

基于消息推送的接口调用

策略方案

现状原因：

移动端 ===》 存在 数据缓存

问题：

什么时候更新移动端数据===》更新同步问题

方案：

- 1、类似轮询机制，每次移动端接口调用，查询是否更改。
- 2、异步消息回调机制，有数据更改=====》发消息给移动端
===》移动端调用接口，获取数据&&更新本地（类似BIO 机制）
- 3、数据修改===》推送数据给移动端 && 移动端不需要调用接口（类似AIO机制）

服务器端：

- 1、服务器端的框架 =====》 数据修改 ===》数据落地（存入mysql、redis） && 发消息（消息会通过 IM 推送给移动端）

=====》 纯异步化，有修改的才拉取信息；减少网路交互，IO优化

- 2、服务器业务端（产生消息） =====》 消息服务中心
（传递） ===》移动端（消费消息）

消息的载体：

- 1、IM服务器 or 消息队列等
- 2、长连接 有心跳检测 在数据包上 透传消息 =====》
01010101011001001010101 (位置代表接口，0 代表没有 修改 1 代表数据修改)
(移动端
根据位置更新调用对应接口===》更细本地数据)

移动端：

- 1、移动端拉取接口策略算法：： 消息事件 + 一定的机制
（比如多久没有查询接口则 调用一次） ==》 会一定程度上 减少 消息达

到率 非100% 的问题

===》感觉这样 ===》 会大大 降低 一定端 调用接口 频率，减少带宽，IO损耗 等

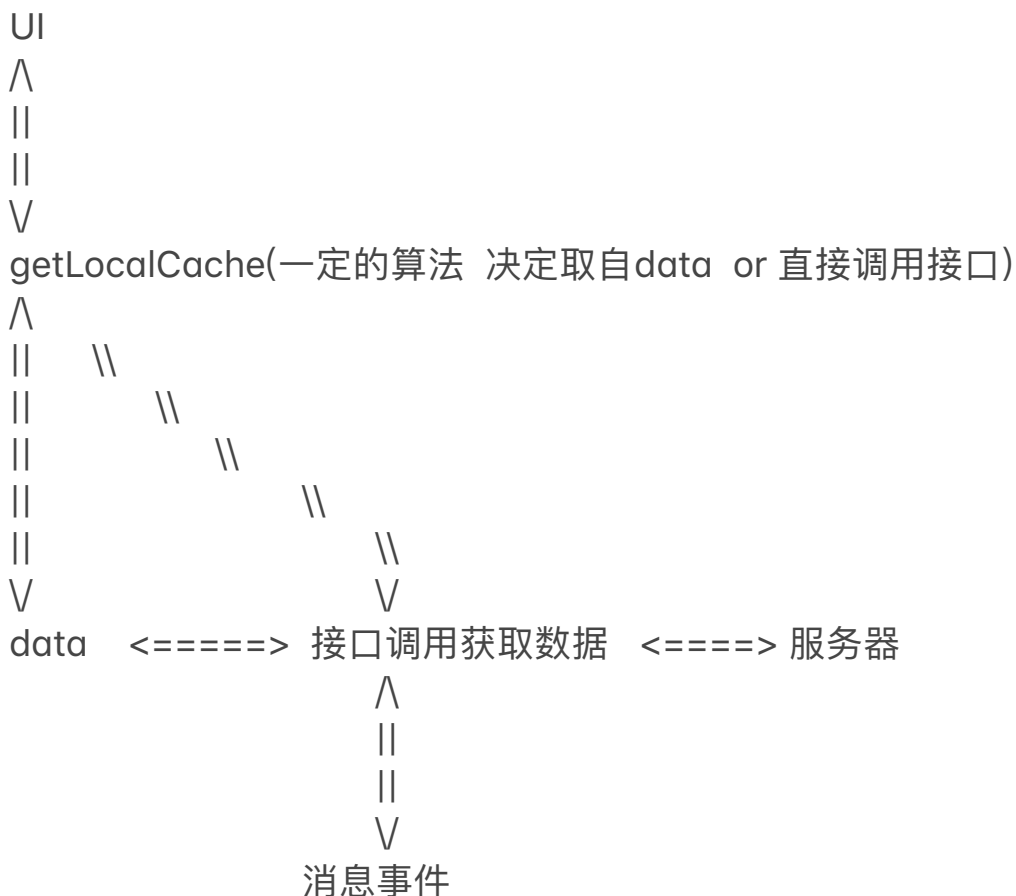
2、程序员调用接口，在使用这套移动端框架的时候，需要有一套注册机制

3、服务器推送消息 =====》具体到某个接口 =====> 接口调用 =====》返回数据 =====》更新本地数据

4、移动端调用流程：

```
UI ==> getLocalCache ==> data
```

5、移动端框架 分层



6、未开启 此模式（或者未注册到此模式时）

===》 每次接口调用 ===》 返回UI && 更新data(判断是否数据修改)

7、开启 此模式

1、这个接口上次调用时间 + 最近事件 > maxtime (比如5s) ==> 在调用接口 ==》等 && 更新本地数据 + 更新接口最新调用时间

2、这个接口上次调用时间 + 最近事件 < maxtime (比如5s) ==> 在调用本地数据，并 更新调用本地数据的最新时间

8、试用接口类型

幂等性一致的接口 (对于 增、改、删的接口不适用，因为消息可能重传，多次发送，不具有幂等性的 接口适合这个框架)

9、困难：

1、这改动量很大呀!!! 移动端 修改修改&& 服务器端需要修改 ==》如何无侵入式 框架设计很重要

2、移动端框架：应该改动很多 (如果能使用类似插件机制 (这个框架是一个插件)，应该会很好 && 服务器端也已一样)

3、服务器端：：原来的 不需要改变 ==》只需要 微服务 ==》发出消息事件即可 (把发消息搞定应该就可以了)

10、固有缺点：

消息到达率 推送的准确率!!! 并不是100%; 但是即使你每次调用接口，也保证不了100%

(IM服务器、消息队列框架 确保 ==》消息的到达率) &&加上自身的框架 的一定机制 ==》来提高 到达率 + 可用性 (即消息队列IM,挂了，依然能用)