**可更新列表性能优化**

软件三部 彭小春

2015年10月

## 一、前言（问题提出）

nubia应用中心可更新列表刷新、加载时间过长，用户体验差，需要对其进行性能优化。

## 二、实现方案（实现、改进方案、解决思路）

优化点：

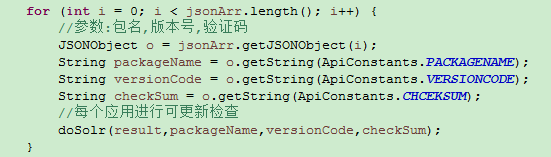
1、把应用封装成列表，调用solr接口，并返回列表，减少调用solr的次数。

2、返回给客户端的更新信息，去掉不需要字段，减少传输大小。

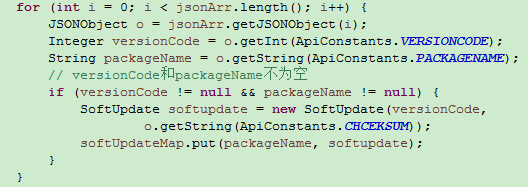
3、尽量避免访问数据库。

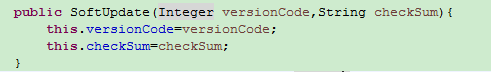
客户端传入json值，调用“获取可更新列表”接口。服务端对json值进行处理，返回可更新信息。

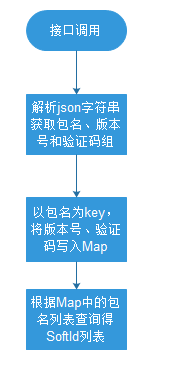
原方案在解析json字符串时，对每个应用调用solr接口进行查询，效率慢。



为了提高效率，新方案在流程上进行了优化。在解析json字符串时，将应用封装在map中，以包名packageName为key值，将versionCode、checkSum写入map中。只需要传入包名列表，调用一次solr接口，便能能到所有满足条件的softid列表，大大提高了访问效率。

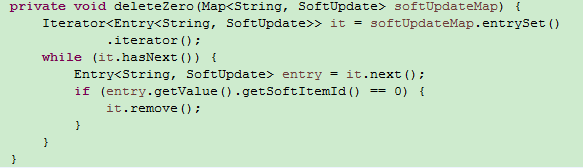


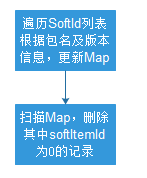




遍历softId列表，根据softId获得根据包名和版本信息，判断通过softId取得的版本号是否大于map中的版本号，若大于则表明有可更新版本，更新Map中的softId值（该值在Map中默认为0）。扫描Map,删除所有SoftId为0的记录，对于无更新版本的应用，不在继续判断是否有差分信息，提高效率。

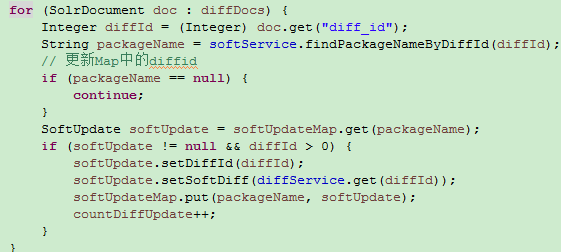




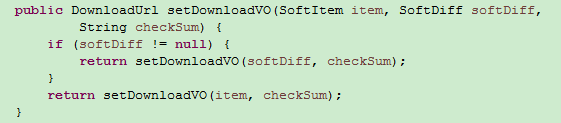


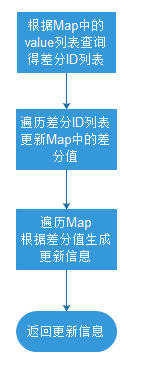
以更新后的Map为参数，调用solr接口得到差分列表。遍历差分列表，更新Map中的SoftDiff值。

C:\Users\pxc\Desktop\捕获.PNG

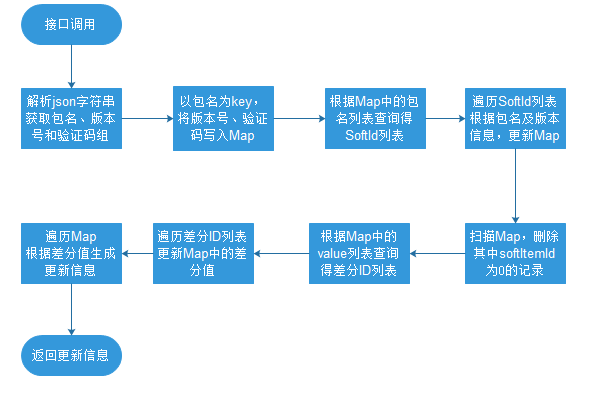


遍历Map表，根据SoftDiff生成更新信息，若SoftDiff不为空，表明有差分包，封装差分包信息，否则封装SoftItem中版本信息。



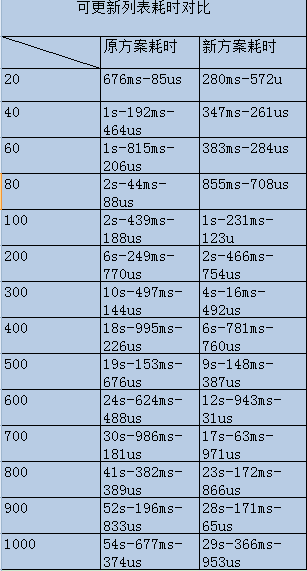


整体流程图如下：



## 三、实践情况（自测、测试验证情况）

按照文档中设计方案进行编码，对结果进行自测，并与原方案进行耗时比较，新方案耗时减少到原方案的一半，自测通过。



## 四、总结及备注

在需要进行多条数据处理的时候，用列表访问并返回列表比逐条访问的效率高。在需要进行数据查询时，访问solr比直接访问数据库的效率高。有其他解决方案时，尽量避免直接访问数据库，避免重复访问数据库，当查询的结果在下次需要被重复访问时，建议把该结果添加到Map中，提高效率。

## 参考资料

无。