

优化方案：

背景介绍：

AOC动态航班数据库一共有5.7W+条数据，一次操作发布10条以上的通知单，会操作大约1w+的数据，操作方式是在一个事务里循环遍历10条通知单，按每个通知单的操作对数据库增删改查，其中新增航班的时候需要根据航线计划生成出所有的航班动态，逐条插入，单独检测每一条数据是否已经有相同的航班，有的话则过滤，没有的话则插入。

所以发布多个航线计划的时候特别是有新增的话，会在一个事务中来回操作很多次数据库，每次操作数据库都是有成本的，时间累积导致多条通知单的时候会耗费较长的时间。

优化思路：

在发布航线计划的时候，尽量减少对数据库的操作次数。这样就能大大缩短执行一次发布的时间。

- 先在内存中修改航班动态，避免操作数据库：

把航班动态的数据加载到内存中，发布多个航线计划的时候，遍历航线计划，新增的数据，直接遍历内存中的航班数据过滤掉重复航班，剩下的添加到内存中，标记状态为INSERT，修改时刻和运力以及取消和恢复的直接修改内存中的航班动态，标记状态为UPDATE，删除的数据，直接标记状态为DELETE。计算完毕之后根据数据状态，把数据再保存回数据库。

- 保存到数据库的时候优化SQL，减少数据库操作：

- 1.对于新增的数据，按一定的批数量batchSize，按批次每次写成一句SQL直接执行

```
INSERT INTO t_fligh() VALUES(),(),();
```

- 2.对于修改的数据，不能一句一句update，创建一个临时表，把修改的数据也按批次插入进临时表，然后再执行一句update，通过临时表和航班动态表的id想关联，一次更新完毕。

```
CREATE TEMPORARY TABLE IF NOT EXISTS t_flight_tmp LIKE t_flight;

INSERT INTO t_flight_tmp () VALUES (),(),();

UPDATE t_flight f, t_flight_tmp tmp SET f.departure_time=tmp.departure_time,
f.arrival_time = tmp.arrival_time,f.airplane=tmp.airplane, f.canceled =tmp.canceled
WHERE f.id = tmp.id;
```

- 3.对于删除的数据，也优化成一条删除语句：

```
DELETE FROM t_flight WHERE id IN();
```

测试结果：

- 硬件环境：

公司开发台式机：cpu i7-6700@3.4GHz 内存 8G

- 软件环境：

MySQL 5.7.16

编写语言Java，运行时JVM进程配置-Xms1024m -Xmx1024m，使用内存1G。

- 测试结果

数据库中一共56150条航班动态数据。

- 新增一条航线，时间段为1年，排期为1,2,3,4,5,6,7，所以共需要插入365条航班动态，第一次耗时1400ms，第二次及以后680ms
- 一次发布十条通知单，包含2条航线新增，4条航线修改，2条航线取消，2条航线删除,共插入365条数据，修改1672条数据，删除1460条数据，耗时1204ms

配置和运行：

在MySQL数据库中创建flight数据库。程序会自动建表。

- 配置文件：application.properties

```
#MySQL数据库连接
spring.datasource.url=jdbc:mysql://localhost:3306/flight
```

```
#数据库用户名
spring.datasource.username=root
#数据库密码
spring.datasource.password=root
```

- **直接执行jar包**

打开程序目录在命令行执行`java -jar -Xms1024m -Xmx1024m flight-1.0-SNAPSHOT.jar`即可运行程序

打开浏览器访问`localhost:8080/index`即可

- **Maven打包运行**

- 配置文件目录`src/main/java/resources/config/`

- 打包

打开程序目录，`mvn package`，成功后会在`target`目录下生成`flight-1.0-SNAPSHOT.jar`

- 运行

在命令行执行`java -jar -Xms1024m -Xmx1024m flight-1.0-SNAPSHOT.jar`即可运行程序

打开浏览器访问`localhost:8080/index`即可

- **IDEA导入和配置**

- File --- Open打开项目

- 配置文件目录`src/main/java/resources/config/`

- 运行

选择Run---Configuration添加一个Application

配置JVM参数 VM options:-Xms1024m -Xmx1024m

主类Main class:net.rlair.flight.config.Application

选择Run---Run即可运行程序

打开浏览器访问`localhost:8080/index`即可

- **初始化航线计划**

在`src/main/java/resources/database`目录下有`init.sql`文件，可以用来初始化航线计划，会新增80条航线计划，全部执行后会生成2.7w条航班动态