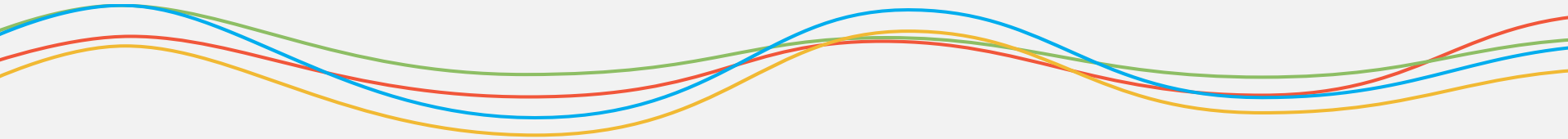
APMCon

网易APM hook方案探索

网易 郑文



自我介绍

APMCon

2013~至今

网易杭研院-前端技术部

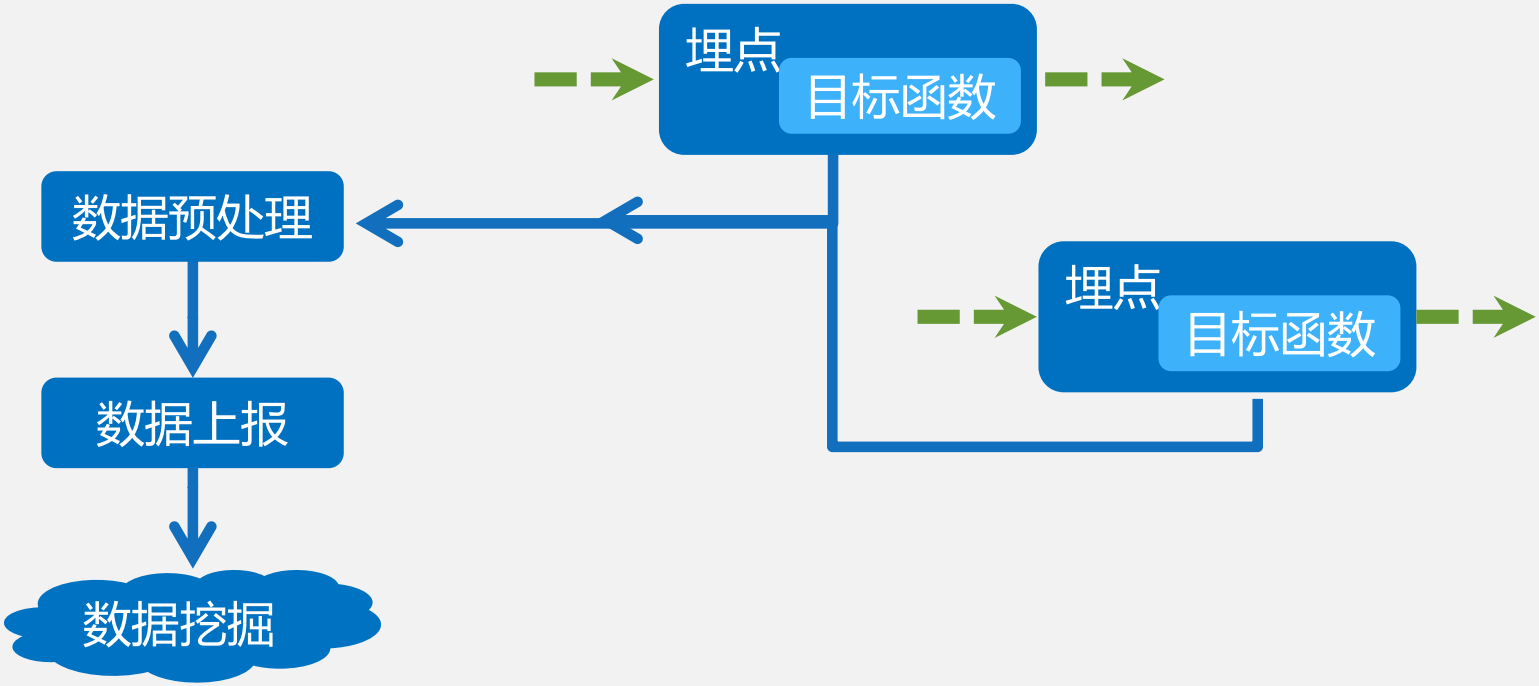
2010~2013

阿里云OS-虚拟机组

目录

APMCon

- 背景&选型
- Ant构建的hook方案
- Gradle构建的hook方案
- 我们的工作




```
@Override
protected void onCreate(Bundle savedInstanceState) {
    ActivityInstrument.onActivityCreating(this);
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    ActivityInstrument.onActivityCreated(this);
}
```

