软件方案设计说明书

**项目名称：智能代理概要设计**

**文档修订历史**

|  |  |  |
| --- | --- | --- |
| 修订版本 | 修订作者 | 修订备注 |
| 20180713 | 张守年 | 新增 |

## 术语表【可选】

|  |  |
| --- | --- |
| 名称 | 解释 |
| 智能代理 | 根据名单做自动的切换 |
|  |  |

**目录**

[术语表【可选】 1](#_Toc16161)

[1 业务分析【可选】 3](#_Toc17165)

[2 系统框架设计 4](#_Toc20544)

[3 功能设计 5](#_Toc6446)

[3.1 模块1功能设计 5](#_Toc31358)

[3.1.1逻辑流程图： 5](#_Toc14644)

[3.1.2 时序图： 6](#_Toc20561)

[3.1.3 数据存储【可选】 6](#_Toc24489)

[3.1.4依赖关系： 7](#_Toc2121)

[3.2 模块2功能设计 8](#_Toc19552)

[4 埋点设计 8](#_Toc16482)

[5 测试建议 8](#_Toc29949)

[6 兼容性和可扩展性设计 9](#_Toc28926)

[7 安全性设计 9](#_Toc3186)

[8 性能设计【可选】 9](#_Toc13533)

[9 功耗设计【可选】 10](#_Toc4026)

[10 可测试性设计【可选】 10](#_Toc30076)

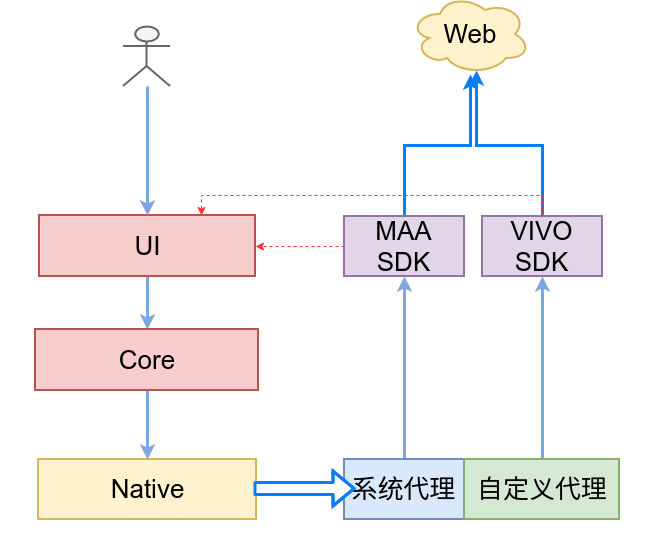
# 业务分析【可选】

按照年初始讨论的计划，将有价值的站点走VIVO加速，价值不大的走MAA加速，只提供加速功能。

# 系统框架设计

当前的结构下面图一所示，MAA和VIVO分别使用了内核的系统代理阶段和自定义阶段，是两条独立的线条，这两条线路是互斥的，而智能代理的需求决定了我们必须将两条线整合在一起，但是又不方便在Native进行整合，主要有如下两个原因：

1. 从UI将用户的相关信息及代理策略传到Native比较麻烦。
2. Native对于一些信息的管理，以及对Web的支持没有Java层友好。
3. 当前VIVO SDK已经在Java层建立一套类似的东西。



