



京东技术

IT大咖说  
知识共享平台

# 微服务架构监控及监控原理介绍

2017/12/23



京东技术



- 1 微服务架构下的应用性能监控
- 2 监控平台架构及原理介绍





## 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？



## 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



## 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



# 微服务架构下的应用监控



京东技术

IT大咖说  
知识共享平台

监控工具：

- cAdvisor ( docker )
- Heapster ( k8s )
- 微服务应用



京东技术

# 微服务架构下的应用监控



京东技术

IT大咖说  
知识共享平台

## 搭建监控体系：

```
FROM registry.tingyun.com/common/ceonts7-jdk6
MAINTAINER netop "netop@tingyun.com"
ENV REFRESHED_AT 2016-03-14

ADD apache-tomcat-6.0.45.tar.gz /opt
RUN ln -s /opt/apache-tomcat-6.0.45 /opt/apache-tomcat
ADD server.xml /opt/apache-tomcat/conf/server.xml

ENV CATALINA_HOME /opt/apache-tomcat
ENV PATH $CATALINA_HOME/bin:$PATH
ENV TOMCAT_MAJOR 6
ENV TOMCAT_VERSION 6.0.45
ENV JAVA_OPTS -server -Djava.library.path=/opt/apache-tomcat/lib -XX:+UseConcMarkSweepGC -XX:+PrintGCDetails -XX:+PrintGCDateStamps -Xloggc:/opt/apache-tomcat/logs/gc.log

WORKDIR $CATALINA_HOME

ENV TINGYUN_APP_NAME="Java Application" \
TINGYUN_LICENSE_KEY=f451bae910b4c3f4473e0c83d2113742

RUN curl -o /tingyun-agent-java-latest.zip http://download.tingyun.com/agent/java/latest/tingyun-agent-java-latest.zip
RUN yum install unzip -y
RUN mkdir /opt/tingyun_agent;unzip /tingyun-agent-java-latest.zip -d /opt/tingyun_agent
RUN rm /tingyun-agent-java-latest.zip -rf

ENV CONNECTOR='<Connector port="8080" maxHttpHeaderSize="8192"
maxThreads="300" minSpareThreads="25" maxSpareThreads="300"
enableLookups="false" redirectPort="8443" acceptCount="100"
connectionTimeout="20000" disableUploadTimeout="true" />'
ENV CONTEXT=""
ADD run.sh /run.sh
#RUN /usr/bin/chmod 755 /run.sh

EXPOSE 8080

#name: registry.tingyun.com/app/report-app-alpha1:369
#image: registry.tingyun.com/app/report-app-alpha1:369
#command:
#- sh
#- in -s /opt/mapping_files/puid /bin/puid && rm -rf /opt/tingyun/report-app/work/* && /run.sh

#name: TINGYUN_APP_NAME
#value: app-report-app-release
#name: TINGYUN_LICENSE_KEY
#value: f451bae910b4c3f4473e0c83d2113742
#name: JAVA_OPTS
#value: -server -DappName=report-app -Djava.library.path=/opt/apache-tomcat/lib -Xms128m -Xmx2048m -XX:NewRatio=4 -Xs1024k -XX:PermSize=64M -XX:MaxPermSize=256M -XX:+UseConcMarkSweepGC -XX:+PrintGCDateStamps -Xloggc:/opt/apache-tomcat/logs/gc.log -javaagent:/opt/tingyun_agent/tingyun/tingyun-agent-java.jar -Duser.language=zh -Duser.region=cn

#name: CONNECTOR
#value: '<Connector port="8080" maxHttpHeaderSize="8192" maxThreads="300" minSpareThreads="25" maxSpareThreads="300" scheme="https" proxyPort="443" enableLookups="false" redirectPort="8443" acceptCount="100" connectionTimeout="20000" disableUploadTimeout="true" />'
#name: CONTEXT
#value: '-context path="/enable" redirectable="false" allowLinking="true" docBase="/opt/tingyun/report-app/webapp/" workDir="/opt/tingyun/report-app/work/">'
#name: ACCESS_LOG
#value: ''

#volumesMounts:
#name: config-volume
#mountPath: /opt/tingyun/report-app/conf
#name: pv-mapping-files
#mountPath: /opt/mapping_files
#name: log
#mountPath: /opt/tingyun/report-app/logs
#name: share
#mountPath: /opt/tingyun/report-app/work/

/bin/bash

TINGYUN_AGENT_PATH="/opt/tingyun_agent/tingyun/tingyun.properties"
TOMCAT_CONF_PATH="/opt/apache-tomcat/conf/server.xml"

sed -i "s/Java Application/$TINGYUN_APP_NAME/" $TINGYUN_AGENT_PATH
sed -i "s/<%=license_key%>/$TINGYUN_LICENSE_KEY/" $TINGYUN_AGENT_PATH

sed -i "s#CONNECTOR#CONNECTOR#" $TOMCAT_CONF_PATH
sed -i "s#CONTEXT#CONTEXT#" $TOMCAT_CONF_PATH

catalina.sh run
```