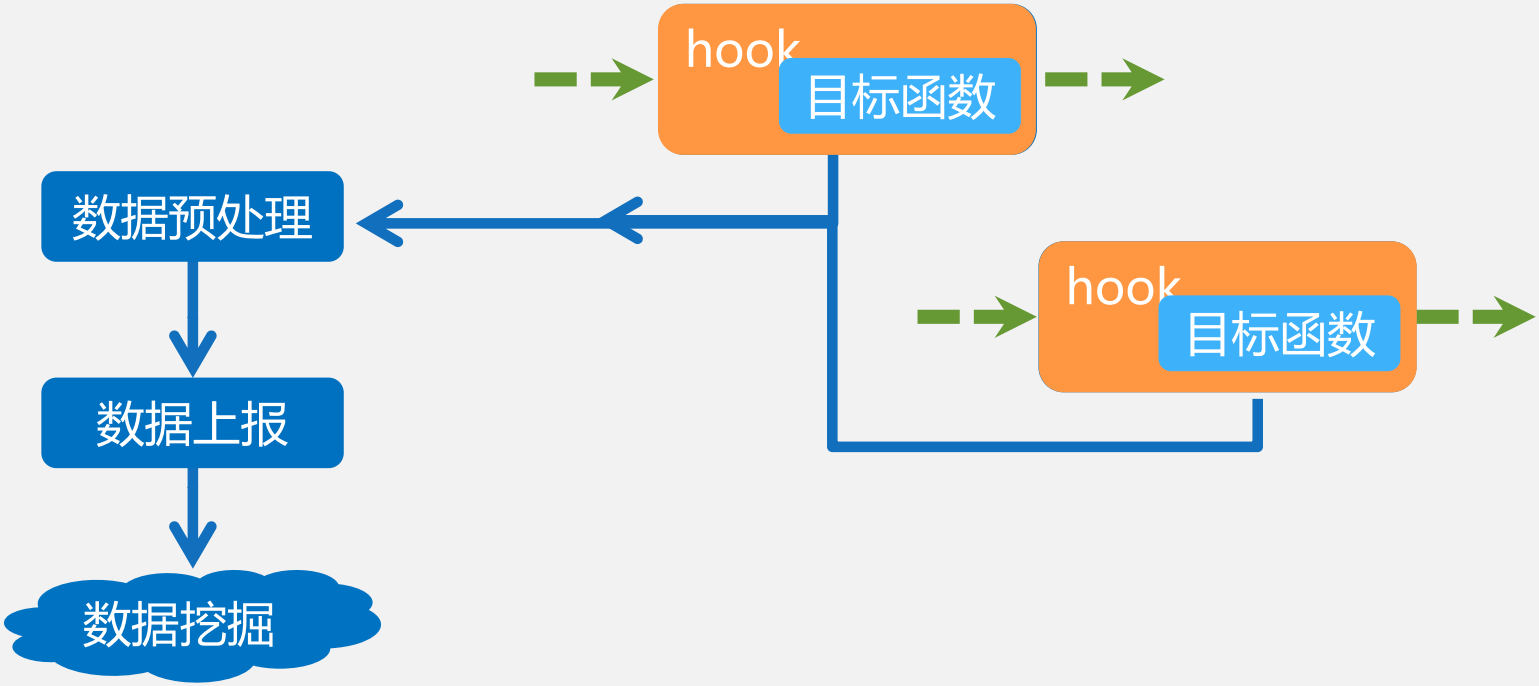
埋点技术的限制

- 接入成本
  - 产品基础框架选型不一
  - 各产品重复的功能埋点
- 源码的限制
  - 封闭的第三方服务

目标

- 无侵入，接入成本小
- 摆脱源码的限制
- 细粒度的监控

APM流程




```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
}
```

```
DexposedBridge.findAndHookMethod(MainActivity.class, "onCreate",
        Bundle.class, new XC_MethodHook() {
    @Override
    protected void beforeHookedMethod(MethodHookParam param)
            throws Throwable {
        MainActivity instance = (MainActivity) param.thisObject;
        ActivityInstrument.onActivityCreating(instance);
    }

    @Override
    protected void afterHookedMethod(MethodHookParam param)
            throws Throwable {
        MainActivity instance = (MainActivity) param.thisObject;
        ActivityInstrument.onActivityCreating(instance);
    }
});
```