图一 当前MAA与VIVO代理的结构

需要的工作从Native到UI分别有：

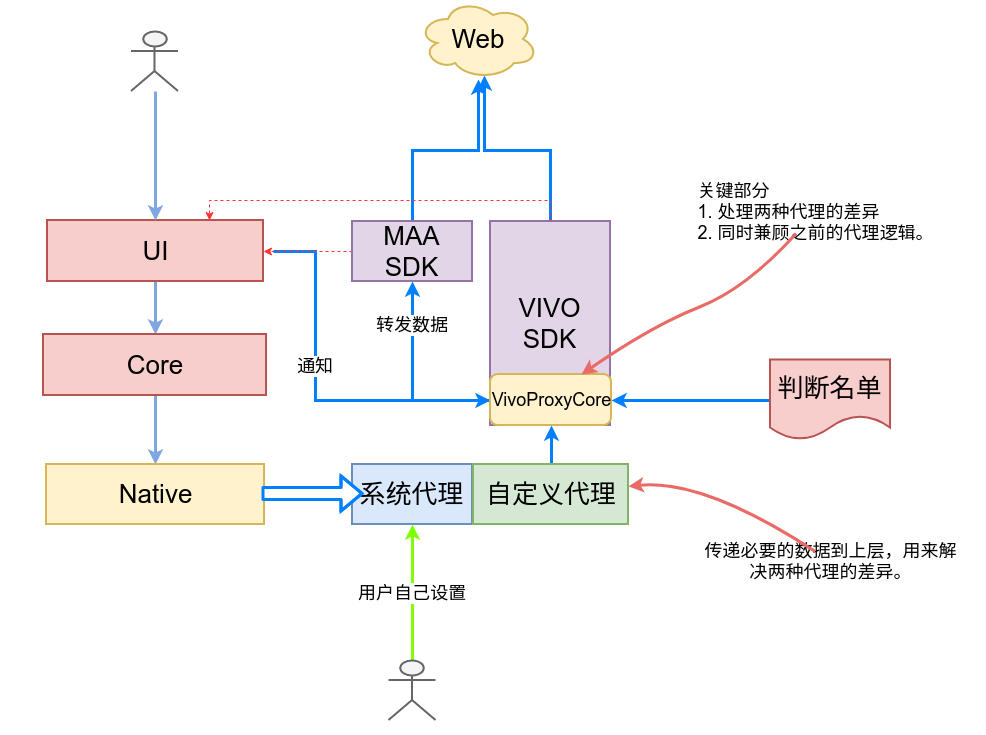
1 Native层要将系统代理阶段和自定义代理阶段收拢，归并成一个，内核的逻辑及MAA之前的适配部分要做简单调整。MAA原来会使用系统代理，会和用户的设置冲突，因此将两种代理全部收拢至我们自定义代理阶段。即，MAA借用原有的VIVO代理通道将数据传上来。

2 既然是借用通道，而两个代理SDK对代理规则的支持是各不一样的，因此会涉及到一些规则的调整，将大部分判断的规则上浮到VivoProxyCore Java层进行处理，按不同的代理处理不同的代理规则。