京东技术

# 微服务架构下的应用监控



京东技术

IT大咖说  
知识共享平台

案例分享：

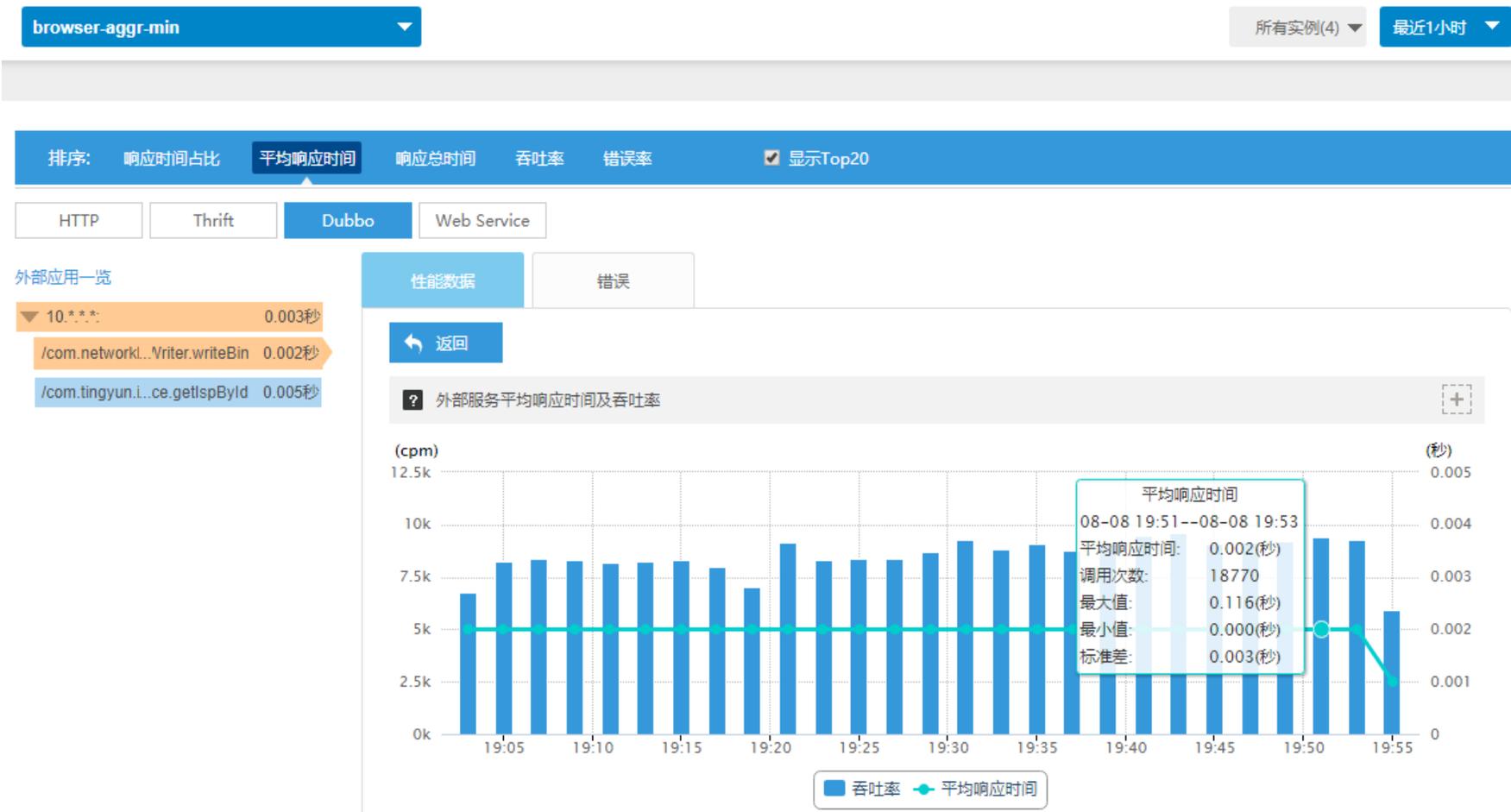
Dubbo服务client端到server端问题追踪



京东技术



## Client端服务调用监控





## 服务端监控

server\_server-nbf

事务一

事务二

事务三

事务四

事务五

事务六

事务七

事务八

