**App版本更新接口的设计**

工作这几年碰到的版本检测升级的接口也算是五花八门，啥样的都有，但肯定有的功能是有个apk的下载链接，能间接或直接提示你是强制还是非强制更新：

* **间接**是指提供你后台最新版本号，让你自己与本地版本号通过比较得出是否升级或强制升级；
* **直接**就是后台接口直接返回个Boolean类型告诉你是强制或者非强制更新。

个人认为一个好的版本检测接口需要设计的更灵活更清晰用起来更方便，下面就我理解的接口设计如下（如思路有误，欢迎指正）：

总字段如下(并不是所有字段都要返回给客户端)：

1.最新版本号 ：newVersion

2.最小支持版本号 : minVersion

3.apk下载url : apkUrl

4.更新文案 : updateDescription

5.是否有更新 : isUpdate

6.是否强制更新 : forceUpdate

可选字段：

7.apk文件大小：apkSize

8.apk的文件MD5值：md5

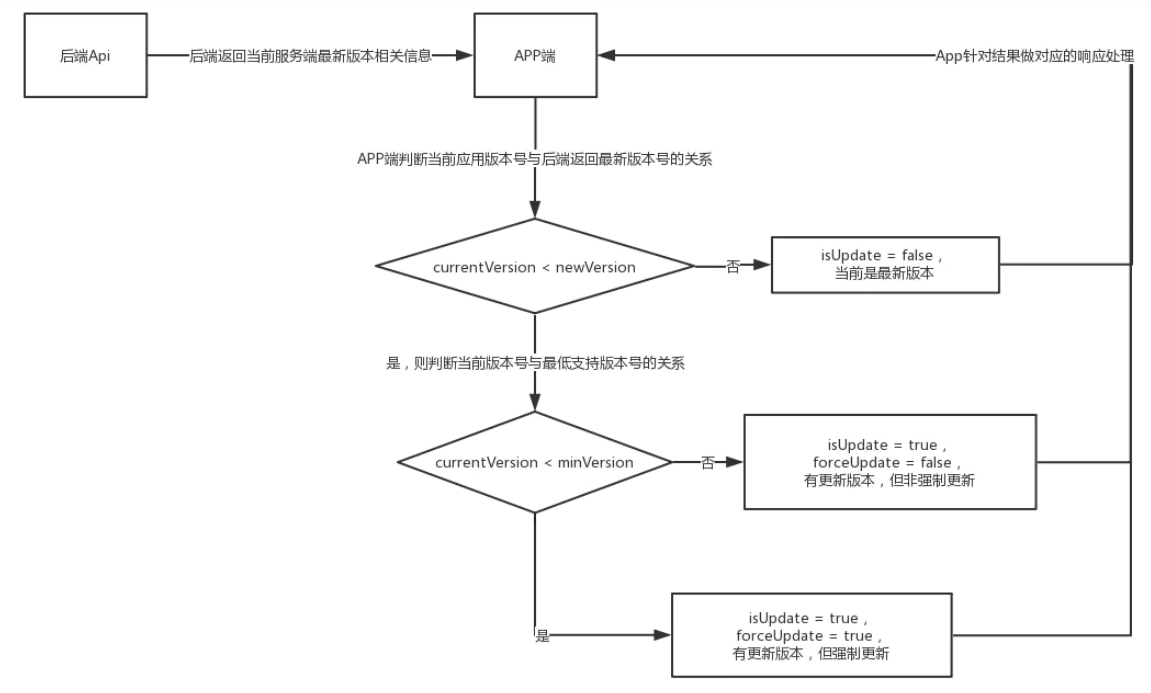
**方案一（前端处理逻辑）：**  
前端调用获取当前服务器端最新版本信息的接口，然后根据返回的服务器端最新版本信息与客户端当前版本信息进行比较，此处需要后端提供最新版本号 ：newVersion 、 最小支持版本号 : minVersion 、apk下载url : apkUrl 、更新文案 : updateDescription 这四个字段。如下表：

| **字段** | **字段描述** |
| --- | --- |
| newVersion | 最新版本号 |
| minVersion | 最小支持版本号 |
| apkUrl | apk下载url |
| updateDescription | 更新文案 |

客户端逻辑如下：

1. 如果currentVersion < newVersion,则有更新信息；
   * 如果currentVersion < minVersion,则需要强制更新；
   * 如果currentVersion >= minVersion,则不需要强制更新;
2. 如果currentVersion == newVersion，则没有更新信息。

