## 提货任务列表慢SQL问题处理总结

**时间：**2月19

**内容：**散航提货列表页面无法显示，经排查是慢sql问题,在755X的运行时间达到31s

**接口作用:**获取提货的航班落地的情况以及已提货物和未提货物的数量情况，并根据车标聚合，这就涉及到了货物表，提货板箱表，提货车标表三个表的关联查询

**原因：**

1.子查询中没有使用分片和父查询中已经定义好的条件，没有在第一步查询时就过滤掉大部分的数据。

2. SQL对应的业务场景较为复杂，跨多个表，长度达到88行



**处理方法：**

1. 将分片查询前置到所有查询子句之前，将分片查询之后的结果字段再拿来进行子查询 时间降低11s
2. 将时间查询前置到所有子查询之前，将时间查询之后的结果字段再拿来进行子查询 时间降低到2s
3. 使用CASE WHEN根据条件出现的概率进行先后顺序的排列,并对索引字段使用is null 不破坏索引 0.3-0.5s



**总结**

1. 在处理复杂的SQL时需要先明确思路，不能一口气写下来
2. 任何时候都需要意识到索引的重要性，特别是分片索引，要合理利用，避免出现大范围检索的情况
3. 如果实在要在SQL中体现了某种特殊的业务场景，一定要注释标明
4. 简单SQL确定索引的使用情况，养成良好习惯，复杂SQL必须先在生产库上跑上一遍，因为数据量的不同，测试环境体现不出这中间的问题
5. 由于表字段的大量冗余，在很多情况已经能够避免跨表查询的问题，但是项目主要需要需要适配不同的业务场景，所以在很多情况下会产生一些奇怪的业务分支来给特殊的业务场景兜底，所以复杂SQL的出现并不少见，这时候需要尤为仔细的检查有没有逻辑漏洞和可优化的地方，这需要项目经验的积累向前辈的不断请教。