Apache Kylin对于Cube的最小存储单位为data segment，类似于Hive的partition，data segment采用左闭右开区间表示，如[2015-11-01，2015-11-02)表示含有2015-11-01这一天的数据。对于Cube数据的管理主要基于data segment粒度，大致分为3种操作: 计算(build)、更新(refresh)、合并(merge)。对于一个具体产品来说，它的数据是需要每天例行计算到cube中，正常例行下，每天会生成1个data segment，但可能会因为数据仓库的任务延迟，2天或多天生成1个segment。随着时间推移，一方面，大量的data segment严重影响了性能，另一方面，这也给管理带来了困难和麻烦。因此，对于1个cube，我们按照1个自然月为1个data segment，清晰且易管理。

假设我们有1个月30天的数据，共23个data segment数据片段，如：[2015-11-01，2015-11-02)， [2015-11-02，2015-11-04)， [2015-11-04，2015-11-11)， [2015-11-11，2015-11-12)， [2015-11-12，2015-11-13)， 。。。[2015-11-30，2015-12-01)

**问题1**: 假设因为数据有问题，需要回溯2015-11-01的数据，因为我们能够在cube中找到[2015-11-01，2015-11-02)这样一个data segment，满足这个时间区间，于是，我们可以直接界面操作或者Rest API启动这个data segment的refresh更新操作。

**问题2**: 假设我们需要回溯2015-11-02到2015-11-03的数据，同理，可以找到一个符合条件的data segment [2015-11-02，2015-11-04)，然后refresh更新这个data segment。

**问题3**: 假设我们需要回溯2015-11-01到2015-11-02的数据，我们找不到直接满足时间区间的data segment。于是我们有2种解决方案，第1种方案是分别依次refresh更新 [2015-11-01，2015-11-02)， [2015-11-02，2015-11-04)这2个data segment实现；第2种方案是先合并(merge)[2015-11-01，2015-11-02)， (2015-11-02，2015-11-04)这两个data segment，合并后得到[2015-11-01，2015-11-04)这样1个data segment，然后我们再拉取新数据后执行更新操作，即可满足需求。

**问题4**: 假设我们需要刷新2015-11-01~2015-11-30这1个月的数据，需要将23个data segment合并成[2015-11-01，2015-12-01)这1个data segment，计1次操作。然后再执行1次更新操作，共2次操作即可完成需求