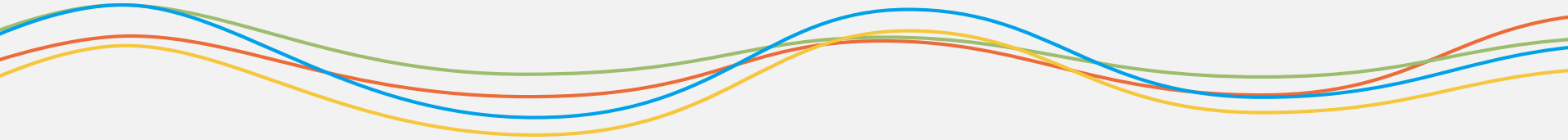
APMCon

微服务架构的应用性能监控

廖雄杰@听云



目录

APMCon

Why micro services ?

微服务架构下的应用性能监控

听云微服务化及监控

复杂调用链性能监控及追踪

Why Micro Services?

APMCon

从单体到微服务

- 单体架构适用于中小型产品前期快速迭代验证
- 服务及数据体量的爆炸性增长
- 分布式环境下单体架构的问题
- 中央集权 vs 领域自治

Why Micro Services?

APMCon

微服务架构的优势

- 低耦内聚
- 轻量，快速迭代，CI/CD更简单
- 更可靠
- 容易监控，发现/定位问题更快捷？

Why Micro Services?

APMCon

微服务架构的几种打开姿势

- 按业务垂直拆分
- 按模块水平拆分
- 垂直+水平拆分

Why micro services ?

微服务架构下的应用性能监控

听云微服务化及监控

复杂调用链性能监控及追踪

微服务架构下的应用监控

APMCon

复杂微服务架构面临的问题

- 服务器体量激增，部署和管理问题
- 调用链复杂
- 监控复杂度大大增加，如何快速发现/定位问题？

微服务架构下的应用监控

APMCon

复杂微服务架构下的应用监控

性能监控优先

- 以听云Server为代表
- 关注性能及慢事务追踪
- 轻量，自动嵌码探针，SaaS，开箱即用
- 内置Http/dubbo/thrift等支持

vs

事务/调用链追踪优先

- 点评Cat, Zipkin（开源）
- 关注Tracing,全量或采样
- 较重，需埋点或开发插件，采集日志量大

Why micro services ?

