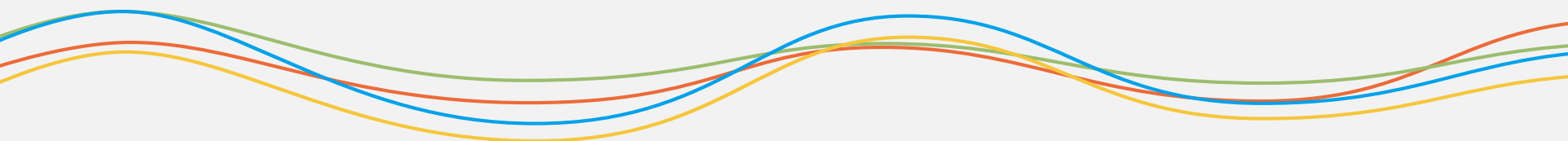


听云平台业务数据实时处理及 性能可视化

听云 - 任燕萍

The bottom of the slide features a decorative graphic consisting of several overlapping, wavy lines in blue, green, orange, and yellow, creating a sense of motion and data flow.



功能:

- ◆ 最终用户体验监控
- ◆ 全站溯源交易追踪
- ◆ 大数据行业基线对标
- ◆ 人工智能自动根因分析

特点:

- ◆ 实时
- ◆ 自动
- ◆ 端到端

主动监测您的应用性能



网络指标

- 基础页面下载时间
- 网络层时间
- 重定向时间
- 发出请求时间
- 总下载时间
- DNS时间
- 建立连接时间
- SSL握手时间
- 首包时间
- 客户端时间
- 内容下载时间

流量指标

- 总下载字节数
- 基础页面下载字节数
- 基础页面下载速度
- 首屏下载字节数

内容指标

- 页面对象数
- 首屏对象数
- 首屏时间
- DOM元素个数

流媒体指标

- 监测时长
- DNS时间
- 缓冲前准备时间
- TCP连接时间
- 缓冲时间
- 再次缓冲时间
- 等待时间
- 首次播放时间
- 播放时间
- 总下载字节数
- 收到第一个包时间
- 内容下载时间
- 用户体验指数
- 比特率
- 页面流媒体首播时间

PING

- 延时
- 丢包率

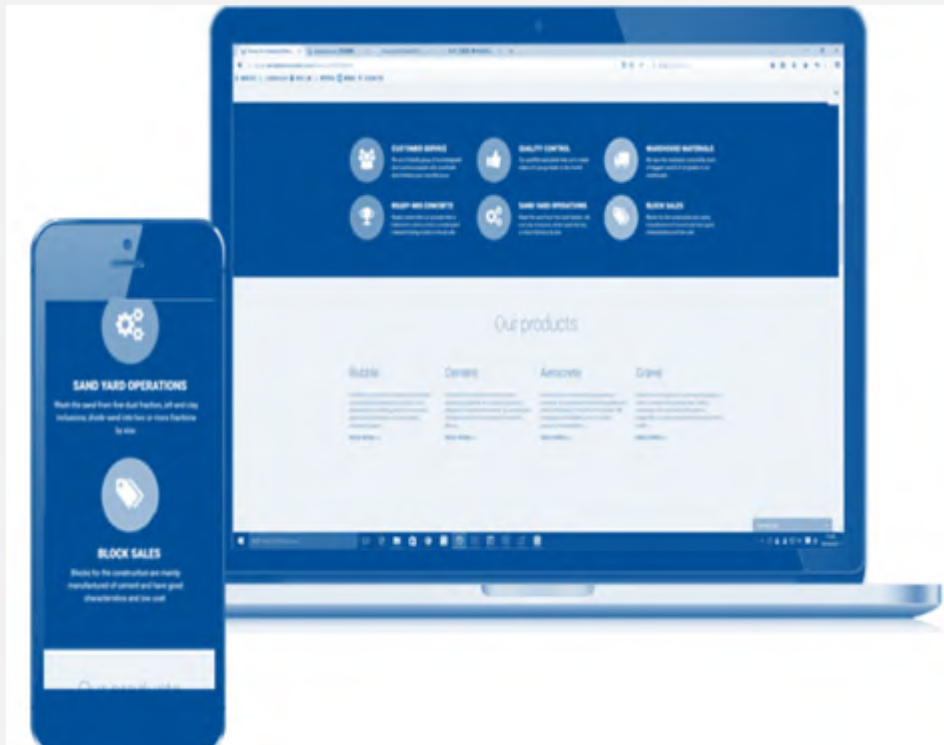
其他

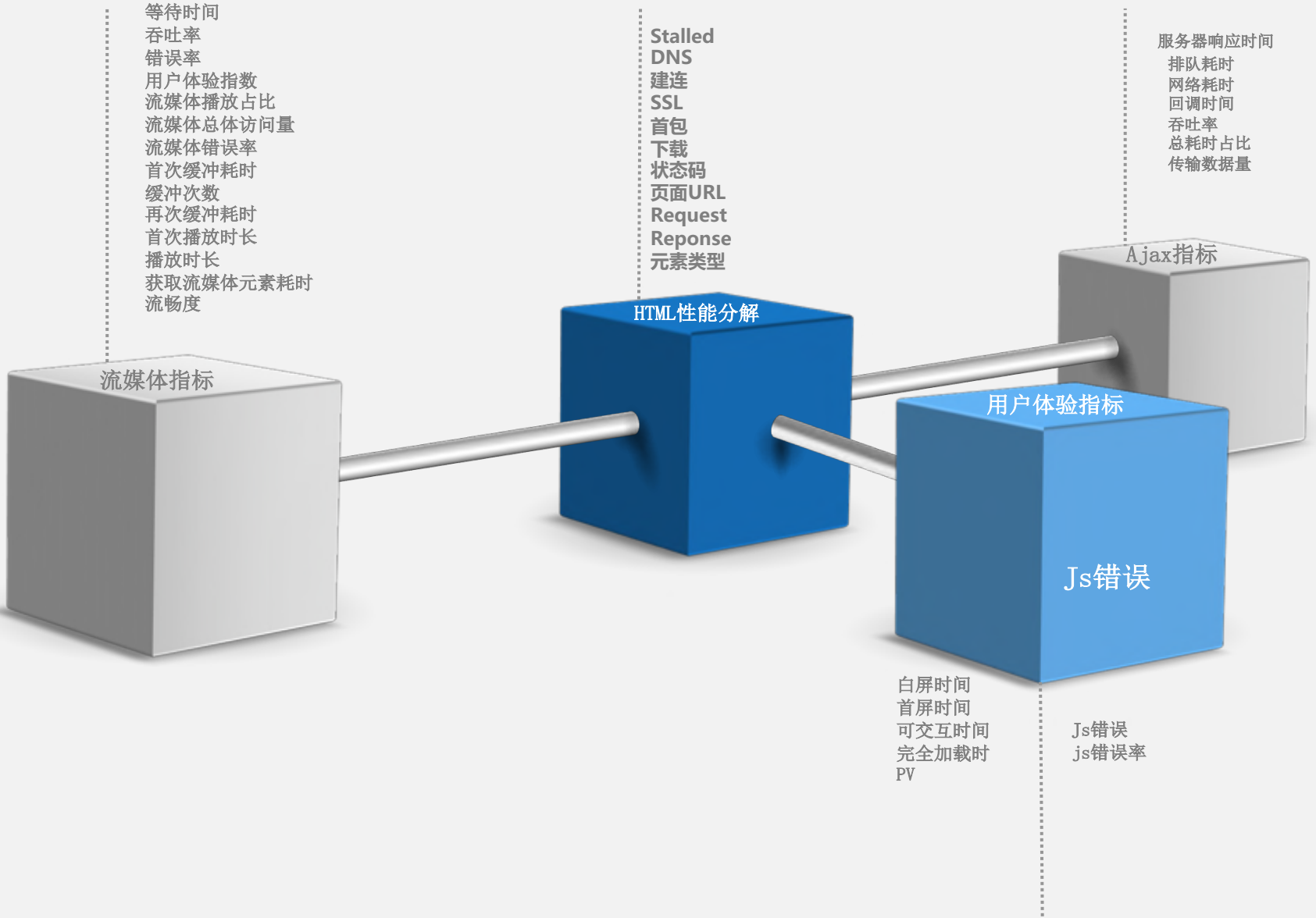
- 城市
- 运营商
- 监测点IP
- DNS服务器
- 目标IP
- 错误代码
- 内容错误
- 截屏
- Ping
- TraceRoute



