## 步骤

1.产品建立pmp

2.接口约定可以先给初版,外包给了“京东自营机票接入接口规范TTS直连V1.10”文档,将文档上传到air.document

3.找外包要包机切位tts客户端jar包,jsf的jar包升级为1.0.6,将jar包放在D:\repository\com\jd\air\self\api\jd-air-flight-tts-api\1.0.6-SNAPSHOT中。

4.airplane-model中重新import-install。

5.airplane-consumer中重新import-install。

6.airplane-gw、airplane-jipiao中重新import-install

7.排期

8.UC设计，完了评审(考虑来源，过滤, 还没有评审)

9.产品经理杨彬彬给交互稿，安排前端资源。与杜守珺协商打标及浮层字段（考虑以后移动端接入）

10.与工单约定，tts.jsf.alias和tts.handler.jsf.alias，开发完打包后jdos中部署，完了告知工单相关人员

## 范围

1.产品建立pmp

2.接口约定可以先给初版,外包给了“京东自营机票接入接口规范TTS直连V1.10”文档,将文档上传到air.document

3.找外包要包机切位tts客户端jar包,jsf的jar包升级为1.0.6,将jar包放在D:\repository\com\jd\air\self\api\jd-air-flight-tts-api\1.0.6-SNAPSHOT中。

4.airplane-model中重新import-install。

5.airplane-consumer中重新import-install。

6.airplane-gw、airplane-jipiao中重新import-install

7.排期

8.UC设计，完了评审(考虑来源，过滤, 还没有评审)

9.产品经理杨彬彬给交互稿，安排前端资源。与杜守珺协商打标及浮层字段（考虑以后移动端接入）

10.与工单约定，tts.jsf.alias和tts.handler.jsf.alias，开发完打包后jdos中部署，完了告知工单相关人员

## 工单新增字段

Jar包：jd-air-flight-tts-api

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **isSupportIntegral** | 是否支持常旅客积分 | Integer | 否 | 0-否 1-是 |
| **limitTicketTime** | 最晚出票时限 | Integer | 否 | 最晚出票时限（小时），自营包机切位为必填项。非包机切位产品返回null，大于0。 |
| **limitOrderMixNum** | 最小预定人数限制 | Integer | 否 | 自营包机切位为必填项 |
| **limitOrderMaxNum** | 最大预定人数限制 | Integer | 否 | 自营包机切位为必填项 |

## 打标

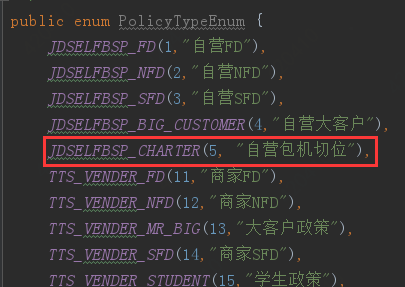
|  |  |  |
| --- | --- | --- |
|  | 打标 | 浮层 |
| 出票慢 | 固定 | slowIssueText;  “特殊产品出票慢，航班起飞前1天出票，支付成功后保障出行”（包机切位都按出票慢处理，limitTicketTime >=72小时按照天显示，<72小时按照小时显示） |
| 退票慢 | 固定 | slowRefundText；固定  “特殊产品，退票审核需3-5天，请耐心等待” |
| 预定人数限制 | 后台拼文案：limitPersonNum ：“限2~ 5人购票” | limitOrderText；  后台拼文案： “2~ 5人同时购买可享受此价格。”  **limitOrderMixNum和limitOrderMaxNum不会为空，最小为1** |
| 包机切位 | 京东特惠（seatType=5） | 无 |
| 发票类型 | 不提供报销凭证  classLimit.bill == “NoReimbursementVoucher” | 无 |