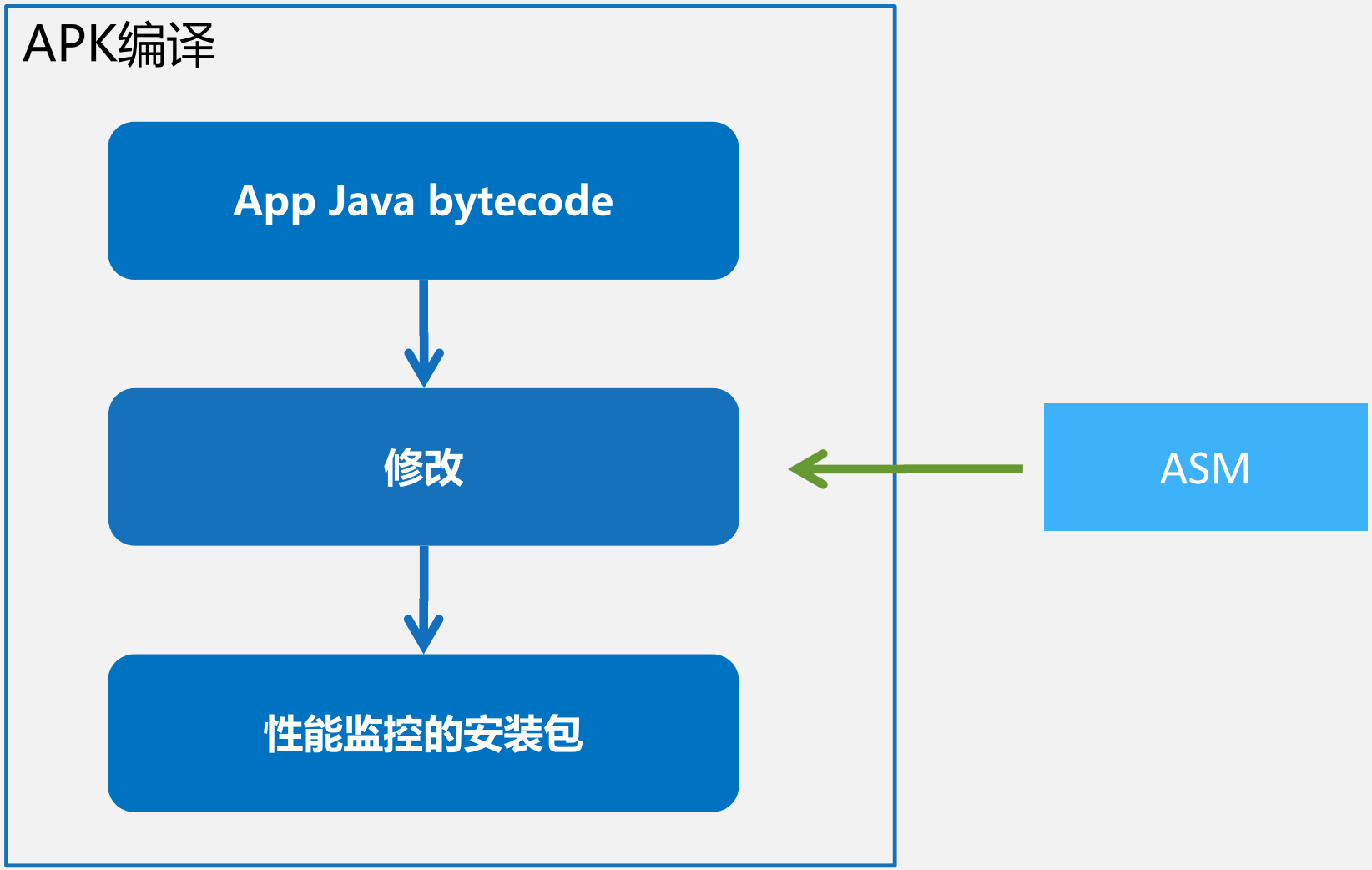
- 动态埋点
- 系统API的hook
- 虚拟机版本兼容

编译期Hook

APMCon

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    ActivityInstrument.onActivityCreating(this);
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    ActivityInstrument.onActivityCreated(this);
}
```

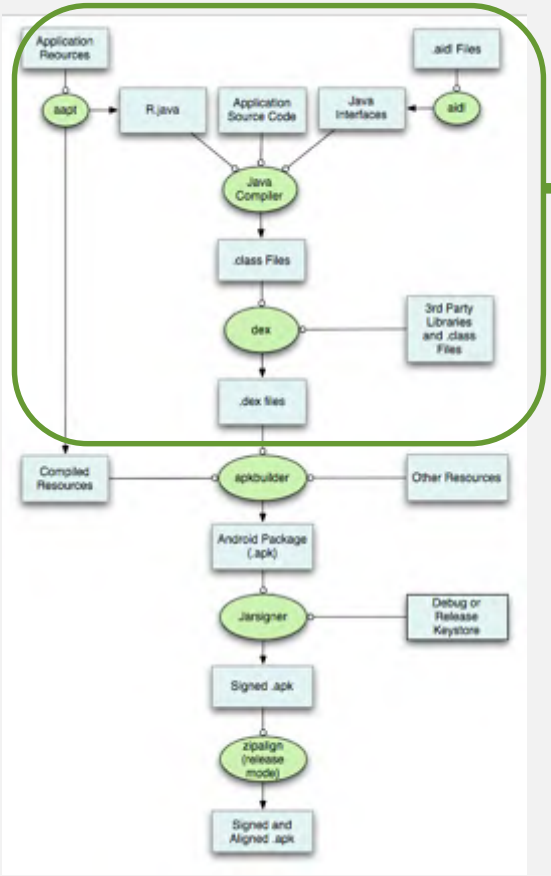
```
aload_0
invokestatic  #2  // Method com/netease/napm/ActivityInstrument.onActivityCreating:(Landroid/app/Activity;)V
aload_0
aload_1
