**深度掌握应⽤性能指标采集⽅法**

1. 项⽬模块设计

2. Service接⼝响应性能采集实践

3. Http响应性能采集实践

**⼀、项⽬模块设计**

概要：

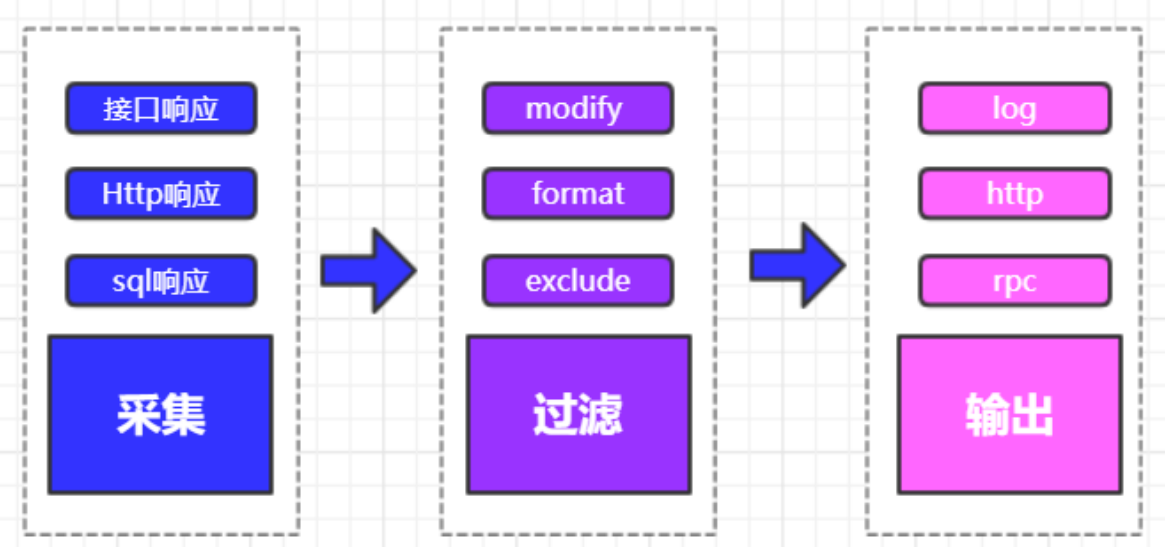
1. 整体设计介绍

2. 采集指标数据模型

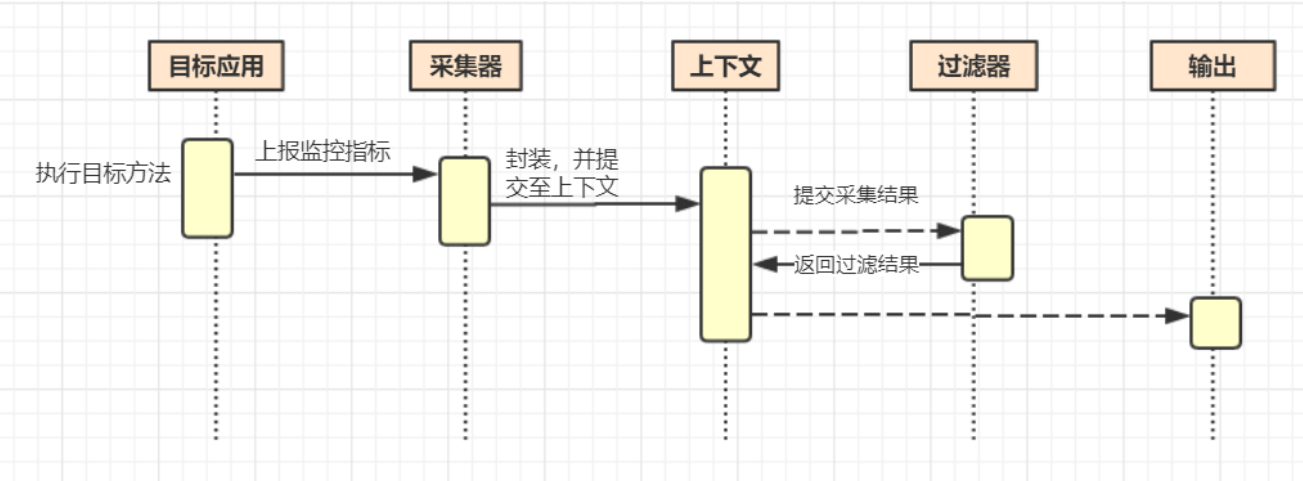
3.具体编码实现讲解

1、整体设计介绍

整体设计图：



时序图：



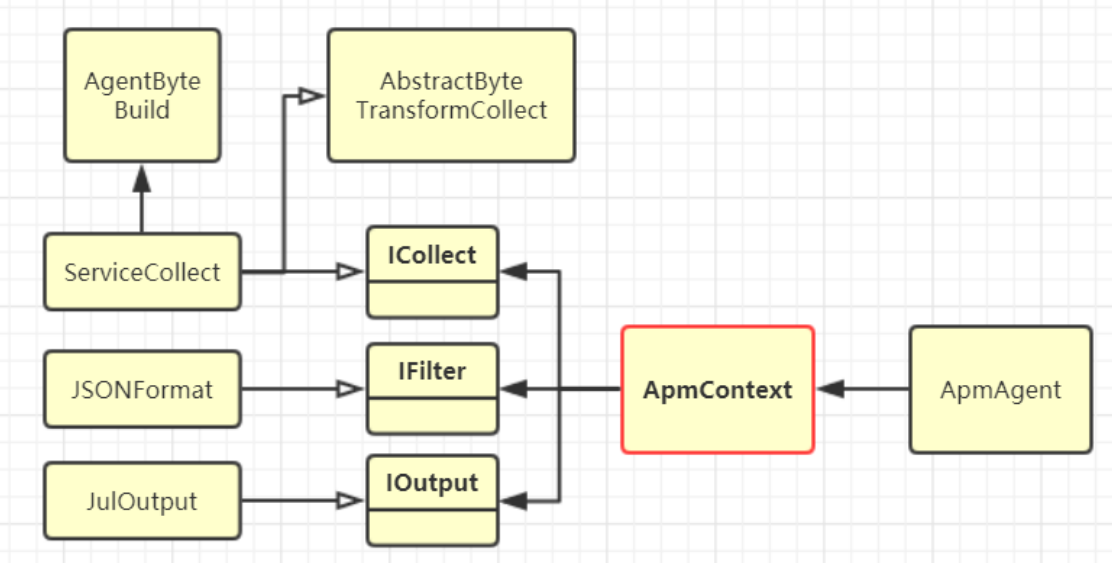
2、采集指标数据模型

基础模型：

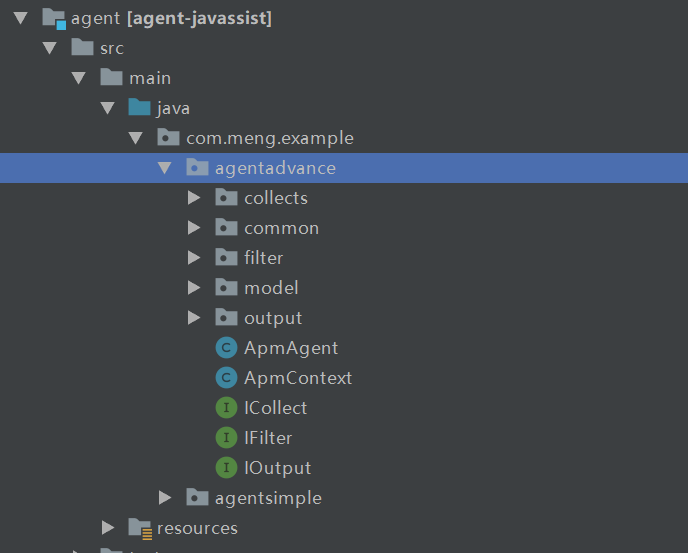


3、具体编码实现讲解

**整UML类图讲解**



**项⽬分包：**



**遇到的难题：**

1. **Json格式如何输出？**

a. 使⽤第三⽅⼯具输出将会污染应⽤ClassPath

b. ⾃定义实现Json串的输出

c. 重新改造第三Json串输出⼯具

2. **⽇志如何输出？**