3 增加了一种代理类型，从UI的ProxyRule到VivoProxyCore中的具体的代理名单数据处理，包括数据的获取使用。

4 增加的逻辑在VivoProxyCore中的适配处理，以及与上层UI的一些信息同步。

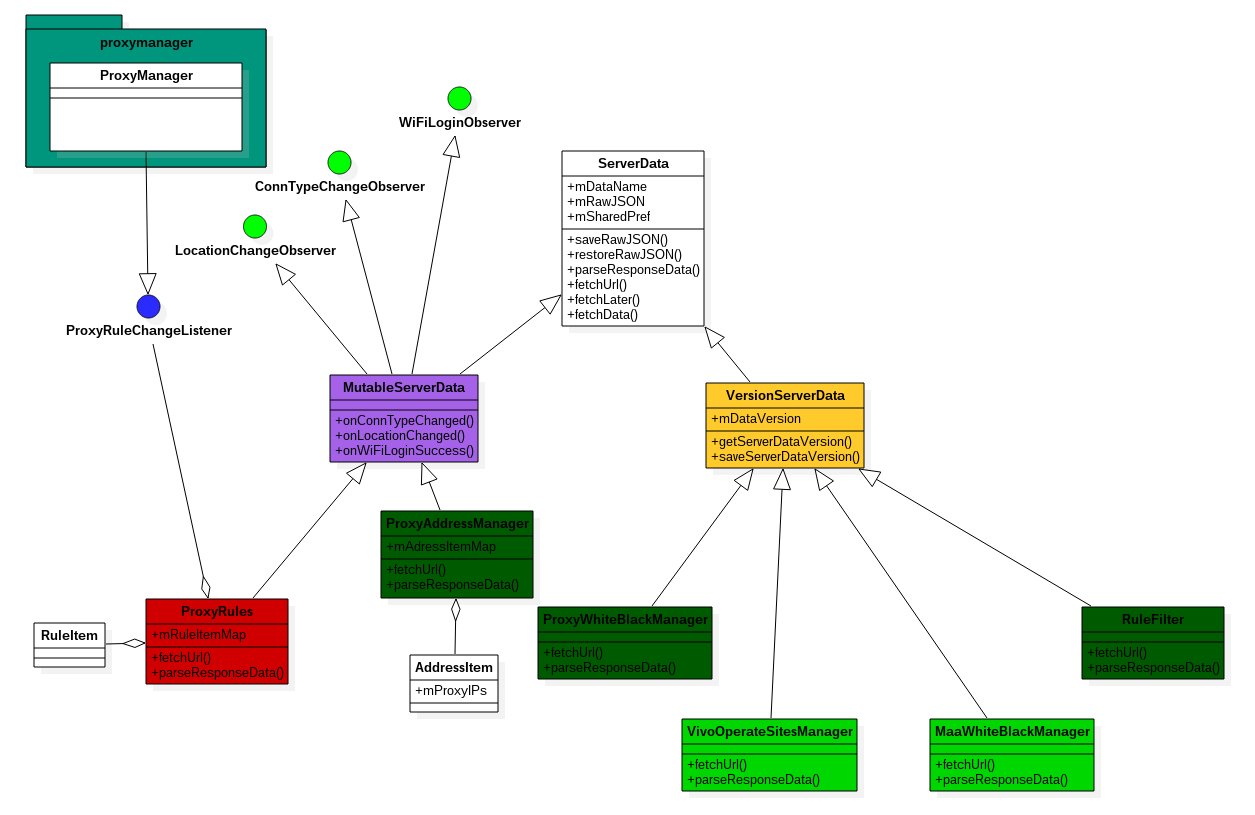
5 代理规则变化，之前是按单个资源进行的，智能代理要求按页面进行，因此内核需要将判断页面及资源类型的两类信息上传到VivoProxyCore层。