基于真实用户的浏览器网站性能监控

- 网络图片临摹改编，支持浏览器监控
- 支持微信，混合式App的HTML5页面监控
- 监控需要验证登录的Web应用，如支持购物流程
- 支持内网监控
- 提供业务访问数据分析，如用户分布，PV等





听云APP 概述



您的App总有问题存在



崩溃

- 崩溃详情
- 崩溃轨迹
- 调用堆栈
- 上下文信息

交互分析

- 慢动作分析
- 慢动作详情
- 慢交互分解

内存

- CPU
- 网络
- 主线程
- 子线程

网络请求据

- 响应时间
- 吞吐量
- DNS时间
- TCP时间
- SSL时间
- 网络延时
- 首包时间
- Socket

主机错误

- HTTP错误
- 网络错误

ANR

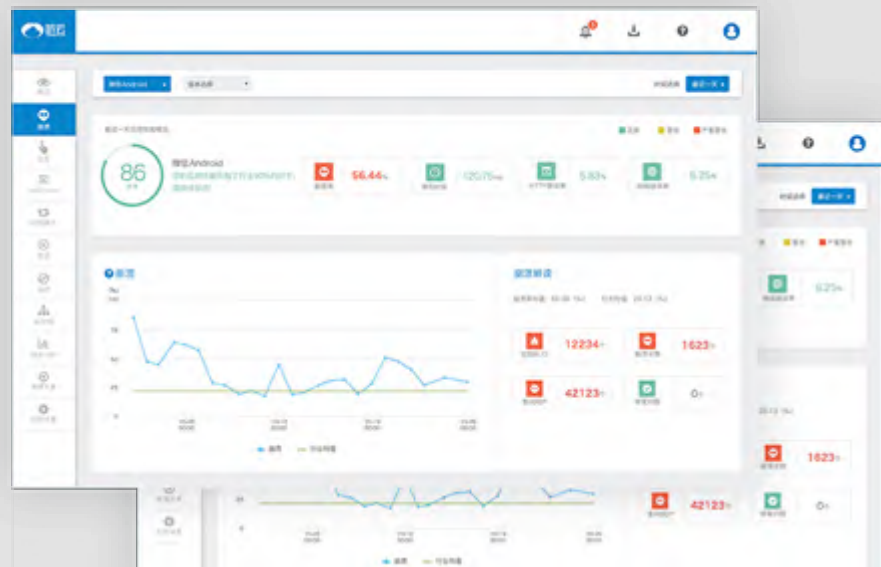
- 卡顿详情
- 卡顿轨迹
- 主线程
- 上下文信息

听云Server 概述

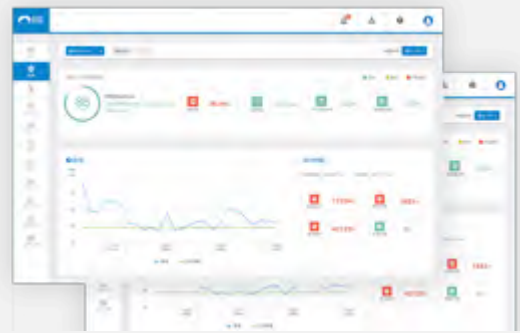
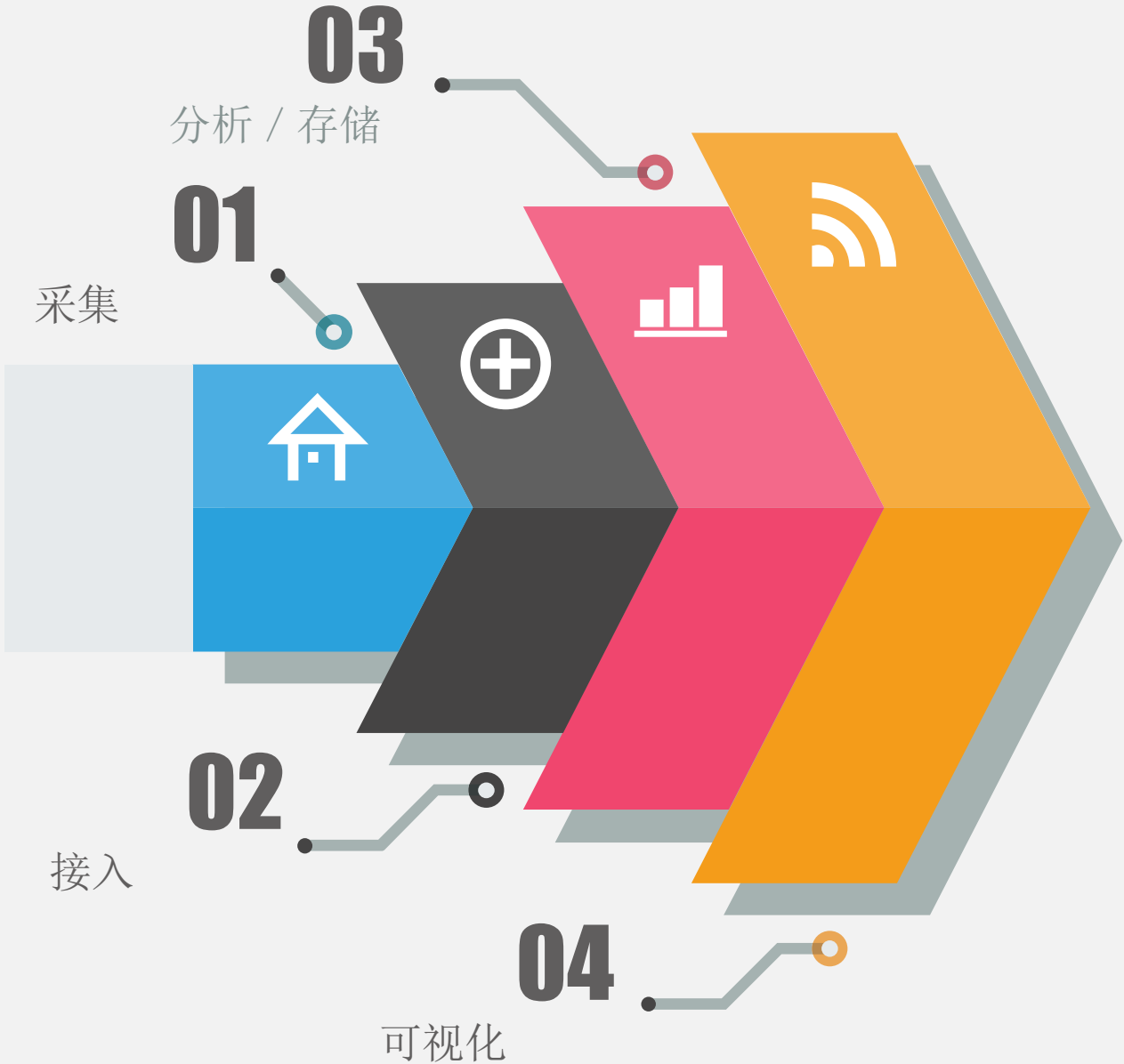