a. 第三⽅框架影响污染应⽤ClassPath,且需⾃定义配置

b. 使⽤JUL打印⽇志

**⼆、Service接⼝响应性能采集实践**

**概要：**

1. service 采集需求讲解

2. service 采集⽅案3. service 具体编写实现

3. service 具体编写实现

**1、service 采集需求讲解**

需求如下：

1. 接⼝指定时间段内调⽤统计包括：调⽤次数、平均执⾏时间、最快、最慢、成功次数、失

败次数、慢执⾏次数。

2. 慢执⾏上报并报警。

3. 异常执⾏上报并报警。

**2、service 采集⽅案**

1. **采集⽅式：**

a. 字节码插桩

2. **采集⽬标判定：**

a. 基于String“@Service"注解

b. 基于⽤户⾃定义配置include 与 exclude

i. ⽤户需要配置

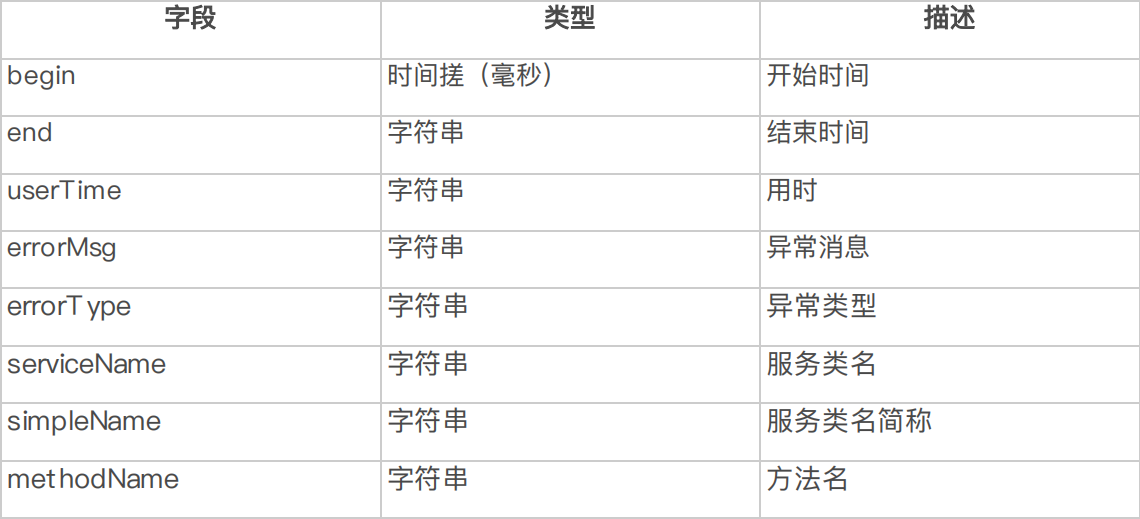
ii. 加⼤了开发成本 com.tuling.\*.\*\*

3. **统计⽅案**

a. 基于最原始数据上报每条数据，在基于ElasticSearch 进⾏统计

1. **service 具体实现**

数据模型



**三、Http响应性能采集实践**

1. Http响应性能监控需求

2. Http响应采集⽅案探讨

1、Http响应性能监控需求

**需求如下：**

1. http 指定时间段内调⽤统计包括：调⽤次数、平均执⾏时间、最快、最慢、成功次数、

失败次数、慢执⾏次数。

2. 慢执⾏上报并报警。

3. 异常执⾏上报并报警。

4. URL分类统计

2、采集⽅案

1. **采集⽅式：**

a. **字节码插桩**

2. **采集⼊⼝：**

a. SpringDispacherServlet .doServer()

b. 基于Spring @control 注解，监控对应⽅法@ requestMapping(url)

c. 采集 httpServlet service()

3. **统计⽅案**

a. 全部上传

b. 30秒上传⼀次

4. URL分类统计 // ⽤例 添加购物⻋、查看商品详情⻚、