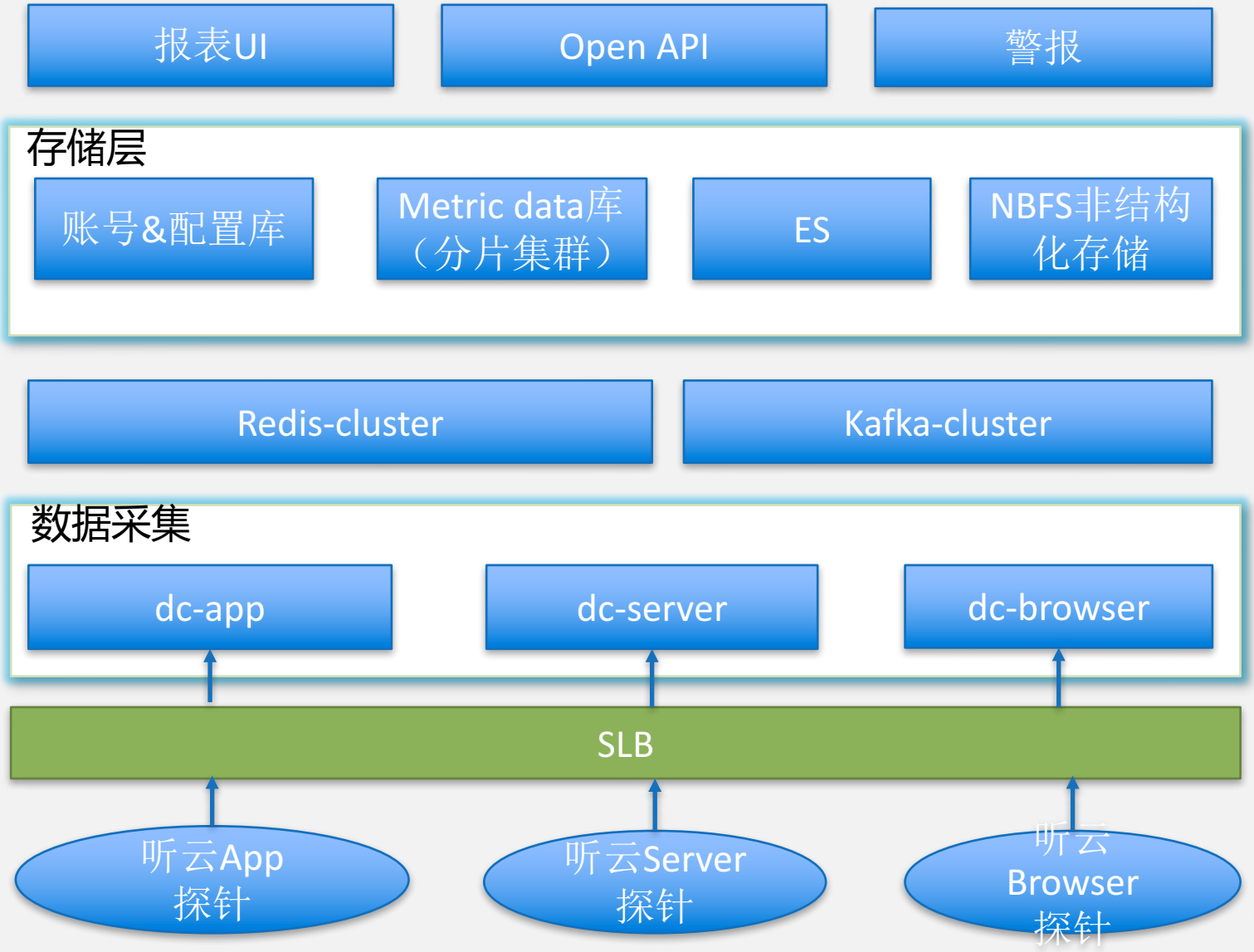
微服务架构下的应用性能监控

听云微服务化及监控

复杂调用链性能监控及追踪

听云的微服务化历程

APMCon



听云后端架构1.0

听云的微服务化历程

APMCon

1.0单体架构面临的问题

- 组件依赖多，迭代效率低下
  升级=开发*1d+功能测试*1d+回归*3d+部署*1h
- 核心组件升级周期长
- 单一配置库，DB问题影响多个系统，排查困难