事务九

### 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:

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

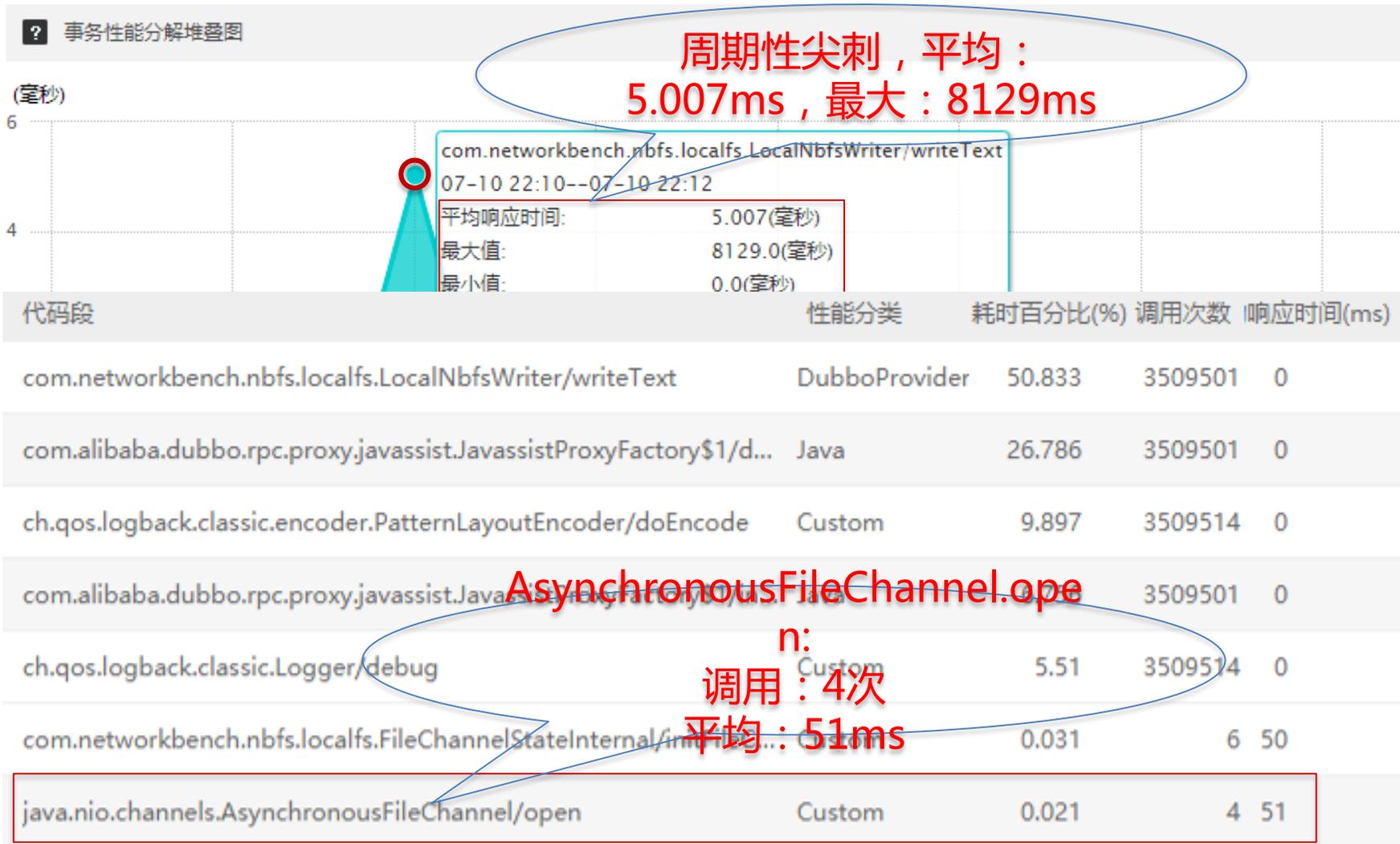


案例分享：

业务高峰NBFS偶发性响应耗时突增，持续几秒~几分钟