invokespecial #3  // Method android/support/v7/app/AppCompatActivity.onCreate:(Landroid/os/Bundle;)V
aload_0
ldc           #5  // int 2130968602
invokevirtual #6  // Method setContentView:(I)V
aload_0
invokestatic  #7  // Method com/netease/napm/ActivityInstrument.onActivityCreated:(Landroid/app/Activity;)V
return
```

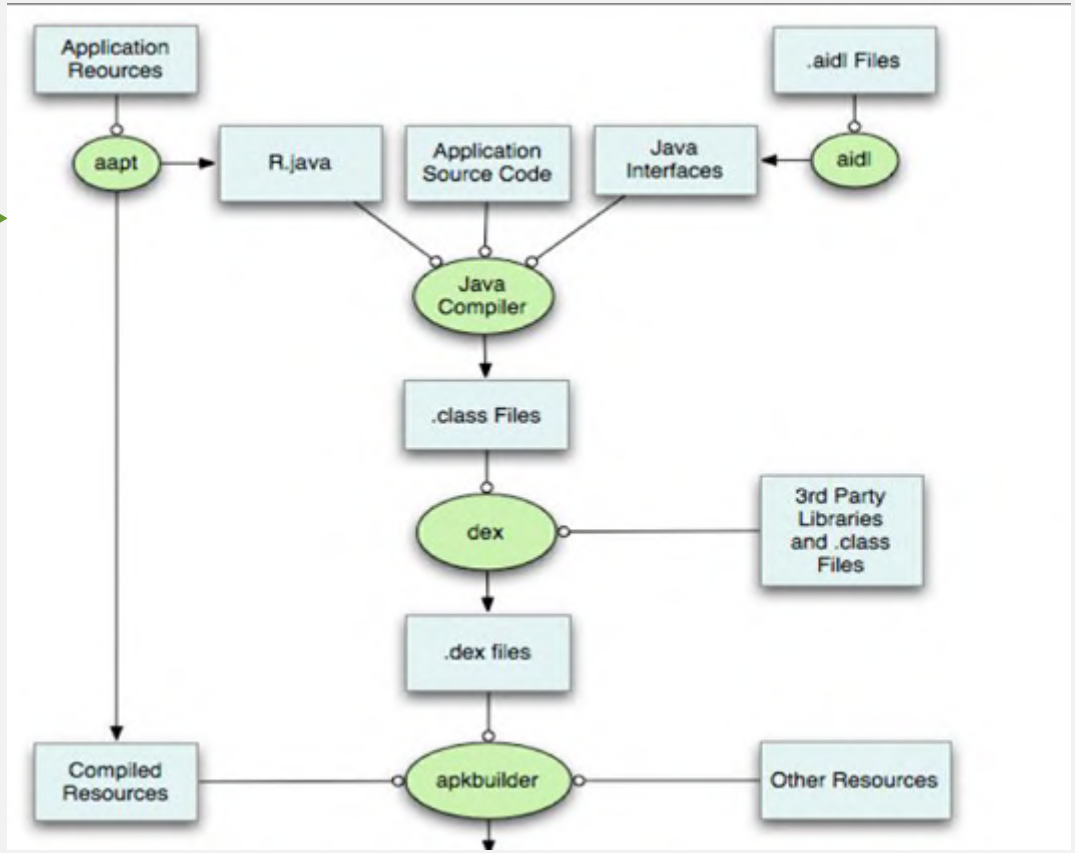

APK编译
App Java bytecode
修改
性能监控的安装包
ASM

- 预先埋点
- hook业务代码
- 兼容性佳
- 运行期轻量

- Ant
- Gradle

构建工具之Ant


Application Reources
.aid Files
aapt
R.java
Application Source Code
Java Interfaces
aidl
Java Compiler
.class Files
dex
3rd Party Libraries and .class Files
.dex files
Compiled Resources
apkbuilder
Other Resources
Android Package (.apk)
Jarsigner
Debug or Release Keystore
Signed .apk
zipalign (release mode)
Signed and Aligned .apk


Application Reources
.aidl Files
aapt
R.java
Application Source Code
Java Interfaces
aidl
Java Compiler
.class Files
dex
3rd Party Libraries and .class Files
.dex files
Compiled Resources
apkbuilder
Other Resources

