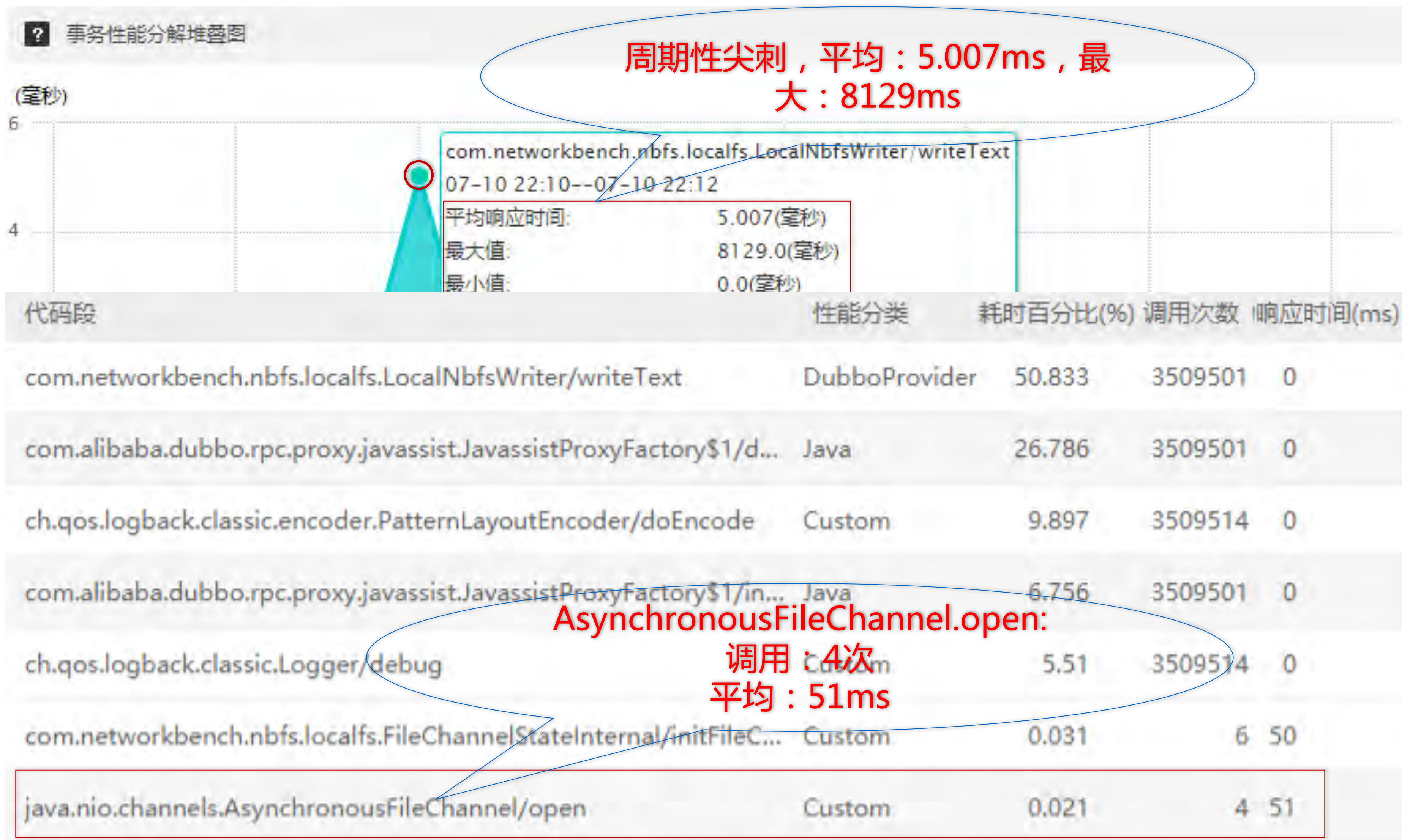


# 听云微服务架构的应用性能监控





# 听云微服务架构的应用性能监控

## 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

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

AsynchronousFileChannel.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

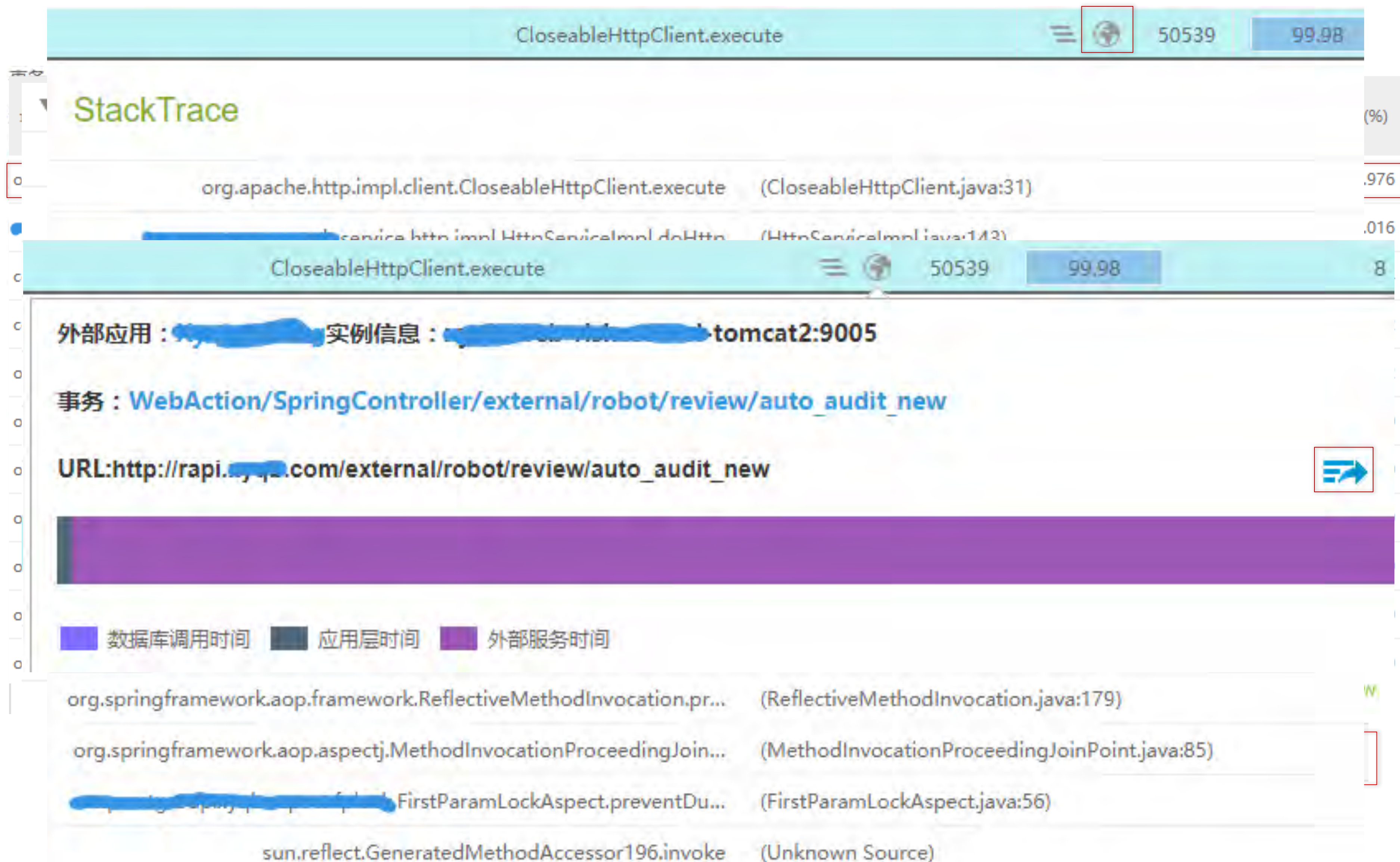
复杂调用链性能监控及追踪







# 复杂调用链的应用性能监控





# 复杂调用链的应用性能监控

The screenshot displays a performance monitoring interface with the following components:

- Header:** CloseableHttpClient.execute | 50064 | 99.06 | 469
- Stack Trace:**
  - org.apache.http.impl.client.CloseableHttpClient.execute (CloseableHttpClient.java:38)
  - ...service.http.impl.HttpServiceImpl.doHttp (HttpServiceImpl.java:178)
  - ...service.http.impl.HttpServiceImpl.get (HttpServiceImpl.java:98)
  - ...service.http.impl.HttpServiceImpl.get (HttpServiceImpl.java:76)
  - AutoScreeningServiceImpl.java:131** (highlighted in red)
  - ...service.robot.impl.AutoScreeningServiceImpl.autoAuditScreening (AutoScreeningServiceImpl.java:131)
  - ...service.robot.impl.AutoScreeningServiceImpl\$\$FastClassBySpringCGLIB\$\$70af2b30.invoke ()