听云server是您服务端应用的私人医生

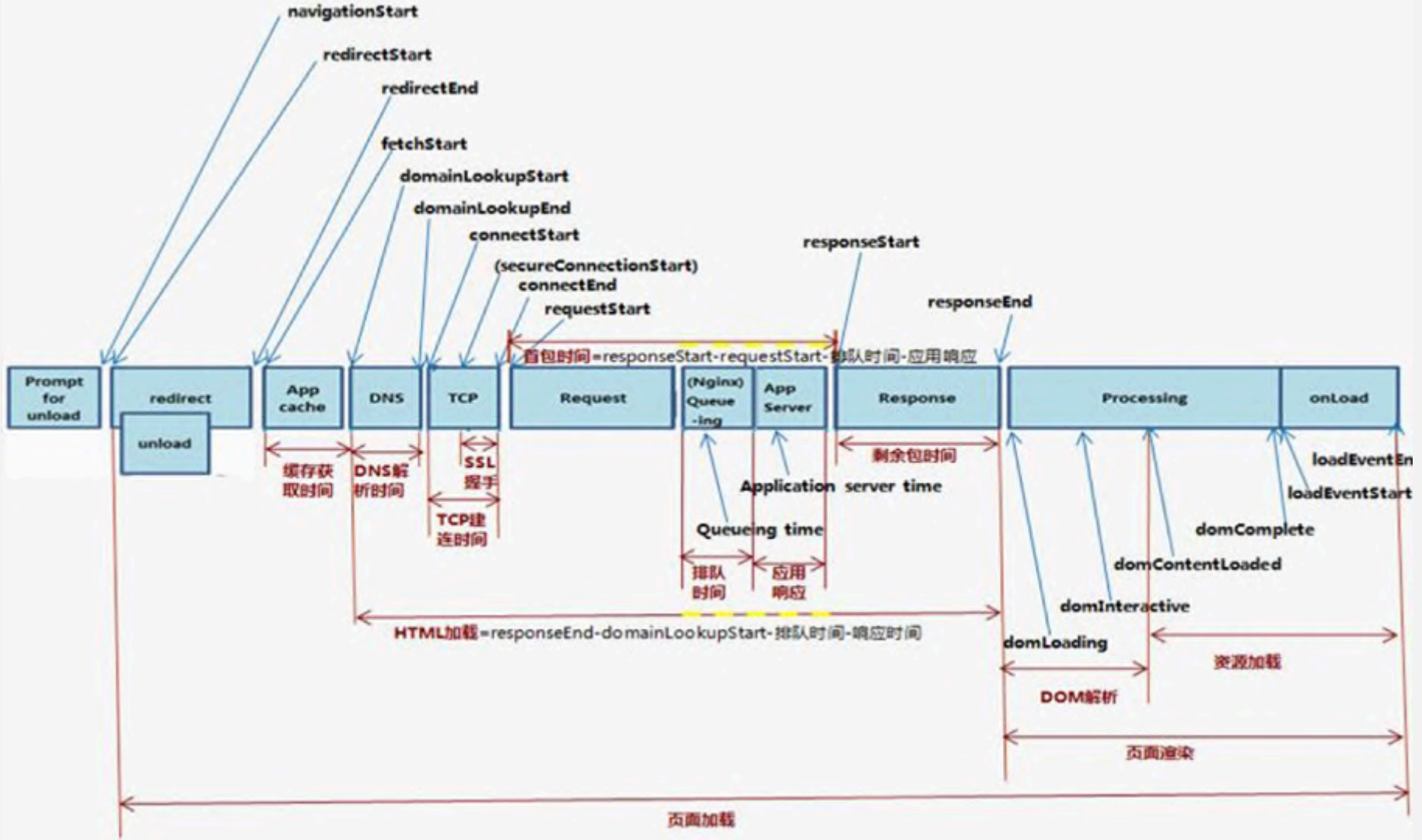
- 代码级定位性能问题
- 解决数据库性能差应用程序访问慢
- 代码报错
- API接口等性能问题