# 微服务架构下的应用监控



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

# 微服务架构下的应用监控



京东技术

IT大咖说  
知识共享平台

## 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

查询

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

AsynchronousFileChannel.open

服务器响应时间(ms)
170
115
107
106
106
104
103

京东技术

# 微服务架构下的应用监控

案例分享：

复杂调用链问题追踪

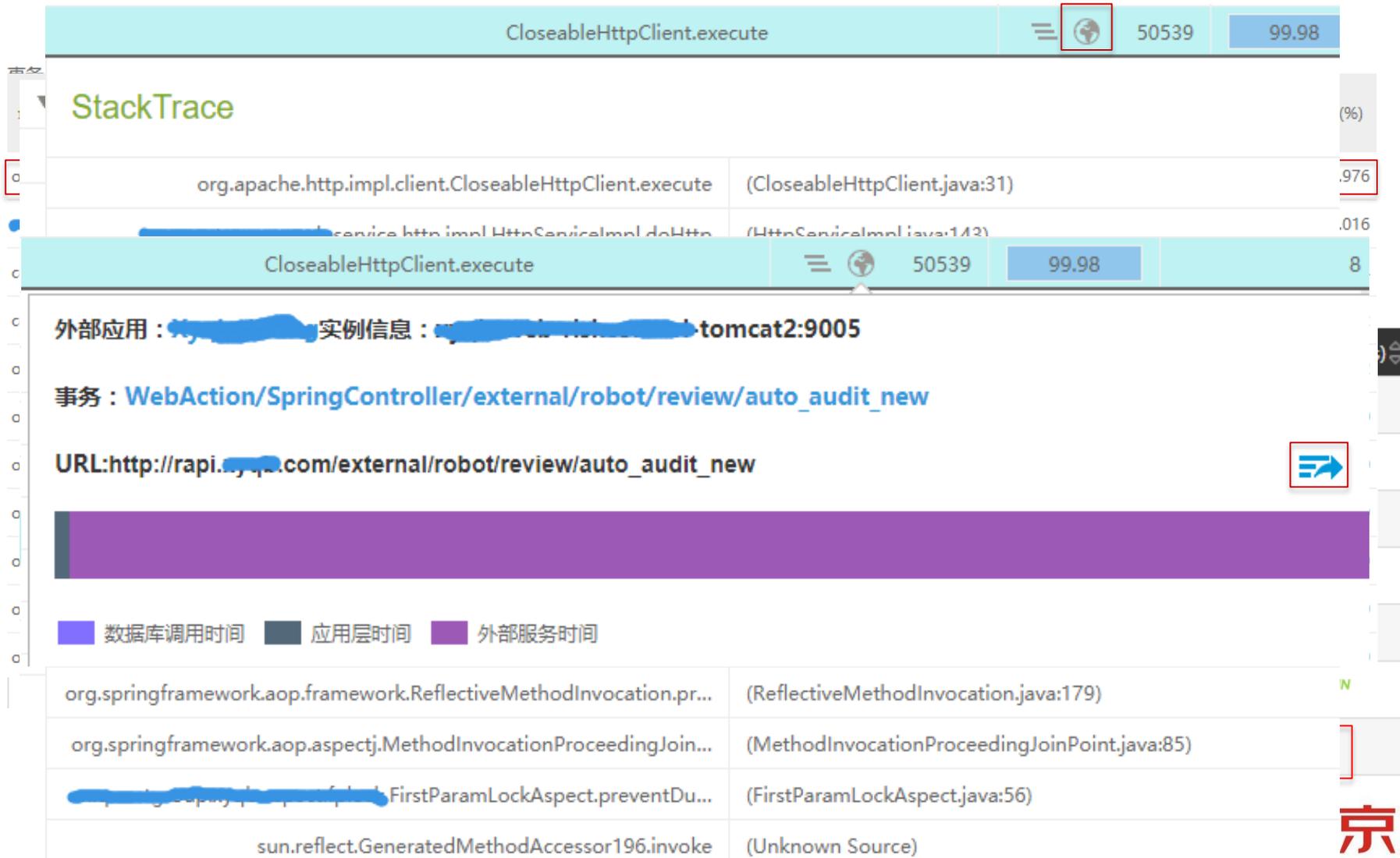


# 复杂调用链的应用性能监控



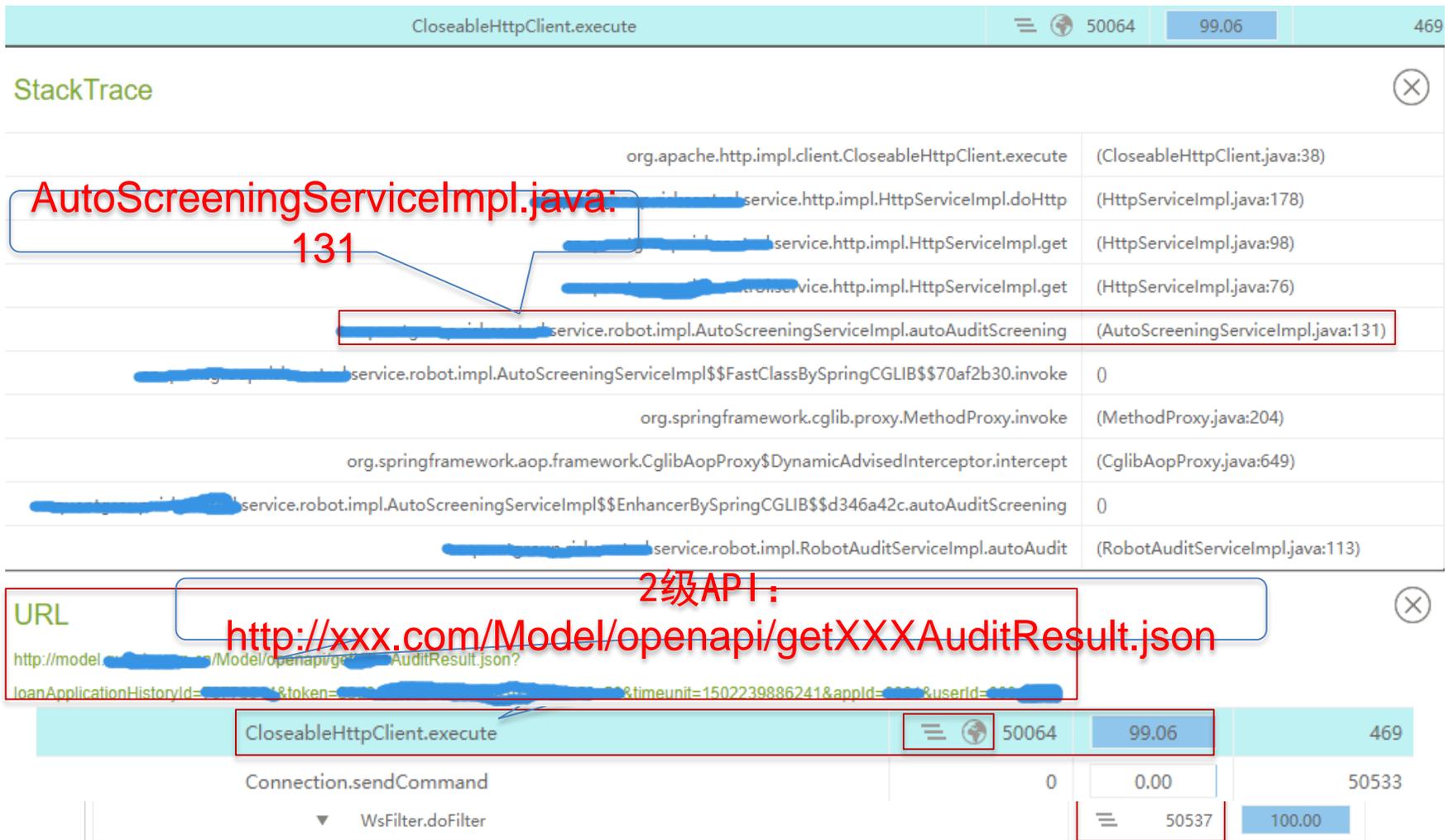
京东技术

IT大咖说  
知识共享平台



京东技术

