboarrd/dashboard-oss/OSS%E5%A4%A7%E7%9B%98-1557749311378.json

**(2).文件上传服务监控之主要代码**

**1.自研框架注解方式集成**

cdn和oss的写操作都集成到了自研框架中，目前只对oss做了prometheus监控。

标注1：封装阿里云OSS的ossClient操作，方便引入prometheus。

标注2：封装的ossClient的方法。

标注3：prometheus组件。



**2.prometheus组件主要代码**

统计putObject次数的prometheus-Counter：



统计putObject的fileSize和timeCost的prometheus-Histogram：



**3.主体逻辑**



**(3).总结**

1.后续需要在每个指标中/新增指标加入返回的cdn的域名。

可以顺带监控不同CDN域名的负载情况。

原因：

不论是pc还是手机上的浏览器，同一时间访问同一个域名的请求个数是有限的，通过多个域名可以提高用户体验。

2.持续完善oss/cdn的接口封装和对应的监控。

3.最终要进行容器化。