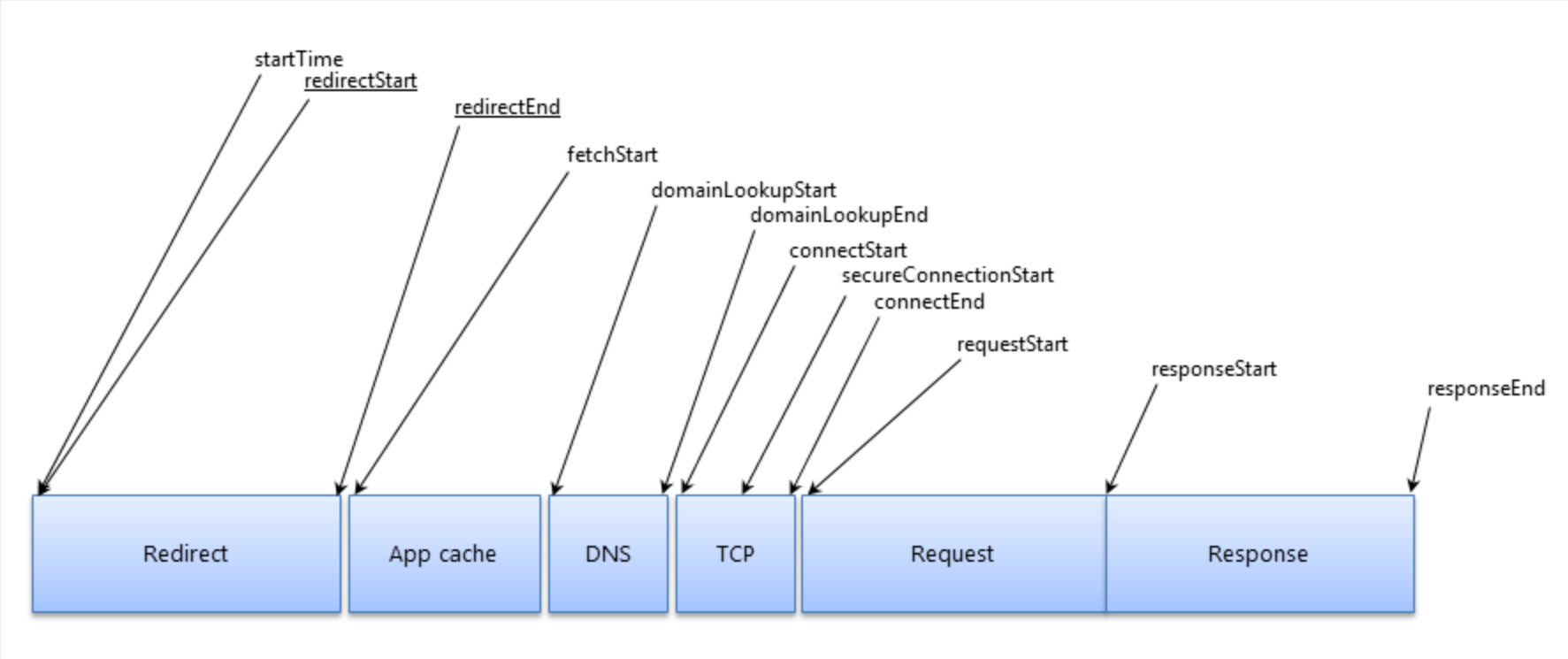


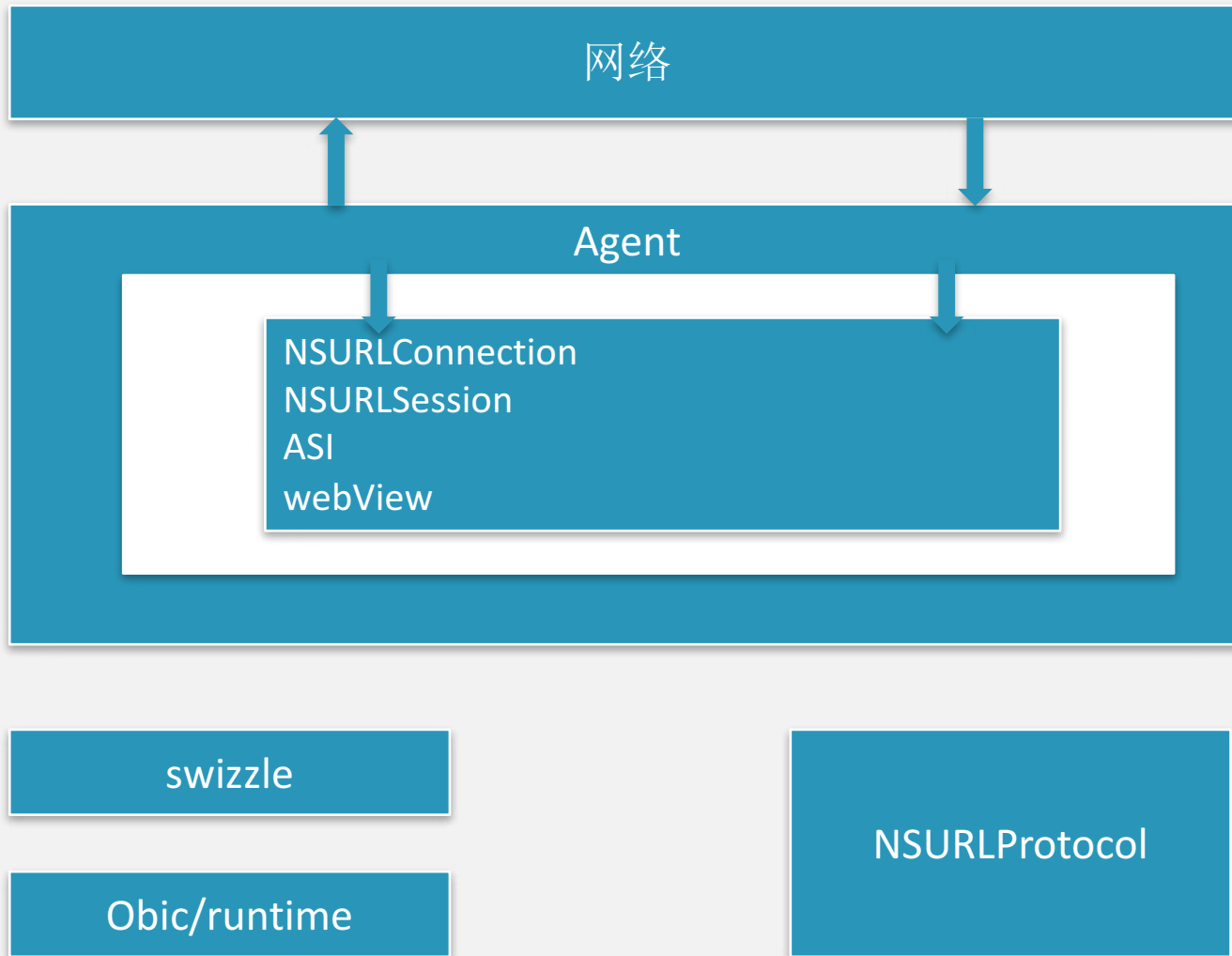


Navigation Timing



Resource Timing

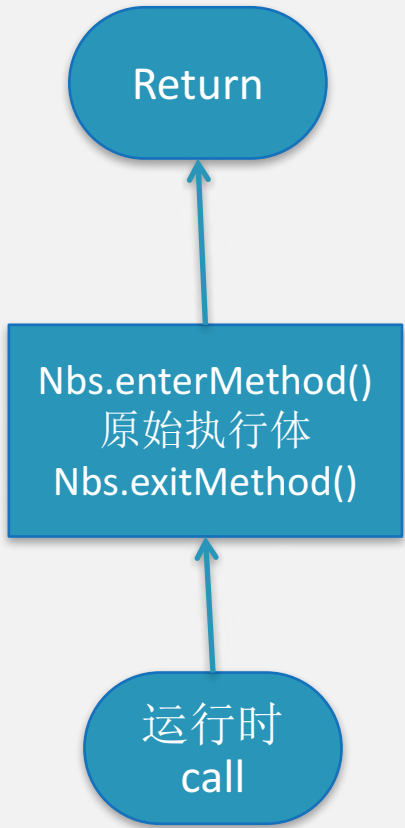




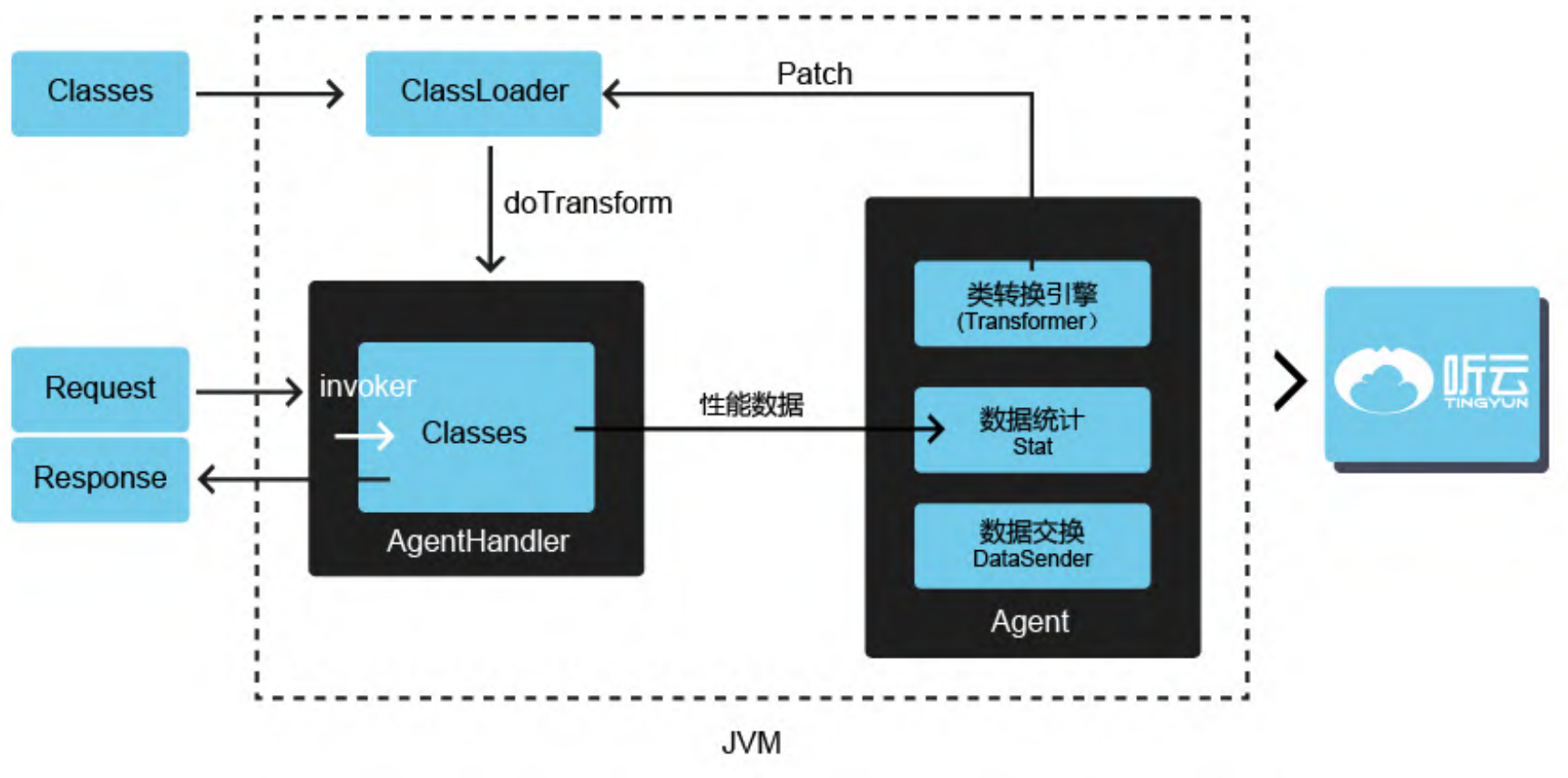


嵌入听云的sdk的代码前

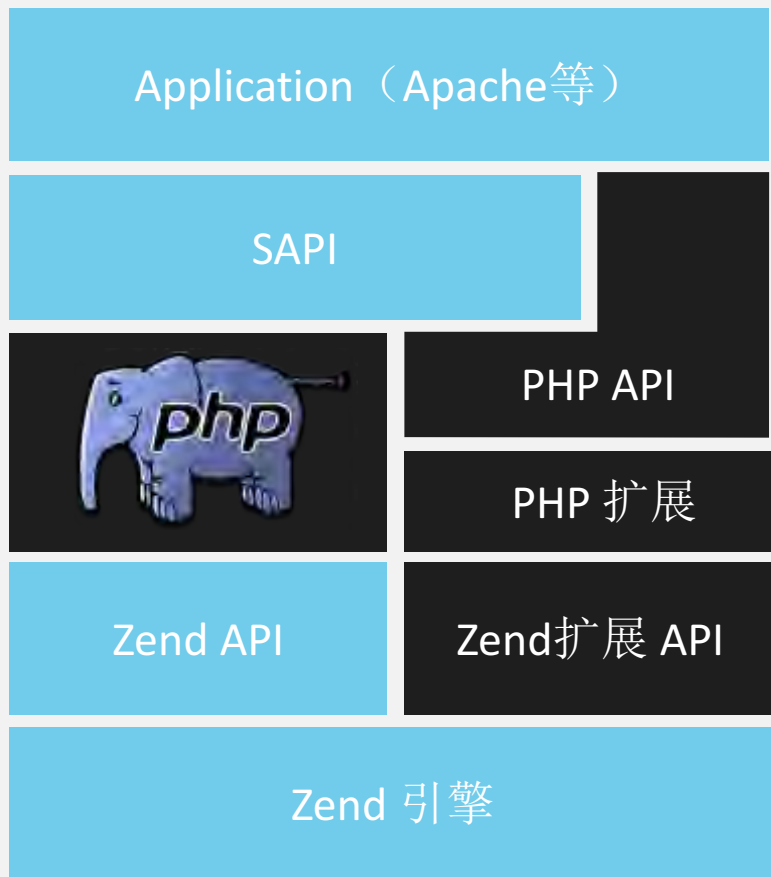