听云的微服务化历程

APMCon

单体 -> 微服务架构：

- 核心组件微服务化

  nbfs/IP Location/Metric service

- 配置库按业务线垂直拆分

  账号及全局配置库：conf_global

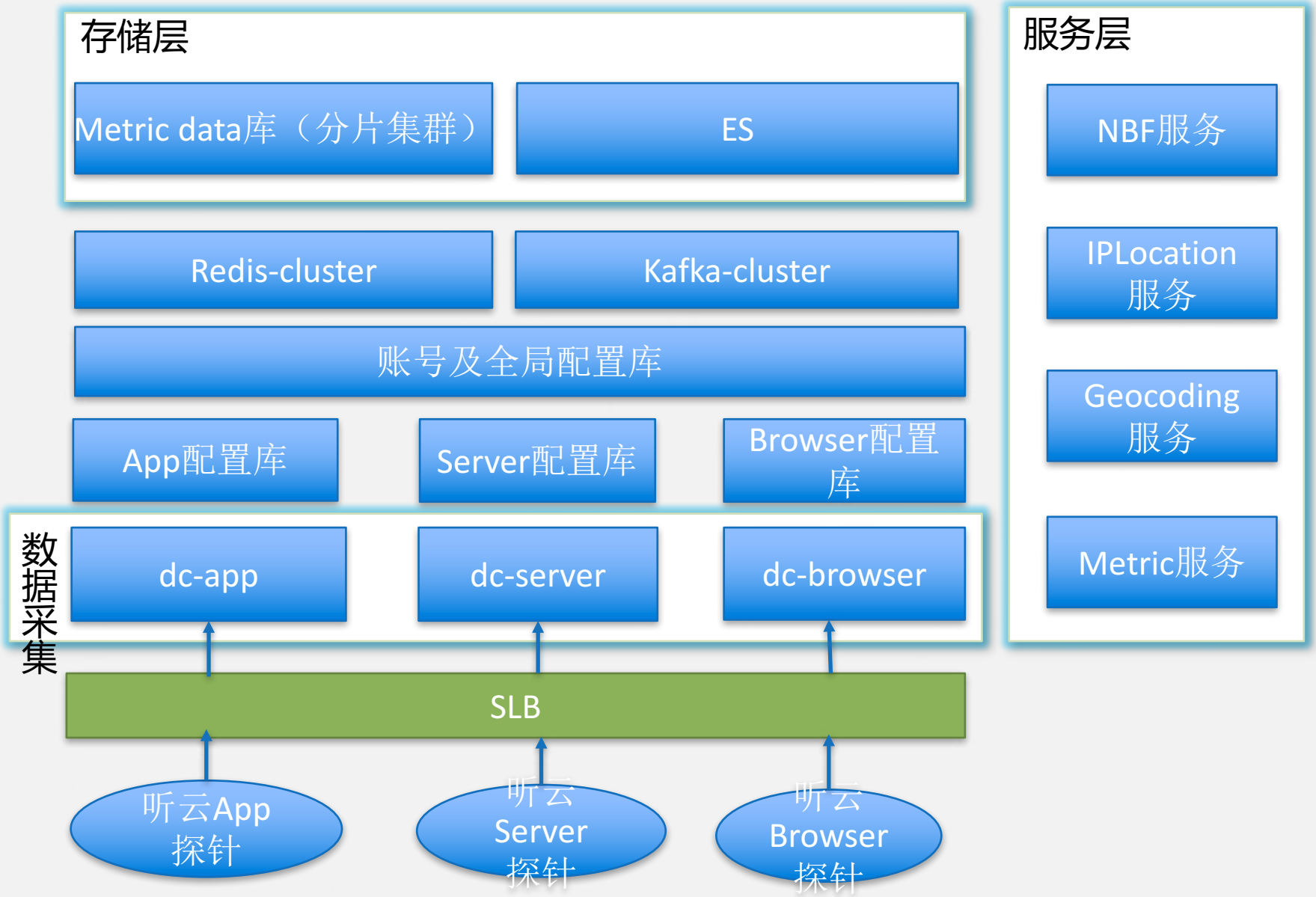
  业务线配置库：conf_app/conf_svr/conf_brs等

  听云警报服务配置库：conf_alarm

  * 跨业务线查询通过API接口调用

- 核心微服务按业务线资源隔离
- 日志统一入EFK

听云的微服务化历程



听云后端架构2.0

听云的微服务化历程

微服务化后的效果：

- 核心组件独立为原子服务，升级对应用几乎0影响
- 监控由面向应用调整为面向服务，粒度更细
- 可靠性高，核心组件对应用的性能影响更透明
- 配置库按业务线拆分，不同业务线数据库资源隔离