```
/**
 * Processes one classfile.
 * @param name  name of the file,
 * @param bytes contents of the file
 * @return whether processing was successful
 */
private static boolean processClass(String name, byte[] bytes) {
    if (! args.coreLibrary) {
        checkClassName(name);
    }
    try {
        new DirectClassFileConsumer(name, bytes, null).call(
                new ClassParserTask(name, bytes).call());
    } catch(Exception ex) {
        throw new RuntimeException("Exception parsing classes", ex);
    }
    return true;
}
```

hook Main.processClass ?

- Agent是class加载之前的拦截器
  - 静态Agent：-javaagent:/path/agent.jar
  - 动态Agent:：VirtualMachine.loadAgent
- 修改内存中的字节码

Agent and Instrumention

```
public class Agent {
    public static void premain(String agentArgs,
                               Instrumentation instumentation) {
        instumentation.addTransformer(new DxClassTransform());
    }
}
```

```
public class DxClassTransform implements ClassFileTransformer {
    @Override
    public byte[] transform(ClassLoader loader, String className,
                            Class<?> classBeingRedefined,
                            ProtectionDomain protectionDomain,
                            byte[] dxByteCode) throws IllegalClassFormatException {
        ClassReader cr = new ClassReader(dxByteCode);
        ClassWriter cw = new ClassWriter(ClassWriter.COMPUTE_MAXS);
        //TODO transform dxByteCode,hook Main.processClass
        ClassVisitor classVisitor = new DxTransformVisitor(cw);
        cr.accept(classVisitor,0);
        return cw.toByteArray();
    }
}
```

Agent and Instrumention

APMCon

```
private static boolean processClass(String name, byte[] bytes) {
    byte[]  newBytes = ClassTransformer.transformClassBytes(bytes);
    if (! args.coreLibrary) {
        checkClassName(name);
    }
    try {
        new DirectClassFileConsumer(name, newBytes, null).call(
                new ClassParserTask(name, newBytes).call());
```

```
public  static byte[] transformClassBytes(byte[] bytes) {
    ClassReader cr;
    ClassWriter cw;
    ClassVisitor cv = null;
    try {
        cr = new ClassReader(bytes);
        cw = new ClassWriter(cr, 1);
        //....
        if (className.startsWith("android/support/")) {
            cv = new ActivityClassVisitor(cv, context,log);
        } else {
            cv = new OKHttpAccessClassVistor(cv, context,log);
            cv = new ActivityClassVisitor(cv, context,log);
            cv = new ApacheHttpClassVistor(cv, context, log);
```

0. hook ProcessBuilder.start添加javaagent参数
1. hook Main.processClass修改参数
2. 应用bytecode植入监控代码

```
export ANT_OPTS="-javaagent:/path/agent.jar"
```

```
dx -javaagent:/path/agent.jar
```

APMCon

构建工具之Gradle

- 任务依赖
- Gradle守护进程
- Transform API

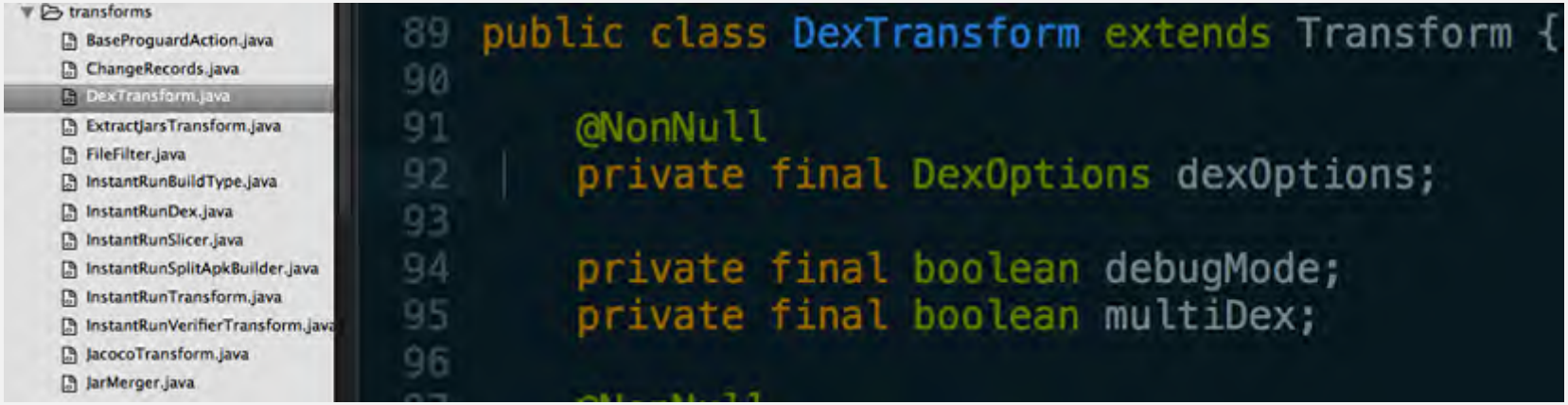
- Starting with 1.5.0-beta1
- The dex class is gone. You cannot access it anymore through the variant API

动态挂载Agent