编译时字节码注入, 修改bytecode



嵌入听云的sdk的代码后



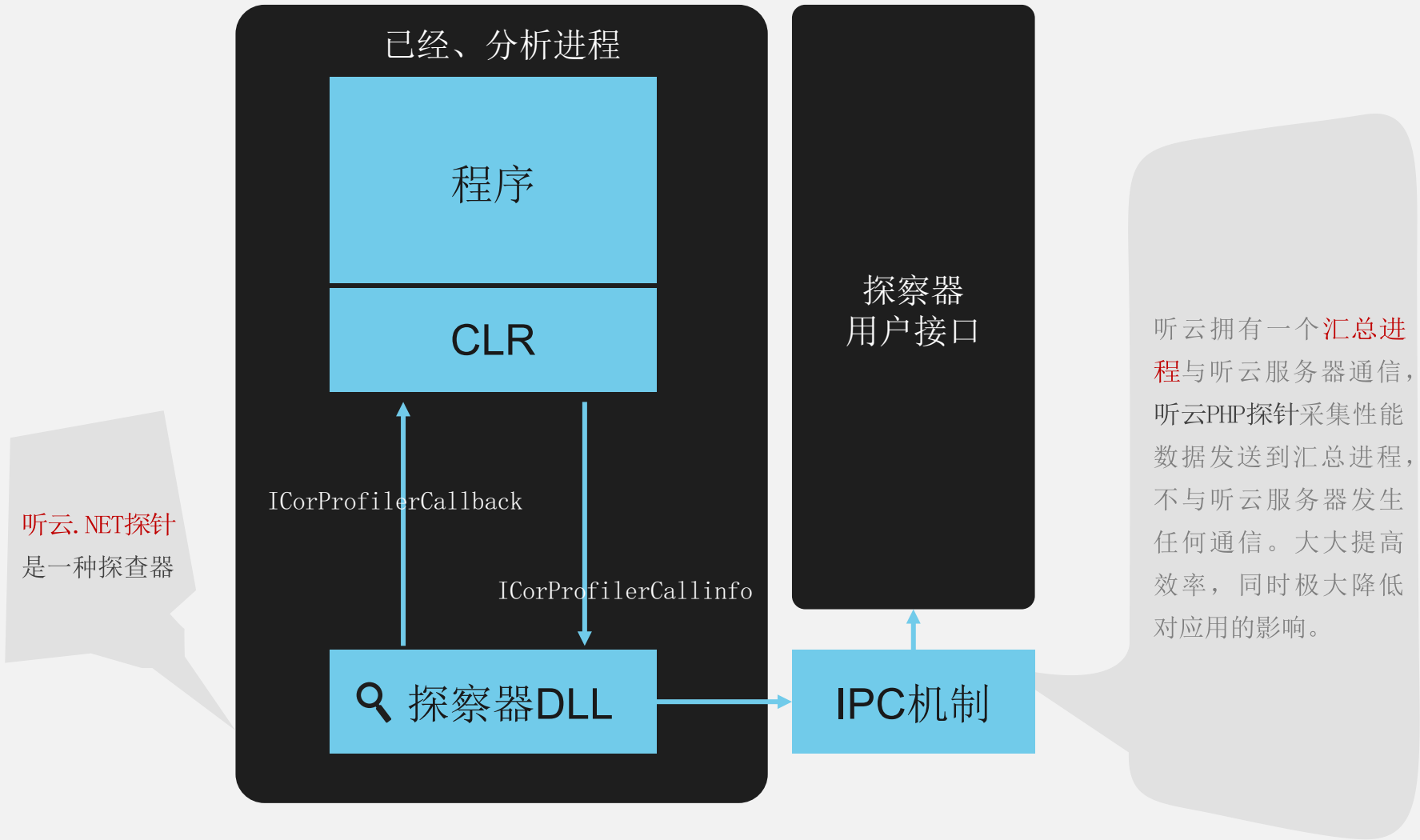
- Java Agent基于JDK 提供的**Instrumentation**机制，在class文件被加载的时候，通过**字节码技术**，动态对Framework、数据库、NoSQL、Web Service、组件等特定方法实施监控，从而获得方法执行时间、数据库调用时间、NoSQL响应时间以及外部服务响应时间；并在这些时间超过一定阈值时，抓取调用堆栈。



