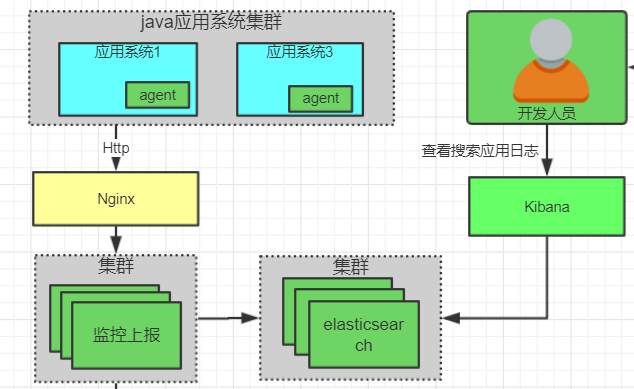
# Service 与Control 插桩埋点实现

## 一、项目架构介绍



## 二、采集端执行流程说明

**需求与目标**

采集指定数据，服务响应能、WEB响应性能、JDBC响应性能

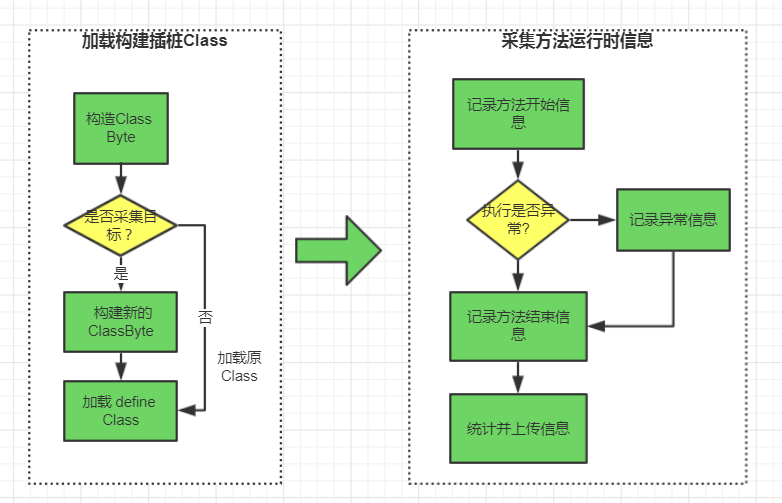
**处理流程**

1、判定谁是采集目标类

2、构建插桩后的Class字节

3、采集方法运行时信息

4、上传运行时信息



**结论：（绝对必须这么去做）**

所有采集器必须要有判断是否监控目标的方法。

所有的采集器必须对我们的Class进行改造，生成插桩之后的字节码。

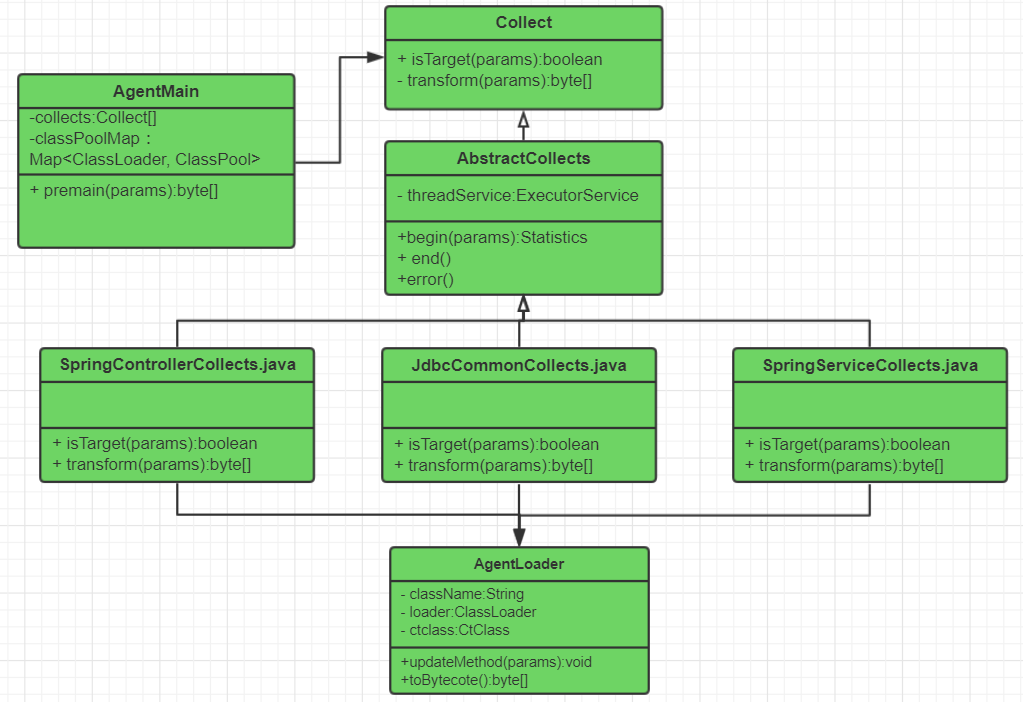
得出一个接口：**Collects**

记录开始信息、结束信息、异常信息、统计上传信息。**（一般情况都会这么去做）**

得出一个抽象类：**AbstracetCollect**

通用的方法：开始信息、结束信息、异常信息、统计上传信息

三、采集端架构UML类图及介绍



**AgentMain:**

*监听器入口方法，所有采集器注册至该对象。由该对象的*transform 来传递改造后的Class byte 至 ClassLoader进行加载。

**Collect:**

*采集器接口，isTarget方法判定指类是否为采集目录，*transform 构建插桩后的Class

**AbstractCollects:**

采集器的通用方法实现：begin 采集方法执行开始信息，error 采集异常信息 ，end 采集方法的结束信息。sendStatisticByHttp 基于Http 上传统计信息。

**AgentLoader:**

采集类修改器：updateMethod 改造指定方法已插入监听代码。toBytecote() 构建修改后的类字节。

## Service 采集

**判定目标类？**

基于XML 的配置如何 判定Service 为采集目标？只能基于配置完成service 服务的判断。

**判定目标方法？**

公共的、非静态、非本地

**监听方法代码构建**

MethodSrcBuild: 开始代码、异常时执行代码、结束时执行代码。

**统计信息传递**

ServiceStatistics

**异常堆栈传递**

sendErrorStackByHttp("", throwable);

## Control 采集

**判定目标类？**

基于@Control判定是否为采集目标。

**判定目标方法?**

屏蔽非公共方法、屏蔽静态方法、屏蔽本地方法、必须带上 RequestMapping 注解

**获取URL地址**

* 类 @RequestMapping 注解获取 value值.
* 方法 @RequestMapping 注解获取value值.
* 以上value值基于正则表达示获取.

a