听云的微服务化历程

听云NBFS服务简介：

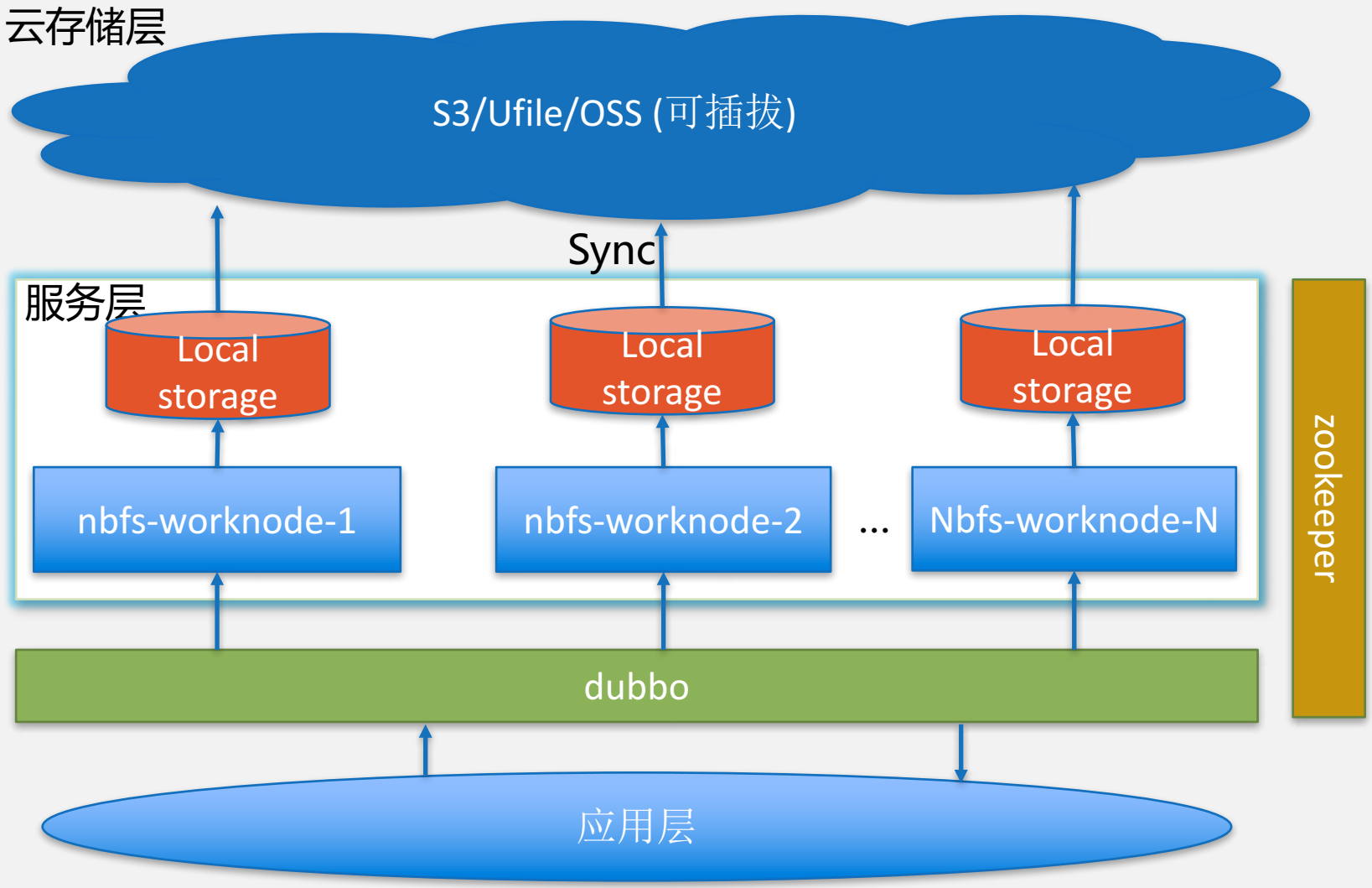
功能：

非结构化数据存储（类似于TFS或S3的功能）

场景：

1. 海量小对象存储(80% < 4KB)
2. 写多读少，写入延时要求高

听云的微服务化历程



听云NBFS服务化架构

听云微服务架构的应用性能监控

关注指标：

- API响应时长
- 吞吐率
- API响应耗时分解
- 网络层耗时（TODO）
- API调用排队时长（框架相关，例如dubbo，TODO）
- 慢请求堆栈及调用链

听云微服务架构的应用性能监控

APMCon

监控工具：

- cAdvisor（docker）
- Heapster（k8s）
- 听云Server (应用层)