## 与工单约定接口

1. 约定TTSSeatInfo中的字段，TTSSeatInfo在jar包中，jar包升级到1.0.6

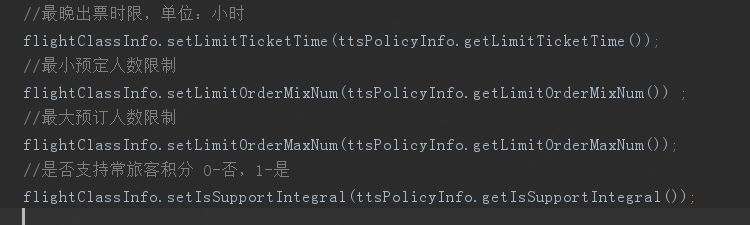
## 网关与工单

网关请求工单的ttsHandler中不需要加字段,但是policyType需要加一个枚举值（若接口文档修改了4.1.6的agencyInfo内容，请求就要变）



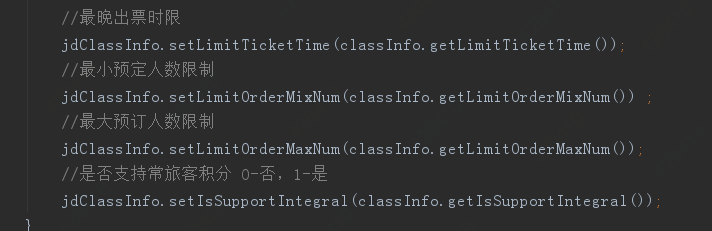
在gw请求工单的参数时policyType是{1,2,3,4,5}

1. 在common中的flightClassInfo中增加字段（网关和前台公用）
2. 在TTSHandlerUtil中seatInfoToFlightClassInfo方法设置（回调），TTSseatInfo转成网关的flightclassInfo（common）



## 网关与jipiao内部

1. 将网关的response处理后转换成内部的，DataCenterImpl中coverFlightInfo方法中转换



## Jipiao内部

1. FlightClassInfo中增加字段（用于内部逻辑处理）

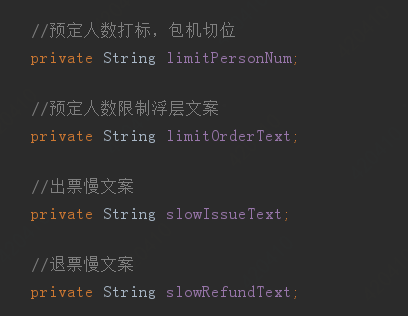


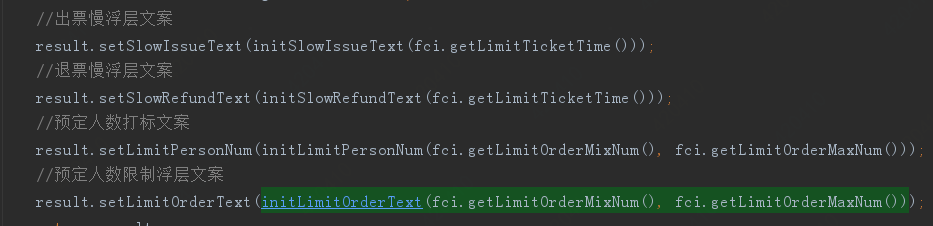
## jipiao与前端

1. FlightClassBean中添加字段（给前台用，model包）
2. QueryServiceImpl的convertFlightClassBean中设置字段（新增），flightClassInfo（jipiao内部）设置到flightClassBean

技改后将代码改到FlightBeanTransfer类中

在dataCenter.queryFlights(query, ip)之后，getFlightResult方法中给前台打标





## Model与工单

1. Model中引入了工单的包，所以工单升级包，改 jar包的version为1.0.6

<dependency>  
 <groupId>com.jd.air.self.api</groupId>  
 <artifactId>jd-air-flight-tts-api</artifactId>  
 <version>1.0.5-SNAPSHOT</version>  
</dependency>