听云拥有一个**汇总进程**与听云服务器通信，听云PHP探针采集性能数据发送到汇总进程，不与听云服务器发生任何通信。大大提高效率，同时极大降低对应用的影响。



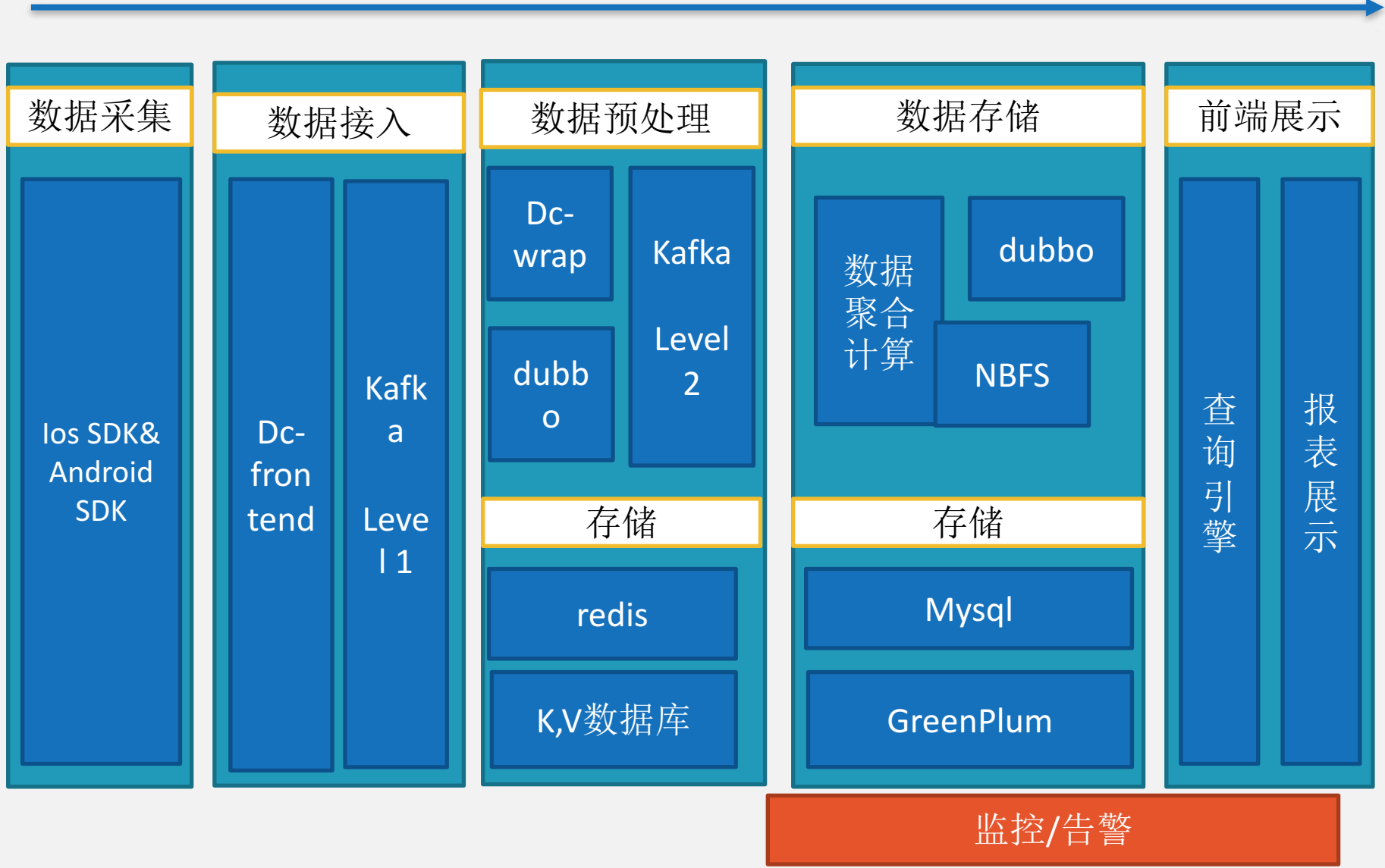
听云的PHP探针也是一个PHP扩展，在Zend引擎循环执行操作码的过程中追踪获得调用各类函数的性能及错误信息。



数据流向



数据流向



- 最终用户体验无法得知
- 接口性能无法掌控
- 私有协议监控
- 用户投诉定位问题无从下手
- 客户投诉问题难以复现
- Docker集群部署，监控变的越来越难

搭建监控体系 - Network

页面监测

- 数据上报接口监控
- 探针下载监控
- 官网重要页面监控

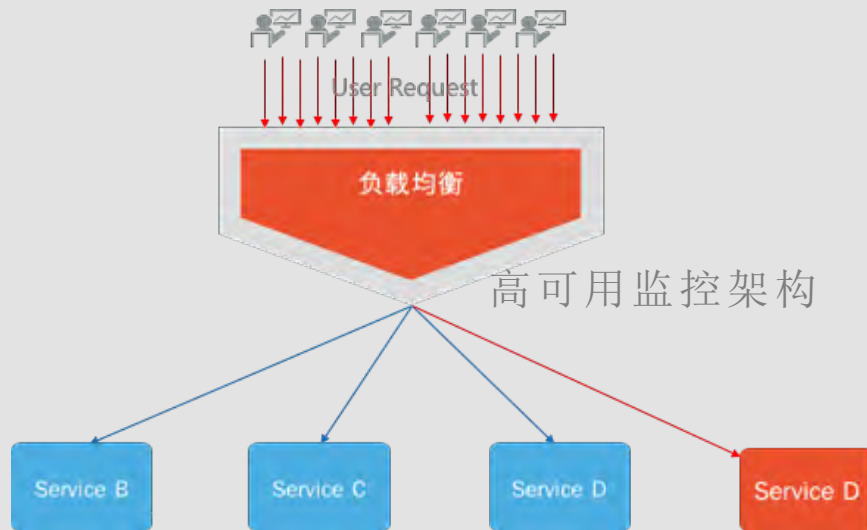


事务监测

- 模拟app上传数据
- 系统登录
- 高可用架构监控

私有协议

- 私有协议监控



The screenshot displays a browser window with the source code of a webpage. A red box highlights the following code snippet:

```
<meta charset="UTF-8"><script src="/browser-web/config/tingyun-rum.js"></script>
```

Below the source code, the browser's developer tools are open, showing the 'Network' tab. The 'XHR' section is expanded, displaying a list of requests. The first request is selected, and its response is shown in the 'JSON' tab:

```
[{"xhr": [{"GET https://report.tingyun.com/browser/user/get?_=1502080256020", 140, 0, 8 更多... }, {"GET https://report.tingyun.com/browser/common.html", 30, 0, 4 更多... }, {"GET https://report.tingyun.com/browser/get?_=1502080256020", 140, 0, 8 更多... }, {"GET https://report.tingyun.com/browser/get?_=1502080256020", 140, 0, 8 更多... }], [{"GET views/common/common.html", 92, 0, 4 更多... }, {"GET views/overviewheader/overviewheader.html", 30, 0, 4 更多... }, {"GET views/maincontainer/main_tab.html", 52, 0, 4 更多... }, {"GET views/overview/browser_overview.html", 65, 0, 4 更多... }, {"GET ./assets/data/i18n/message_en.json?v=15", 370, 0, 4 更多... }]
```