```
variant.dex.dependsOn agentInstrumentTask
variant.dex.finalizedBy agentDeinstrumentTask
```

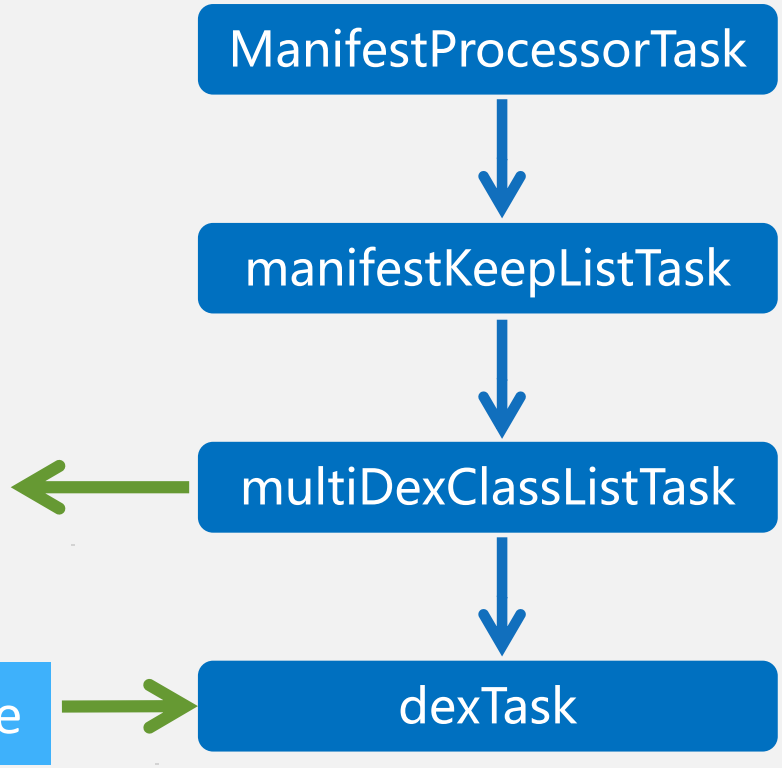
```
class AgentInstrumentTask extends DefaultTask {
    @TaskAction
    def instrument() {
            //...
        try {
            VirtualMachine vm = VirtualMachine.attach(pid);
            vm.loadAgent(jarFilePath, agentArgs);
            vm.detach();
        } catch (Exception e) {
            throw new RuntimeException(e);
        }
        //...
    }
}
```

Transform is not final