## Consumer与model

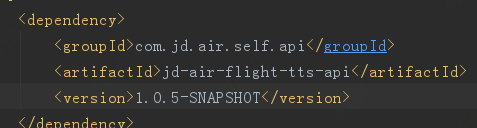
1. Consumer引用model的jar包，1.0.0版本，不需要更改

## Jipiao与model

1. Jipiao引用consumer的jar包1.0.0，consumer引用model的jar包1.0.0，model引用工单的包升级版本

## Gw与model

1. gw引用consumer的jar包1.0.0，consumer引用model的jar包1.0.0，model引用工单的包升级版本



备注：

1. 测试环境版本加日期，不要用1.0.0，若model版本改了，jipiao和consumer改

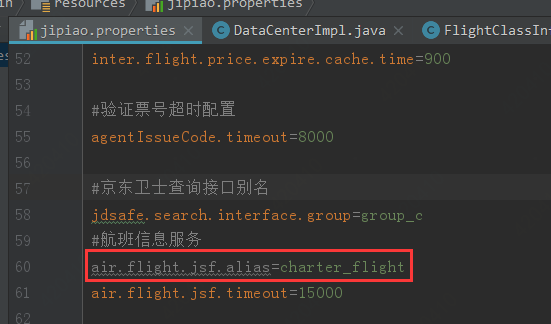
2. 加字段需要加到最后

3. 工单改了版本，只需要model中该版本

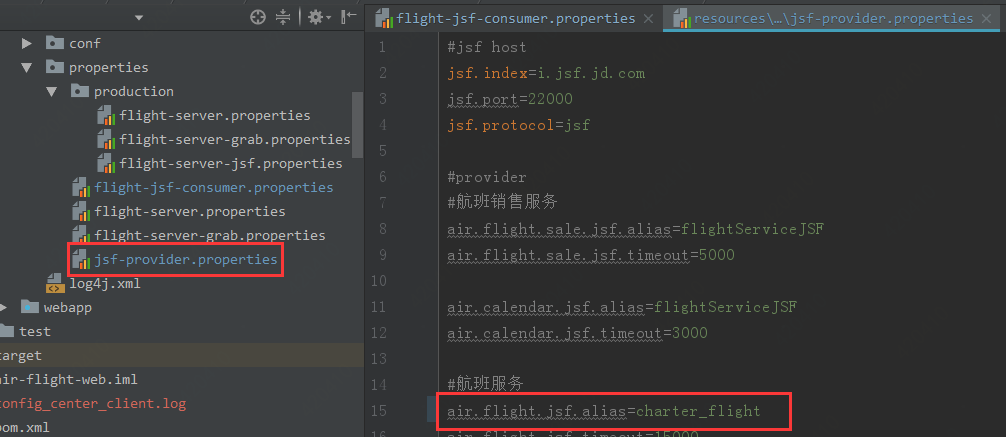
## Jipiao与air-flight-Server（有关退改签）

**Jipiao中调用jsf接口**

**接口：com.jd.airplane.flight.service.IFlightService**



air-flight-Server中提供jsf接口

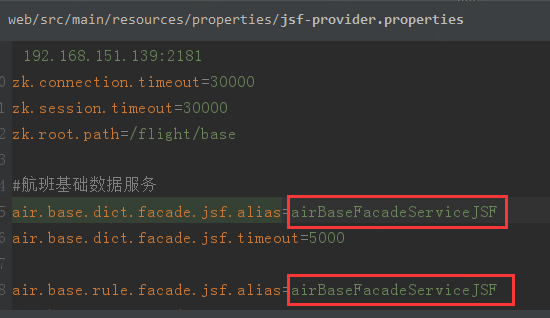


部署到jdos上后，一定要部署war包

