Atitit 频繁插入更新表优化

使用堆表

2、IOT因为是有序的，所以在批量导入大量无序数据方面，IOT的速度要比堆表差很多；

在Oracle数据库中，IOT一般适合于那些不怎么频繁更新且又需要快速按主键访问的数据库表。

二、堆表和索引组织表的比较

所以，堆表的特点就是索引和数据分开，所有索引都是二级索引，或叫辅助索引。所以主键索引也是二级索引，没有完整记录，区别只有唯一或非唯一。索引中存储的是 key 与指针，指针指向具体数据记录。当然，查找 key 的算法都是一样的，使用二分查找，也叫书签查找。

## 索引组织表更新效率更高

这里我们想一下，由于堆表的索引只存储索引 key 以及数据行在堆表上面的绝对位置(页号，页内偏移)，如果这行记录发生了更新，并且不能原地更新，需要进行迁移，那么就会发生一个情况，堆表所有索引都需要修改，指向新的行位置。而索引组织表就不需要这个开销，更新只会新插入变化了的索引 key，不变的索引 key 不需要新插入，只有当主键发生更改才需要对应修改其他二级索引，通常主键也不会更新，所以这也就是索引组织表更新效率更高的由来。

但一般使用堆表的数据库都不会直接这么更新，代价太大了。而是会在数据页预留一些空间，当遇见不能原地更新的记录时，就会在这个页中新插入一条更新后的记录，然后在这个页中做一个指针把老记录指向新纪录，这样就不用更新索引信息了。当然，如果你这个页中没有空闲空间可以插入新的记录，那么还是需要做行迁移，然后更新所有索引。

Oracle支持堆表，也支持索引组织表  
PostgreSQL只支持堆表，不支持索引组织表  
Innodb只支持索引组织表

MySQL的MyISAM 和 oracle都支持堆表

## 那么堆表的缺点是什么呢？

这里我们想一下，由于堆表的索引只存储索引 key 以及数据行在堆表上面的绝对位置(页号，页内偏移)，如果这行记录发生了更新，并且不能原地更新，需要进行迁移，那么就会发生一个情况，堆表所有索引都需要修改，指向新的行位置。而索引组织表就不需要这个开销，更新只会新插入变化了的索引 key，不变的索引 key 不需要新插入，只有当主键发生更改才需要对应修改其他二级索引，通常主键也不会更新，所以这也就是索引组织表更新效率更高的由来。

但一般使用堆表的数据库都不会直接这么更新，代价太大了。而是会在数据页预留一些空间，当遇见不能原地更新的记录时，就会在这个页中新插入一条更新后的记录，然后在这个页中做一个指针把老记录指向新纪录，这样就不用更新索引信息了。当然，如果你这个页中没有空闲空间可以插入新的记录，那么还是需要做行迁移，然后更新所有索引。