```
project.tasks.findByName(
        "transformClassesWithDexFor${variant.name.capitalize()}"
)
```

- MultiDex：
  NoClassDefFoundError
  - maindexlist.txt
- dx in-process
- 多插件冲突



- APM Tranform先于系统Transform执行
  - 代码混淆
  - 代码精简
  - Instant Run
- 兼容Gradle 1.5，2.0，2.1

我们的工作

- 界面启动
- File
- SQLite
- 内存
- 自定义采集

- HTTP
  - 状态码
  - 流量
  - 请求响应时间
  - Header
- WebView
  - 页面加载

org.apache.http.HttpRequest

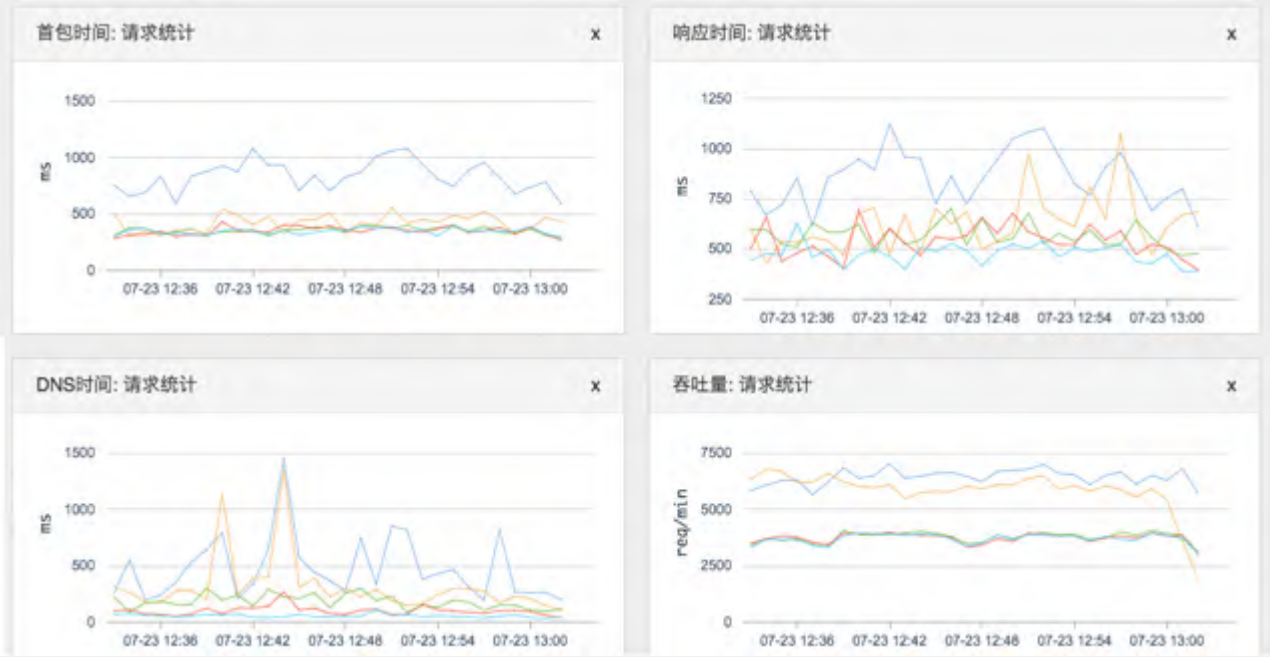
-> com.netease.apache.http.HttpRequest

- Time To First Byte
- Connect time/Read Time
- DNS
- ....

其他在做的

APMCon

- 异常流量报警熔断
- 网络诊断
- HttpDNS
- 私用通道数据上报
- WebCache
- …


首包时间: 请求统计
x
1500
1000
500
0
ms
07-23 12:36
07-23 12:42
07-23 12:48
07-23 12:54
07-23 13:00
响应时间: 请求统计
x
1250
1000
750
500
250
ms
07-23 12:36
07-23 12:42
07-23 12:48
07-23 12:54
07-23 13:00
DNS时间: 请求统计
x
1500
1000
500
0
ms
07-23 12:36
07-23 12:42
07-23 12:48
07-23 12:54
07-23 13:00
吞吐量: 请求统计
x
7500
5000
2500
0
req/min
07-23 12:36
07-23 12:42
07-23 12:48
07-23 12:54
07-23 13:00


LOFTER
網易
NEWS
1元
夺宝

THANK YOU