# 复杂调用链的应用性能监控

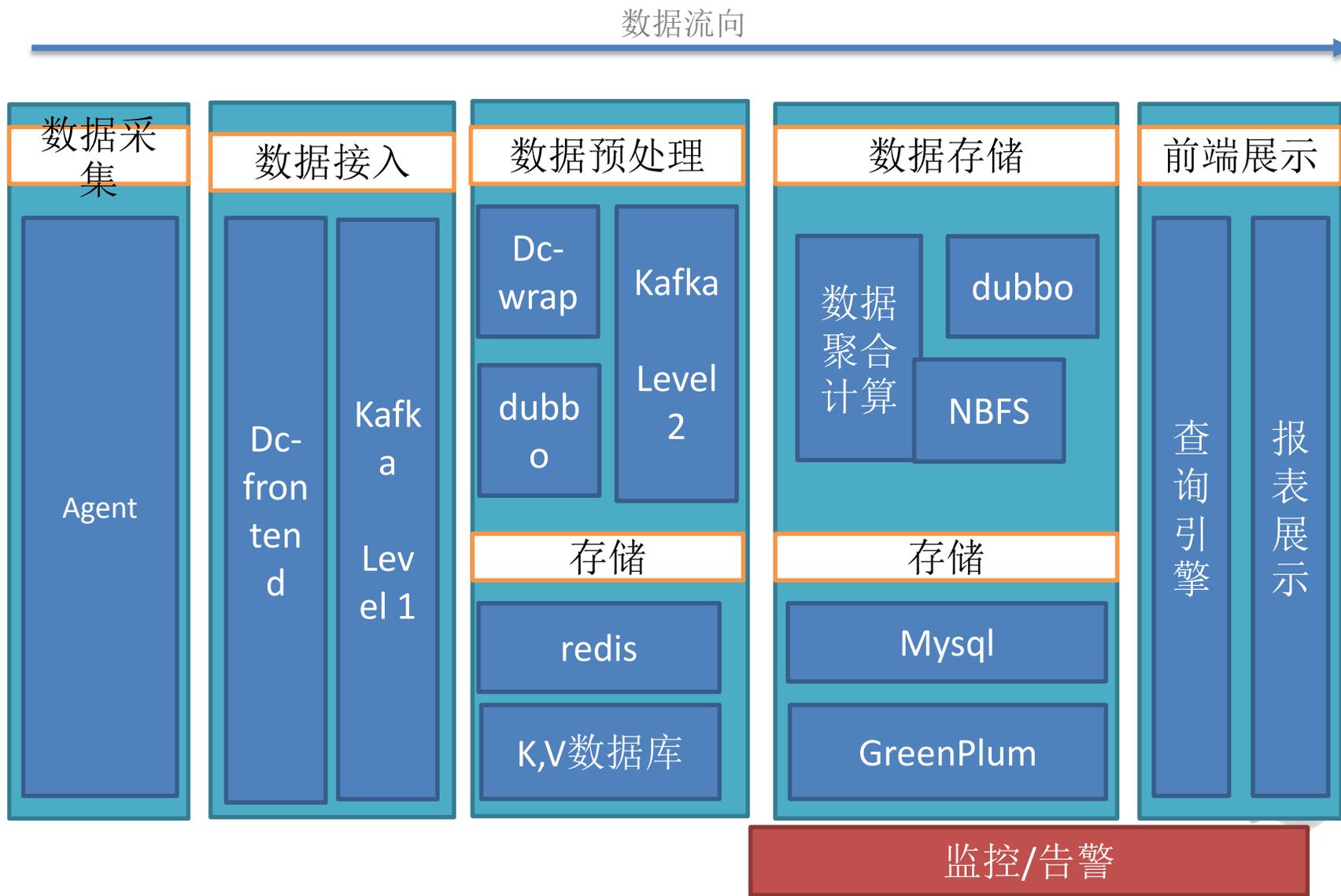


# 监控平台数据流向



京东技术

IT大咖说  
知识共享平台



京东技术

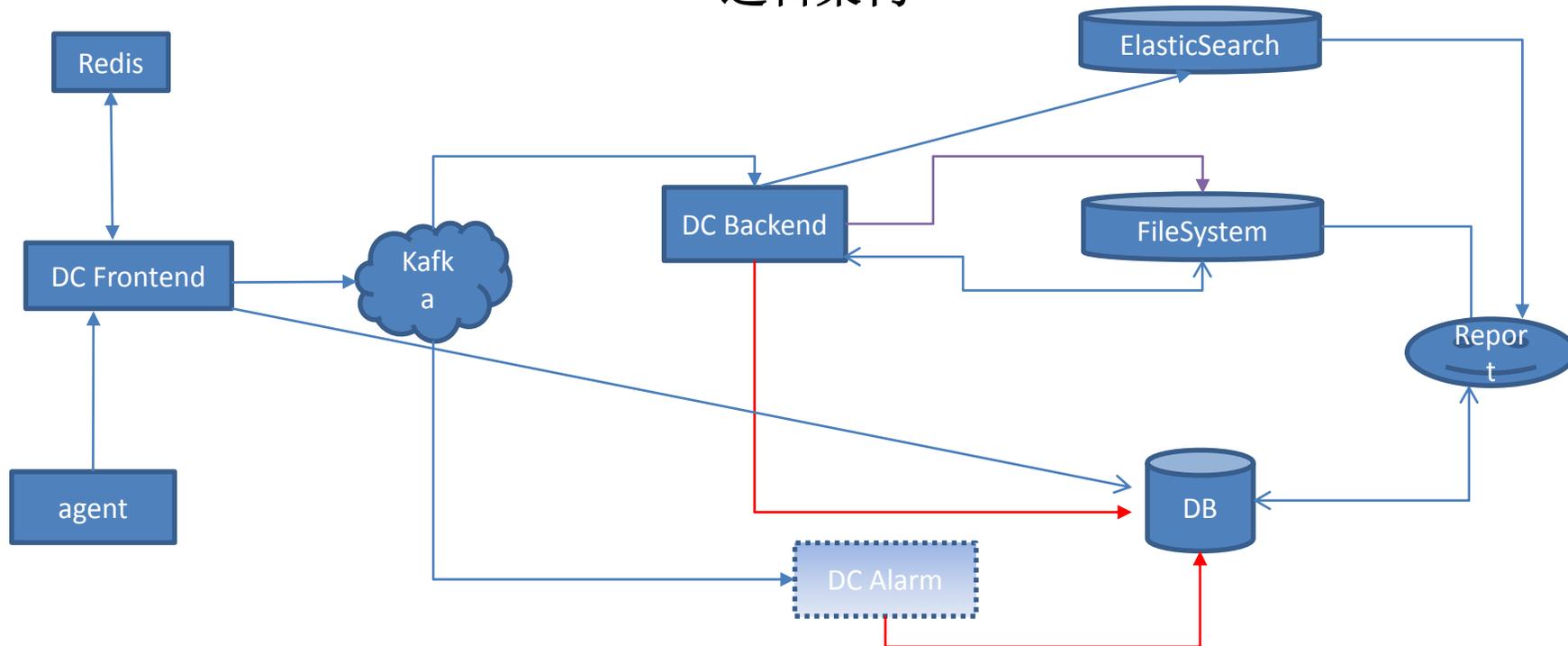
# 监控平台逻辑架构



京东技术

IT大咖说  
知识共享平台

## 逻辑架构



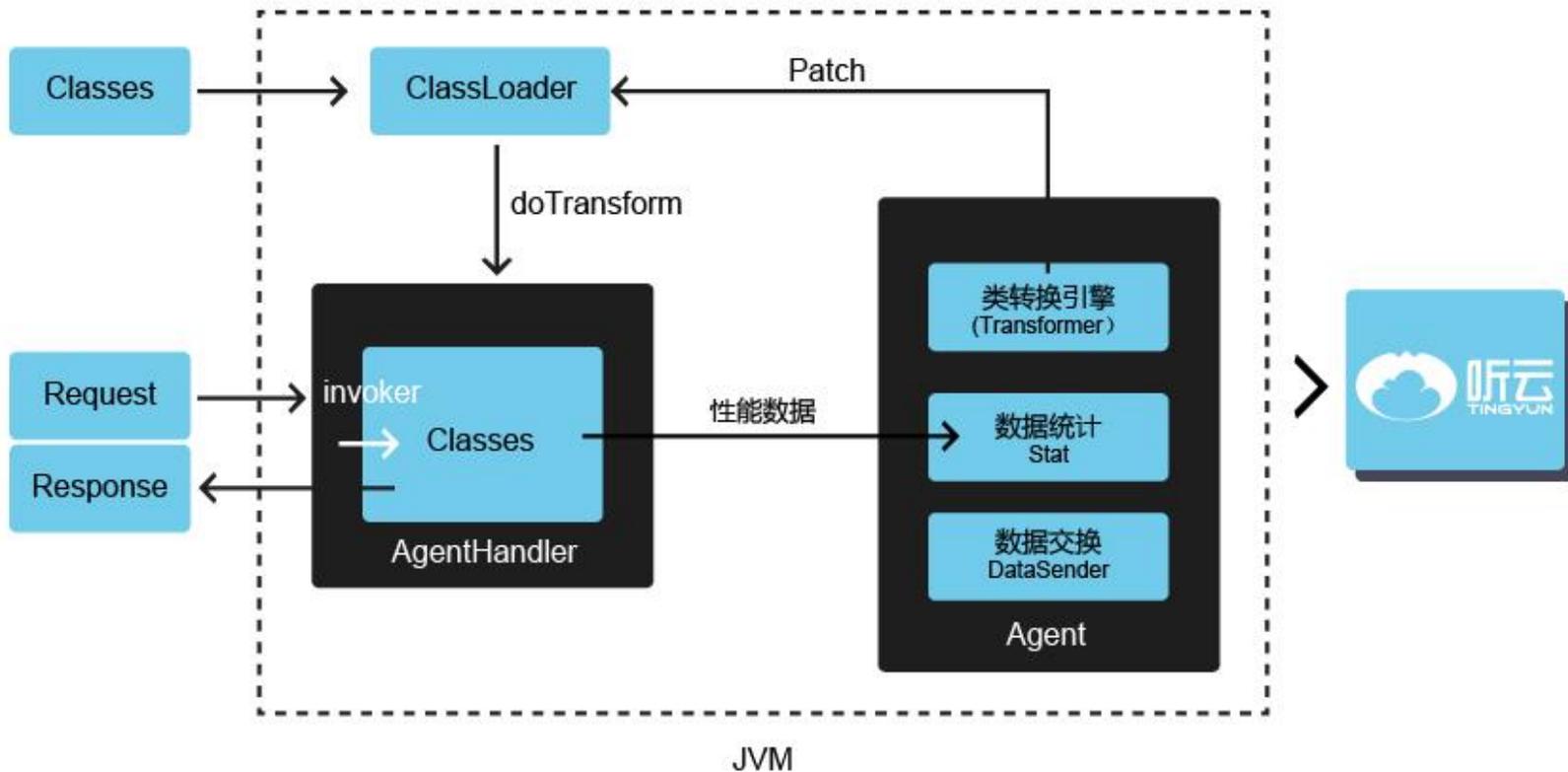
京东技术

# 监控平台数据采集原理



京东技术

IT大咖说  
知识共享平台



Java Agent基于JDK 提供的Instrumentation机制，在class文件被加载的时候，通过字节码技术，动态对Framework、数据库、NoSQL、Web Service、组件等特定方法实施监控，从而获得方法执行时间、数据库调用时间、NoSQL响应时间以及外部服务响应时间；并在这些时间超过一定阈值时，抓取调用堆栈。