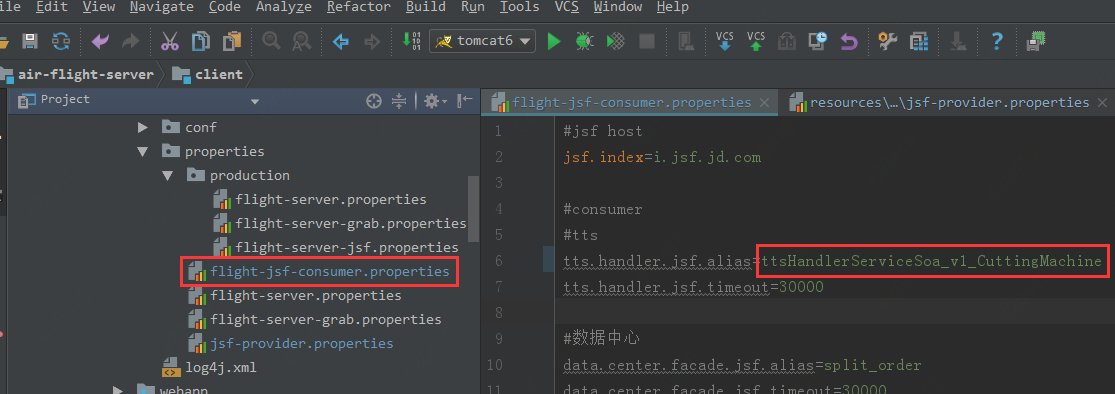


## air-flight-Server与air-base-server（有关部署）

air-flight-Server中部署了就是用本项目中的，若没有部署使用air-base-server中的



## air-flight-Server与工单（有关退改签）

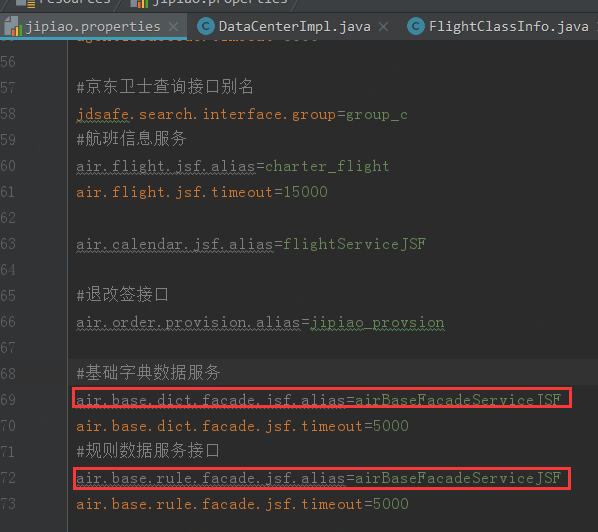


## Jipiao与air-base-server（有关基础数据）

接口：

com.jd.airplane.base.service.facade.RuleFacadeService

com.jd.airplane.base.service.facade.DictFacadeService



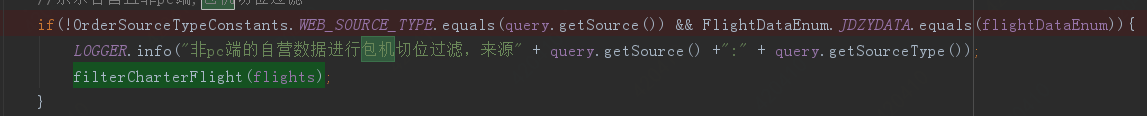
当基础数据挂掉后，访问本地快照文件，若本地没有快照文件，则直接挂掉服务

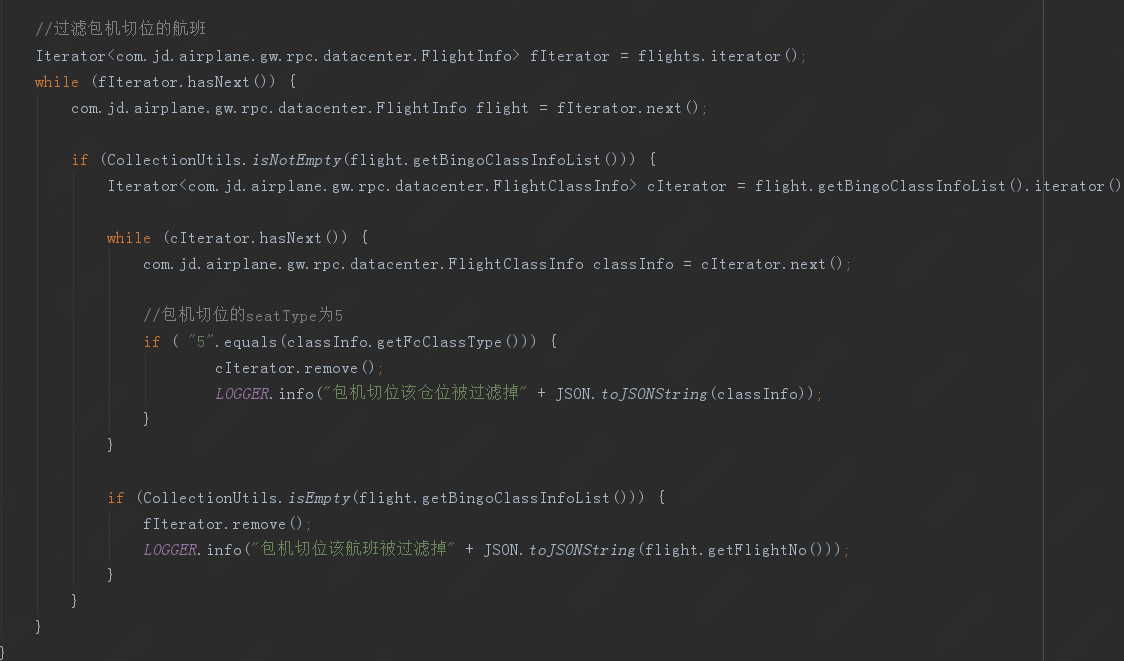
## 过滤移动端数据

移动端吐出的航班信息是flightBean，和给前台的一样

见文档

[京东国内机票JSF接口](http://cf.jd.com/pages/viewpage.action?pageId=96875967)，该文档是给移动端看的，由于移动端没有接包机切位，因此可以不更新