**客户端对应的流程图：**



更新处理逻辑放客户端.png

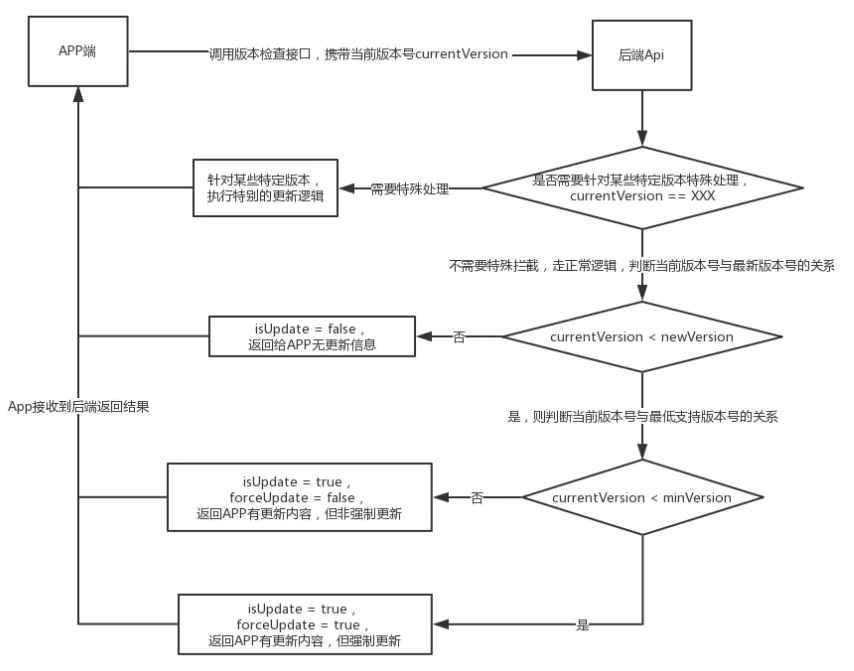
**前端处理逻辑不足之处：**  
对指定版本进行强制更新可能不是那么好，可是为什么要对指定版本进行更新呢？  
此处做个假设：你的应用当前线上运行版本为V2.1.1, V2.1.2两个版本用户量比较多，又快要发布V2.1.3了，此时发现V2.1.2有个bug需要修复一下，然后在V2.1.3修复后发布了，当时感觉小问题就非强制更新（毕竟强制更新比较反感），过了几天查看V2.1.1, V2.1.2版本的用户量依然不减，升级的积极性不高啊，可是现在发现V2.1.2的这个bug风险很大，一旦出问题要卷铺盖走人的地步，那就要强制更新了，可又不想影响V2.1.1和V2.1.3版本的用户（对强制更新比较反感），此时想到要是能对V2.1.2一个版本强制更新就好了。  
**备注：**此场景只是一个假设，当然即便碰到这个假设，只要前期设计的合理都能解决，比如：

* 后端说我再多返回一个字段forceUpdateList给你，内容为指定版本强制更新的版本号列表，只要当前版本在该列表中就强制更新。（此方案前端表示想哭）
* 或者不想要强制更新的版本号列表也可以，给你个字段forceUpdateVersion表示需要强制更新的版本信息,用正则匹配，如果匹配到的版本号就强制更新。（前端能做，后端也能做，看谁能说服谁了）
* 可能也有人会说热修复或者插件化表示不担心，那就绕过吧，此处目的是为了分析版本升级的最优方案。

**方案二(后端处理逻辑)：**  
在客户端请求参数中添加当前版本号currentVersion传输给后台，由后台根据客户端传过来的当前版本号currentVersion做相应的判断后给出是否强制更新。  
**后端逻辑如下：**

1. 根据是否有特殊需求可指定某个版本必须强制更新，如currentVersion == XXX,则forceUpdate = true;
2. 如果currentVersion < newVersion,则isUpdate = true；
   * 如果currentVersion < minVersion,则forceUpdate = true；
   * 如果currentVersion >= minVersion,则forceUpdate = false;
3. 如果currentVersion == newVersion，则isUpdate = false.

**后端逻辑对应流程图：**



更新处理逻辑放后端.png

**结论：**  
返回客户端的字段仅需要apk下载url : apkUrl、更新文案 : updateDescription、是否有更新 : isUpdate 、 是否强制更新 : forceUpdate 这四个字段即可。如下表：

| **字段** | **字段描述** |
| --- | --- |
| apkUrl | apk下载url |
| updateDescription | 更新文案 |
| isUpdate | 是否有更新 |
| forceUpdate | 是否强制更新 |

**总结：**  
细心的你可能会发现上面的可选字段apkSize和md5并没有用到，既然是可选字段也就是可用可不用，根据需要决定是否采用，这里来讲下他们的用处。

* **apk文件大小apkSize**  
  这个用处可以说出于考虑用户体验，需要在升级弹框出来展示给用户将要更新的内容多大，让用户决定在非WIFI状态是否要更新，不能为了拉用户下载量或所谓的UV数直接让用户在不知道大小的情况下去直接下载（土豪用户绕路）。
* **apk的文件MD5值**  
  这个主要是出于安全考虑吧，因为文件内容固定的话对应的md5是一样的，我们可以通过这个md5值来和下载的apk的md5值进行比较去保证我们从服务器更新下载的apk是一个完整的未被篡改的安装包，也就是说如果我们下载的apk的md5值和服务器返回的md5值相等，则说明我们下载的apk是完整的，且没有被相关有心人处理过的apk。