京东技术

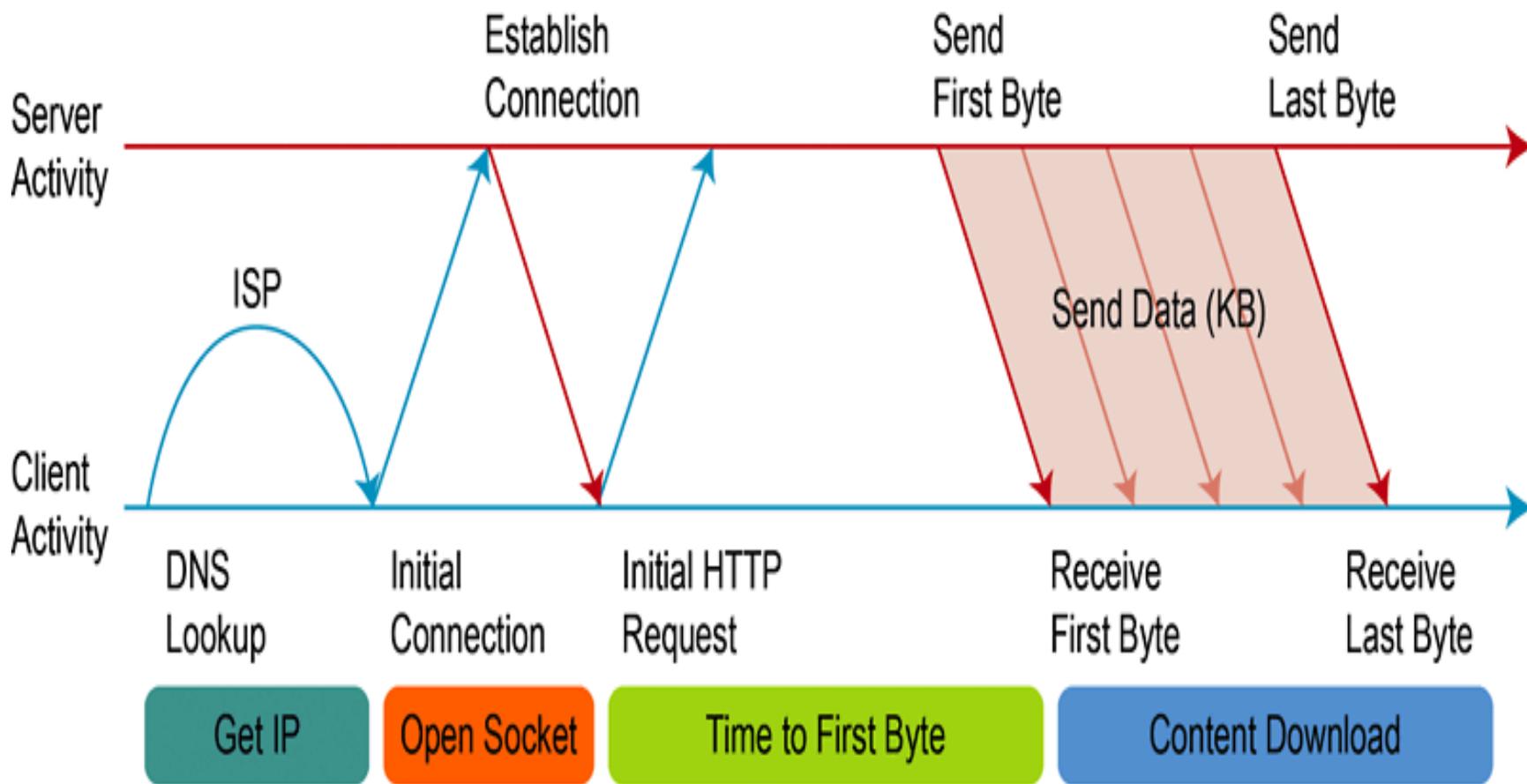
# 跨应用跨语言实现原理



京东技术

IT大咖说  
知识共享平台

## The HTTP Request



京东技术



京东技术

IT大咖说  
知识共享平台

感谢您的时间

THANKS



京东技术