Report


```
1 //
2 // main.m
3 // BaseProject
4 //
5 // Created by 孟瑞瑞 on 15/10/21.
6 // Copyright © 2015年 TingYun. All rights reserved.
7 //
8
9 #import <UIKit/UIKit.h>
10 #import "AppDelegate.h"
11
12 int main(int argc, char * argv[]) {
13     @autoreleasepool {
14
15         [NBSAppAgent startWithAppID:@"87db466644de4f58bdd6fb6c70a79983"];
16         return UIApplicationMain(argc, argv, nil, NSStringFromClass
17             ([[AppDelegate class]]));
18     }
19 }
20
```

getName=&targetType=&timePeriod=2&timeType=2&version=4: {
 "Accept-Language" = "zh-Hans-US;q=1, en;q=0.9";
 "Content-Type" = "application/x-www-form-urlencoded;charset=utf8";
 "User-Agent" = "ZhangShangTingYun/4.0 (iPhone; iOS 10.2; Scale/3.00)";
 "X-Accept-Language" = "zh-cn";
} (null)
2017-08-07 11:43:53.073 ZhangShangTingYun[6658:6232168] POST 'https://
reportalpha1.tingyun.com/alert/alert/0/json/clientevent/list?order?
ty_auth_token=91j8zkg80Fus5k2u98uty8ciq26ikendLine=2017-08-07%2011%3A44%3A
53&eventLevel=0&eventStatus=1&eventType=1&metricId=0&operate=1&readStatus=0&tar

```
FROM registry.tingyun.com/common/centos7-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
CMD ["/run.sh"]
```

dockerfile

```
FROM registry.tingyun.com/app/report_app:alpha1.369
MAINTAINER netop "netop@tingyun.com"
ENV REFRESHED_AT 2016-03-14

ADD report_app.jar /opt/report_app.jar
RUN java -jar /opt/report_app.jar

ENV TINGYUN_APP_NAME="report_app"
ENV TINGYUN_LICENSE_KEY=f451bae910b4c3f4473e0c83d2113742
ENV JAVA_OPTS=""

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

EXPOSE 8080
CMD ["/run.sh"]
```

kubernetes

```
#!/bin/bash

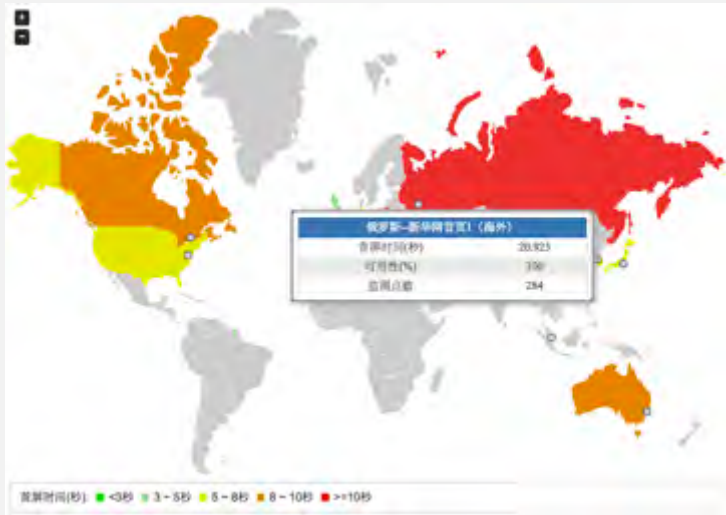
TINGYUN_AGENT_PATH="/opt/tingyun_agent/tingyun/tingyun.properties"
TOMCAT_CONF_PATH="/opt/apache-tomcat/conf/server.xml"

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

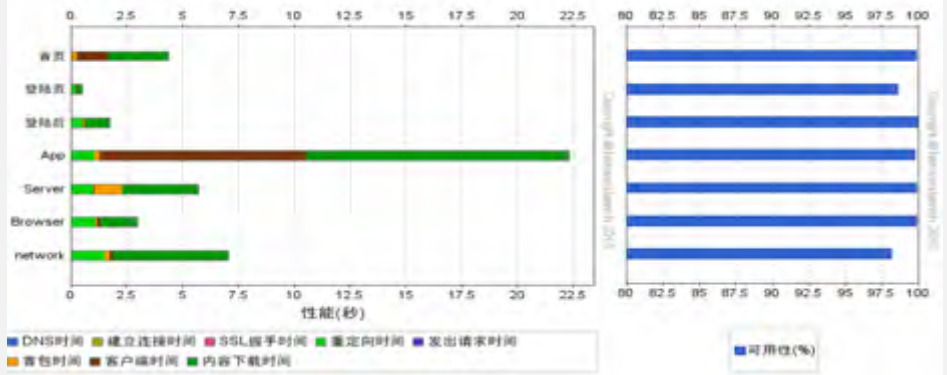
sed -i "s%CONNECTOR%$CONNECTOR%" $TOMCAT_CONF_PATH
sed -i "s%CONTEXT%$CONTEXT%" $TOMCAT_CONF_PATH

catalina.sh run
```

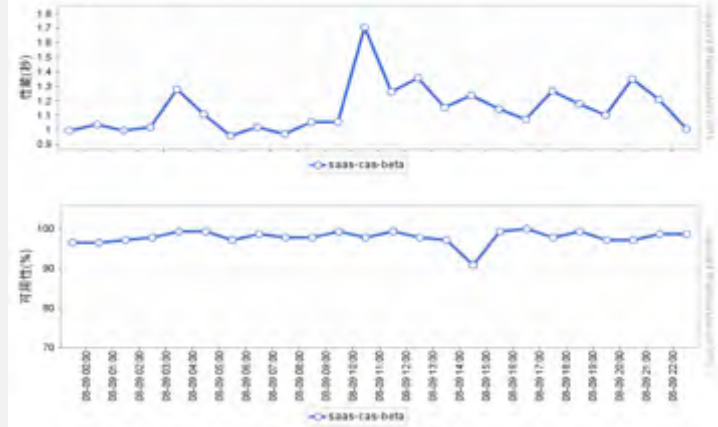
Run.sh



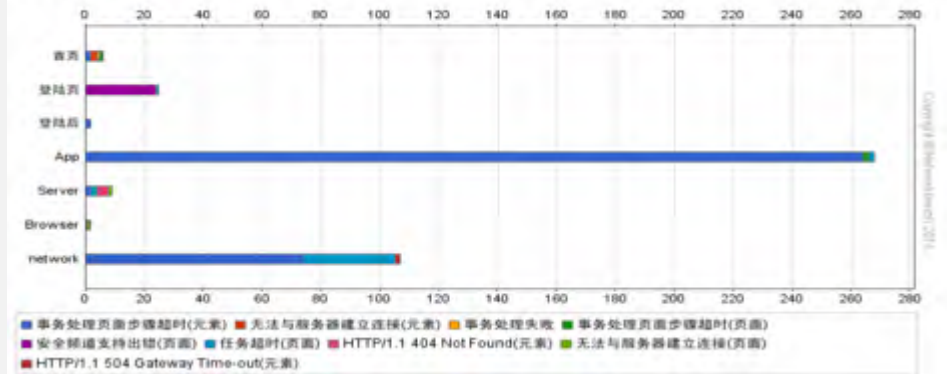
汇总概况图(2017年08月07日 20:30 - 2017年08月08日 20:30)



性能历史曲线图(2017年08月08日 23:30 - 2017年08月09日 23:30)



错误概况图(2017年08月07日 20:30 - 2017年08月08日 20:30)