  - org.springframework.cglib.proxy.MethodProxy.invoke (MethodProxy.java:204)
  - org.springframework.aop.framework.CglibAopProxy\$DynamicAdvisedInterceptor.intercept (CglibAopProxy.java:649)
  - ...service.robot.impl.AutoScreeningServiceImpl\$\$EnhancerBySpringCGLIB\$\$d346a42c.autoAuditScreening ()
  - ...service.robot.impl.RobotAuditServiceImpl.autoAudit (RobotAuditServiceImpl.java:113)
- URL:**
  - 2级API: http://xxx.com/Model/openapi/getXXXAuditResult.json** (highlighted in red)
  - http://model.../Model/openapi/get...AuditResult.json?
  - loanApplicationHistoryId=...&token=...&timeunit=1502239886241&appId=...&userId=...
- Summary Table:**

Method	Count	Time (ms)	Percentage
CloseableHttpClient.execute	50064	99.06	469
Connection.sendCommand	0	0.00	50533
WsFilter.doFilter	50537	100.00	

# 小结

- 微服务架构的利弊
- 微服务架构的拆分是自然而然的过程，如何划分边界有一些通用的原则，但主要还是视业务特性而定
- 微服务架构监控粒度：性能监控优先 OR 事务/调用链优先？前者实现代价低的多，可满足大部分业务系统的监控需求
- 代码级定位是必不可少的，可有效缩短MTTR





关注QCon微信公众号  
获得更多干货!

# Thanks!

INTERNATIONAL SOFTWARE DEVELOPMENT CONFERENCE

主办方: **Geekbang** > **InfoQ**  
极客邦科技