听云微服务架构的应用性能监控

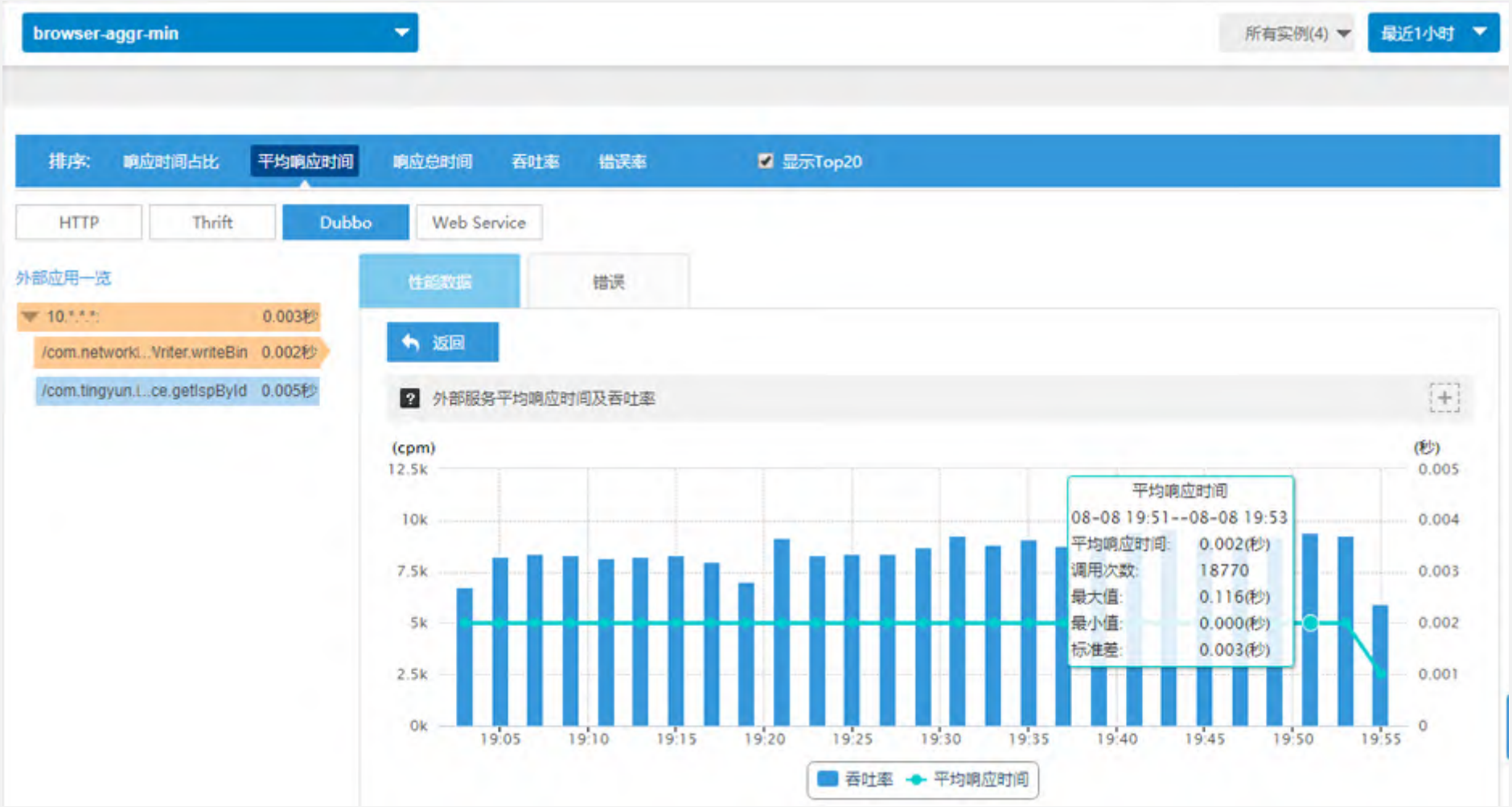
APMCon

自动发现应用拓扑

听云微服务架构的应用性能监控

APMCon

Client端服务调用监控



听云微服务架构的应用性能监控

APMCon

服务端监控

慢事务追踪列表

事务： 最大响应时间： 最小响应时间： 参数名： 参数值： 查询

提示：如根据参数搜索，参数名和参数值必须同时填写，或同时为空

序号	时间	事务	服务器响应时间(ms)
1	2017-08-10 11:35	DubboProvider/com.networkbench.nbfs.localfs.LocalNbfsWriter/writeText	592
2	2017-08-10 11:05	DubboProvider/com.networkbench.nbfs.localfs.LocalNbfsWriter/writeText	139
3	2017-08-10 11:09	DubboProvider/com.networkbench.nbfs.localfs.LocalNbfsWriter/writeText	114
4	2017-08-10 11:20	DubboProvider/com.networkbench.nbfs.localfs.LocalNbfsWriter/writeText	108
5	2017-08-10 11:14	DubboProvider/com.networkbench.nbfs.localfs.LocalNbfsWriter/writeText	103
6	2017-08-10 11:18	DubboProvider/com.networkbench.nbfs.localfs.LocalNbfsWriter/writeText	103
7	2017-08-10 11:32	DubboProvider/com.networkbench.nbfs.localfs.LocalNbfsWriter/writeText	101
8	2017-08-10 11:36	DubboProvider/com.networkbench.nbfs.localfs.LocalNbfsWriter/writeText	101
9	2017-08-10 10:51	DubboProvider/com.networkbench.nbfs.localfs.LocalNbfsWriter/writeText	101

听云微服务架构的应用性能监控

APMCon

案例分享：

业务高峰NBFS偶发性响应耗时突增，持续几秒~几分钟

听云微服务架构的应用性能监控

APMCon

事务性能分解堆叠图

周期性尖刺，平均：5.007ms，
最大：8129ms

代码段	性能分类	耗时百分比(%)	调用次数	响应时间(ms)
com.networkbench.nbfs.localfs.LocalNbfsWriter/writeText	DubboProvider	50.833	3509501	0
com.alibaba.dubbo.rpc.proxy.javassist.JavassistProxyFactory$1/d...	Java	26.786	3509501	0
ch.qos.logback.classic.encoder.PatternLayoutEncoder/doEncode	Custom	9.897	3509514	0
com.alibaba.dubbo.rpc.proxy.javassist.JavassistProxyFactory$1/in...	Java	6.756	3509501	0
ch.qos.logback.classic.Logger/debug	Custom	5.51	3509514	0
com.networkbench.nbfs.localfs.FileChannelStateInternal/initFileC...	Custom	0.031	6	50
java.nio.channels.AsynchronousFileChannel/open	Custom	0.021	4	51