图二 智能代理架构改变情况

最终调整后的结构如图2所示。

2.7内核当前的层次继承关系：



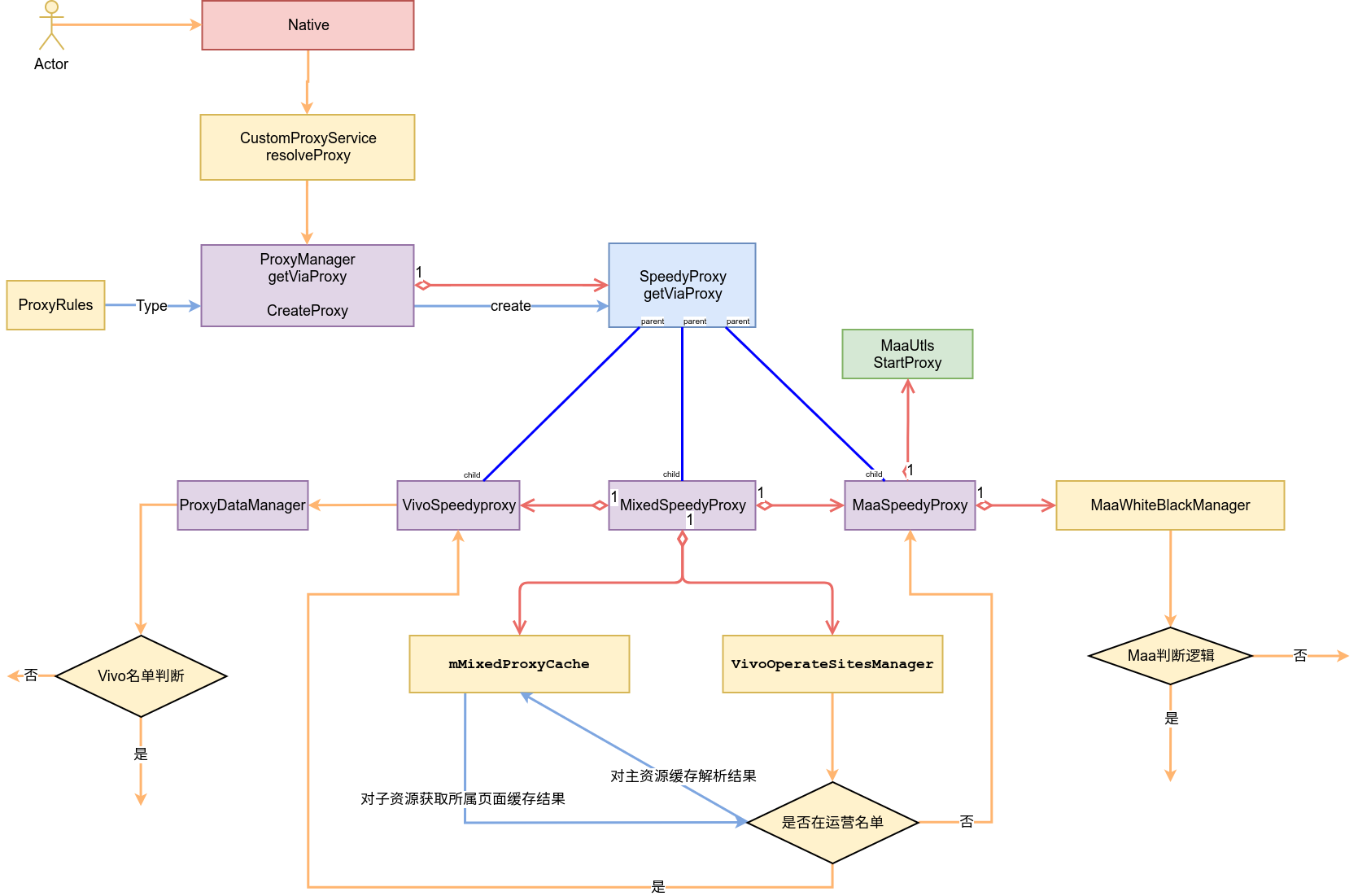
# 功能设计

除此功能外，还有之前外销发版的一些需求合并进来，减少服务器数据的请求次数及一些版本控制逻辑。

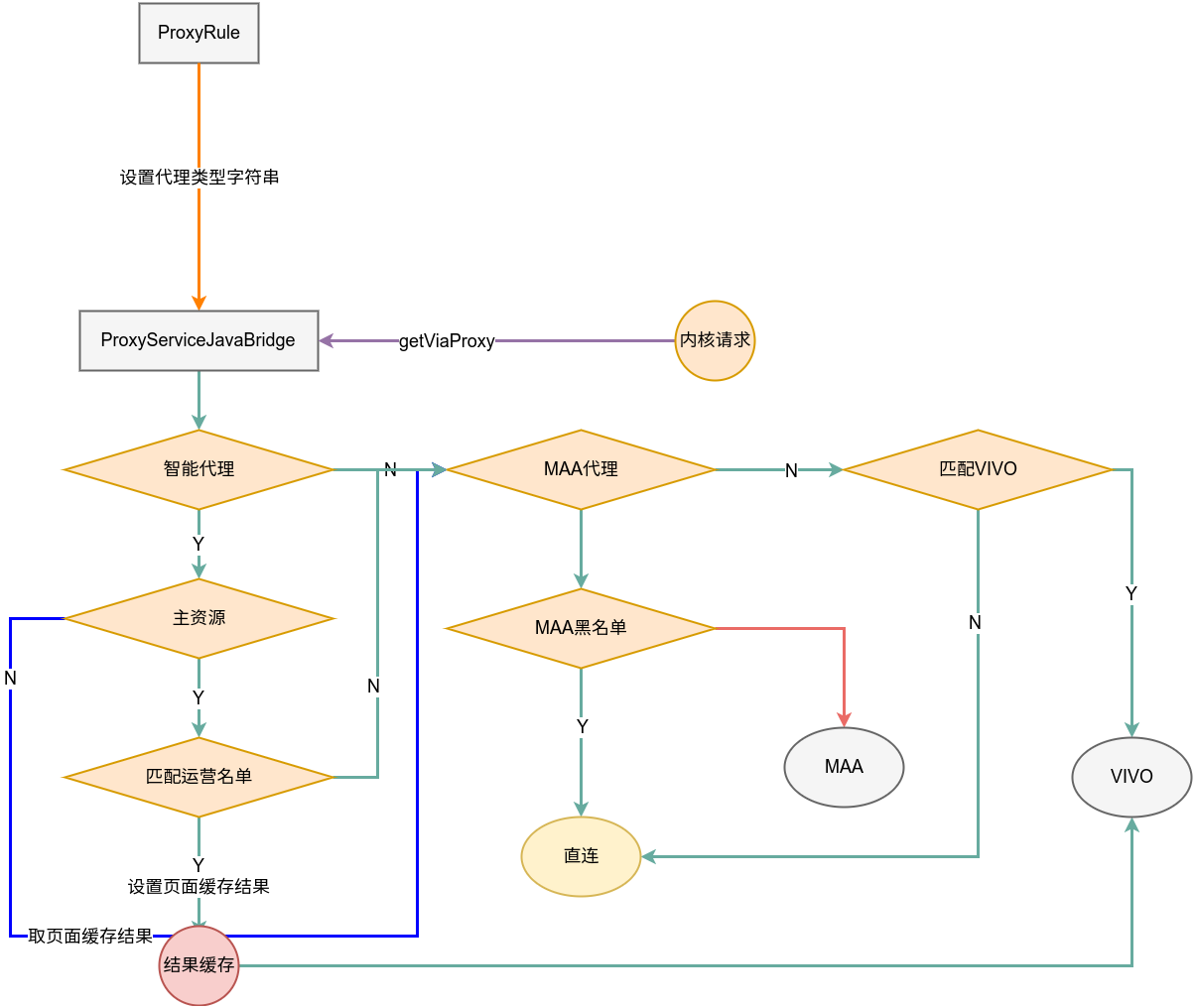
## 模块1功能设计

## 3.1.1逻辑流程图：

在新内核2.7 上代码设计如下：



## 3.1.2 时序图：



## 3.1.3 数据存储【可选】

## 3.1.4依赖关系：

# 埋点设计

# 测试建议

1. 测试Mix智能代理的数据下发是不是正常，/data/data/com.vivo.browser/share\_prefs/proxy\_rule.xml中存储了所有省流相关的配置数据。
2. 测试是否按名单的要求进行了正确的代理情况。通过运营名单是不是正常走了VIVO或MAA加速，页面上的所有资源是不是按主资源的判定结果进行走代理。
3. 代理规则的ProxyRule的下发数据是不是匹配正确的代理类型。
4. 测试Maa黑白名单是不是生效，数据有没有及时更新，包括增加规则，修改规则和删除规则。
5. ProxyRule的版本管理，及对应的四个业务表的更新是不是正常。
6. 分Wifi及APN缓存的数据，在网络切换时有没有进行正确的切换。过期数据有没有如期过期。

# 兼容性和可扩展性设计

# 安全性设计

# 性能设计【可选】

# 功耗设计【可选】

# 可测试性设计【可选】