总览 域名

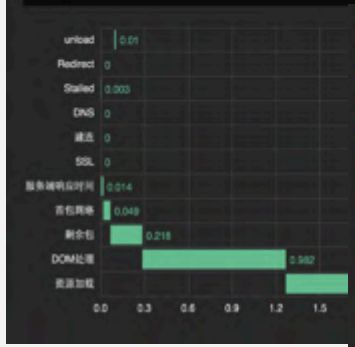
URL	PV(次)	完全加载(s)	白屏(s)	首屏(s)	可交互(s)	JS错误率(%)	慢页面占比(%)	慢页面次数(次)
https://report.tingyun.com/server/application/*/...	76	2.109	0.960	0.961	0.832	0	11.84	9
https://report.tingyun.com/server/overview/ap...	55	2.246	1.373	1.374	1.216	0	27.27	15

完全加载 **8.507s** 白屏 **3.263s** 首屏 **3.263s** 可交互 **1.257s** 自定义加载 **0ms**

用户IP: 111.202.163.114 浏览器: Firefox 54 加载速度: 3.04KB/S 操作系统: Windows 7

地域: 中国 北京 北京市 运营商: 中国联通 分辨率: 1920*1080

UA: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:54.0) Gecko/20100101 Firefox/54.0



瀑布图

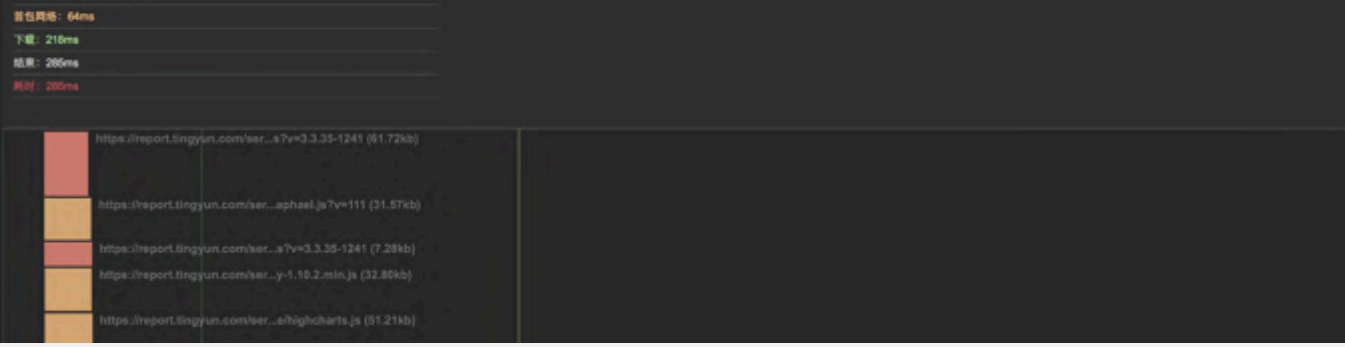
白屏 **3.263s** 首屏 **3.263s** 可交互 **1.257s** 自定义加载 **0ms**

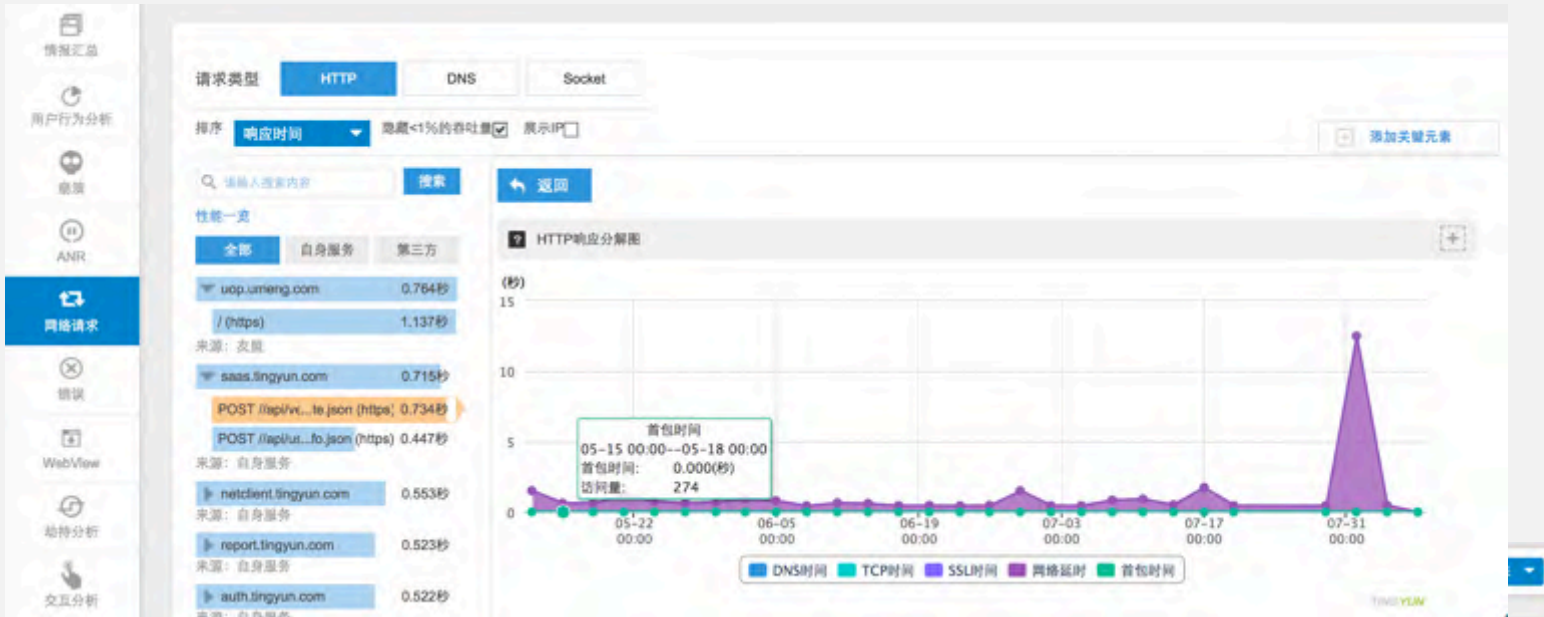
1#

<https://report.tingyun.com/server/overview/application>

开始: 0ms
Stalled: 3ms
DNS: 0ms
连接: 0ms
SSL: 0ms
服务器响应时间: 14ms
首包网络: 64ms
下载: 216ms
结束: 265ms
耗时: 265ms

状态码: 200
页面URL: <https://report.tingyun.com/server/overview/application>
Request: 64ms
Response: 216ms
元素类型: html







事务一选

Q 输入关键字搜索

搜索

排序: 耗时百分比 响应时间 吞吐量 Apex 错误率

性能分析 错误

事务性能分解堆叠图

事务分解表格

代码段	性能分类	耗时百分比(%)	调用次数	平均响应时间(ms)
com.tingyun.report.server.application.controller.ApplicationController/downloadZip	Java	97.444	1	10064
http://download.networkbench.com/*.zip	External	2.498	1	258
Others	Java	0.058	21	0
nl_u_license/SELECT	Database MySQL	0	1	0

添加关键事务

- 应用工具
- 应用设计
- 事务
- 数据库
- NoSQL
- CDN
- 外部应用
- MQ
- 自定义任务
- 错误率
- 线程剖析
- JVM
- 应用环境
- 报警



总结

01



听云业务

- 听云NETWORK
- 听云BROWSER
- 听云APP
- 听云SERVER

02



数据采集

- PC 节点
- JS AGENT
- SDK
- AGENT

03



后端架构

- NETWORK 数据中心
- BROWSER 数据中心
- APP 数据中心
- SERVER 数据中心

04



搭建监控

- NETWORK (页面监控、事务监控、私有协议)
- BROWSER(network报表、browser报表、app报表、server报表)
- APP(掌上听云监控)
- SERVER(后端架构监控)

05



性能可视化

- NETWORK
- BROWSER
- APP
- SERVER

THANK YOU