AsynchronousFileChannel.open:
调用：4次
平均：51ms

听云微服务架构的应用性能监控

APMCon

慢事务追踪

事务：DubboProvider/com.networkbench.nbfs.localfs.LocalNbfsWriter/writeText

追踪时间：2017-08-04 23:06:31

服务器响应时间：0.103 (s)

实例信息：JAVA:svr-c1-m-200.ucd.tingyun.com:20883

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)
▼ AbstractProxyInvoker.invoke	103	100.00	0
▼ JavassistProxyFactory$1.invoke	103	100.00	0
▼ LocalNbfsWriter.writeText	103	100.00	0
▼ FileChannelStateInternal.getAndUpdatePosition	103	100.00	0
▼ FileChannelStateInternal.initFileChannelAndPositionIfNecessary	103	100.00	0
▶ Logger.debug	0	0.00	0
▼ LocalAsynchronousFileChannelManager.createFileChannel	103	100.00	0
▼ AsynchronousFileChannel.open	103	100.00	0
AsynchronousFileChannel.open	103	100.00	0
▼ Logger.debug	0	0.00	103
PatternLayoutEncoder.doEncode	0	0.00	103

AsynchronousFile Channel.open

Why micro services ?

微服务架构下的应用性能监控

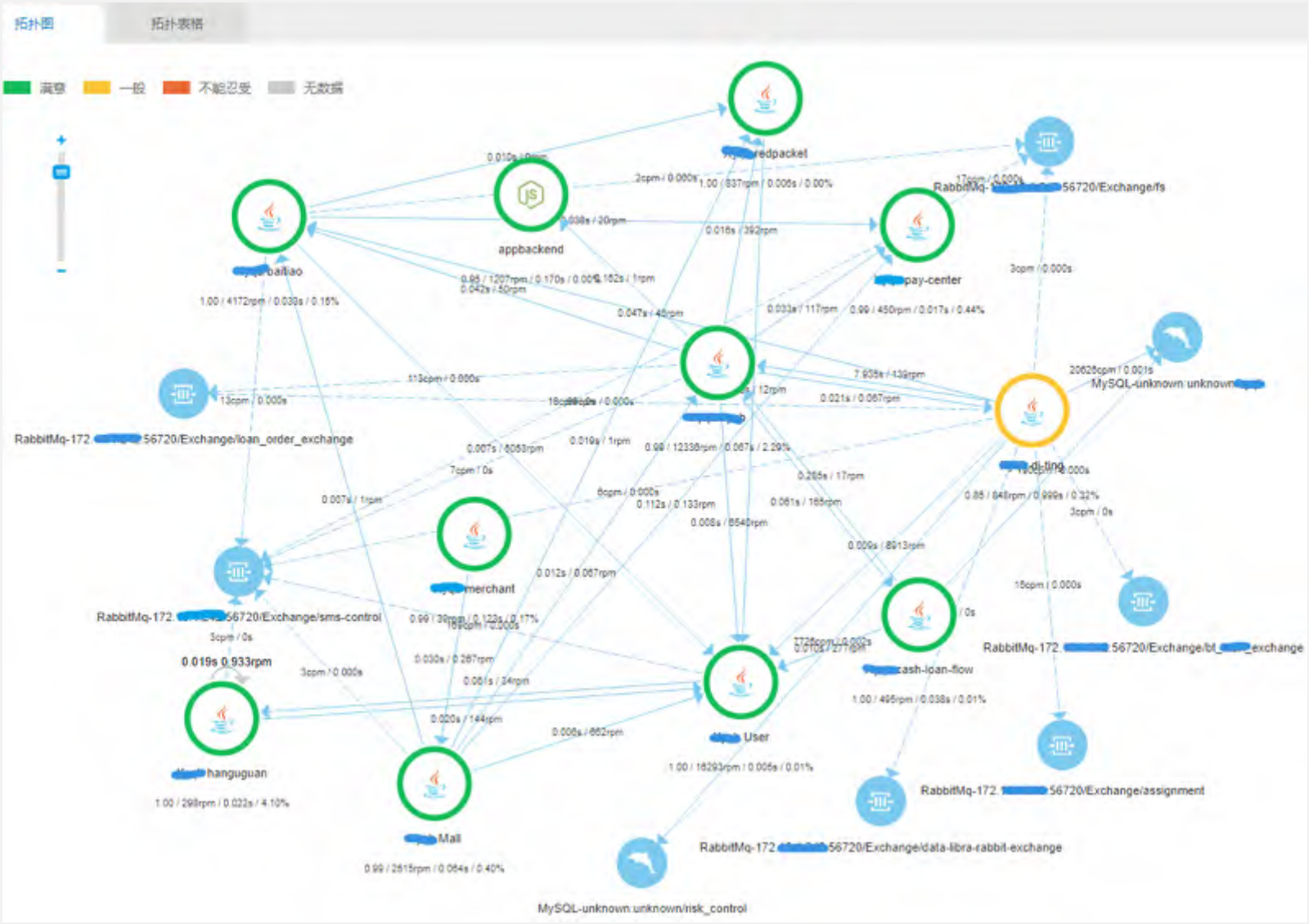
听云微服务化及监控

复杂调用链性能监控及追踪

复杂调用链的应用性能监控

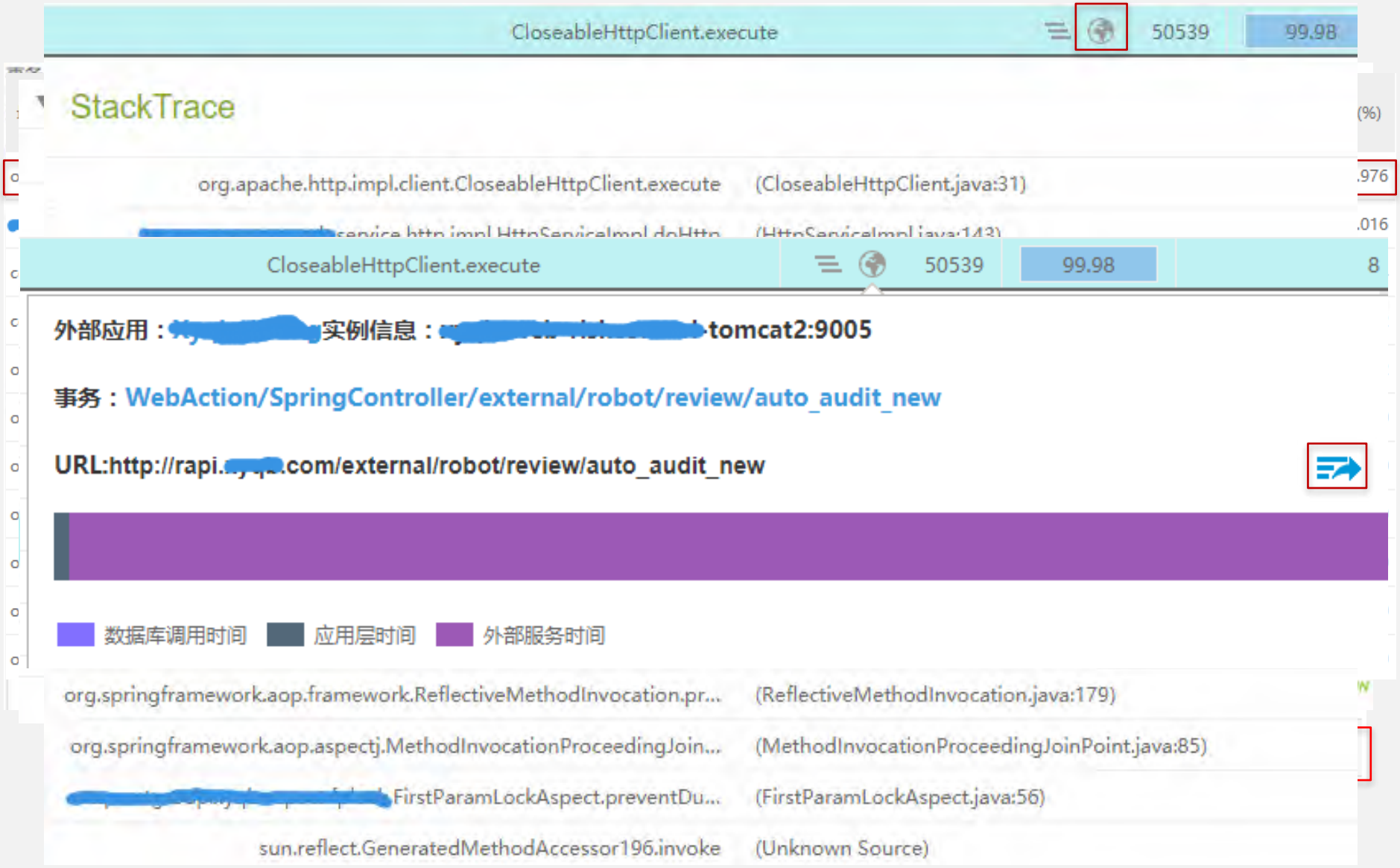
APMCon

复杂调用链拓扑



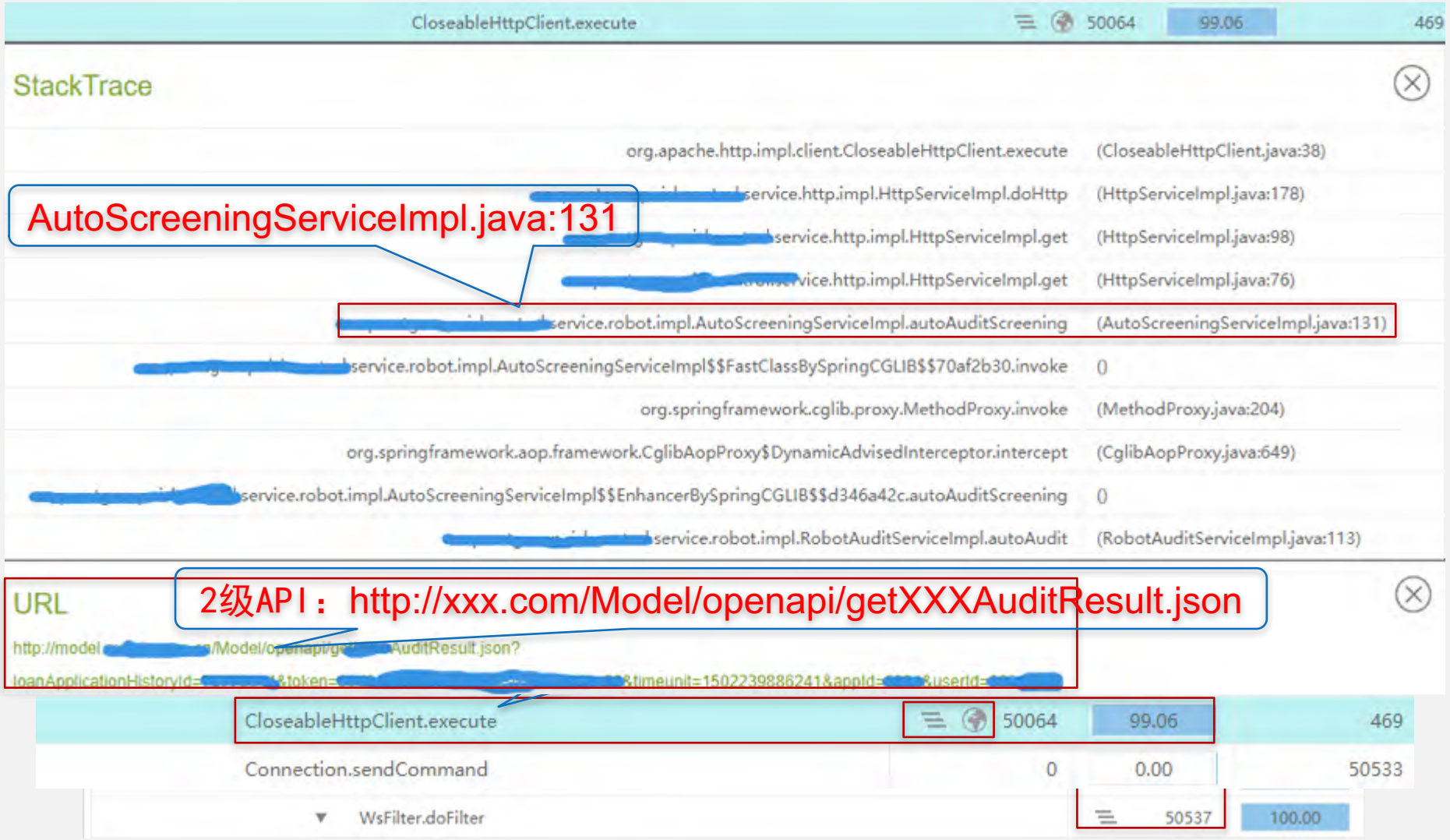
复杂调用链的应用性能监控

APMCon



复杂调用链的应用性能监控

APMCon



THANK YOU