改进：

FlightDataEnum.JDZYDATA中包含的数据除了自营的还有东鹰代运营的。

所以进一步通过venderId过滤，venderId取舱位上的，因为航班上的可能为空。

将字符串“5”使用seatTypeEnum枚举值代替

京东机票前端接口文档V1.8，该文档是与前端交互的文档

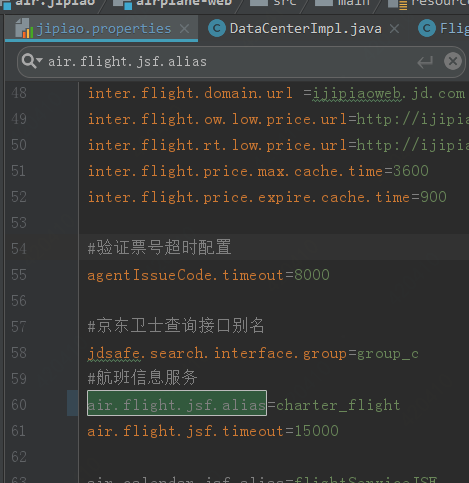
## 包机切位退改签：

Jsf配置

### 前台（jipiao）🡪flight Server

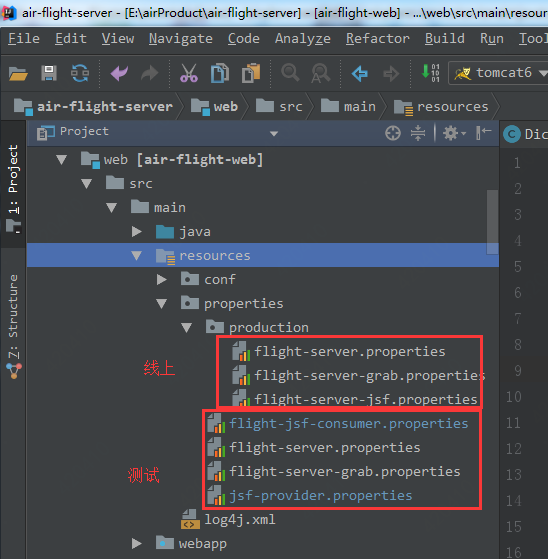
别名：air.flight.jsf.alias

日志：air\_flight.log



### Flight server--工单

air-flight-server项目中，需要引用：air-flight-client的jar包

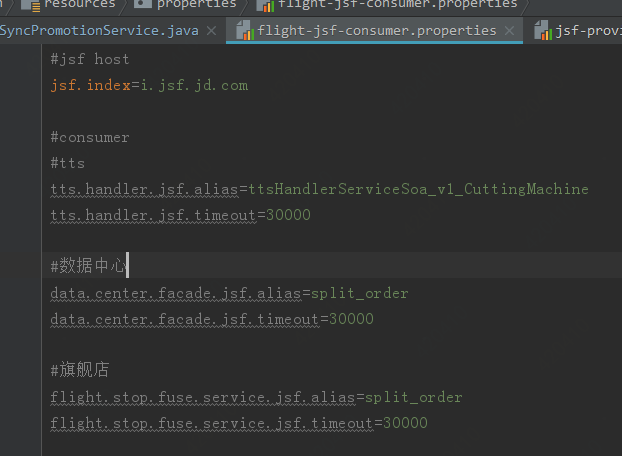


别名：tts.handler.jsf.alias

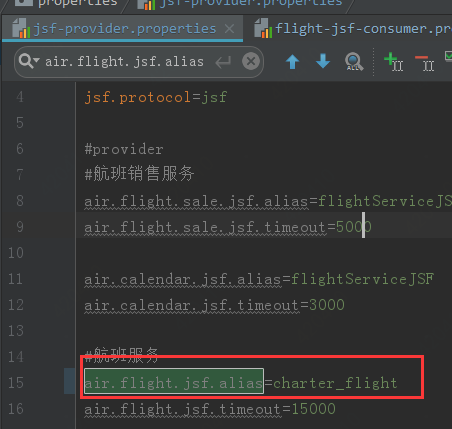
日志：flight.log

数据中心是普通商家。

测试环境



flightServer提供给前台的服务

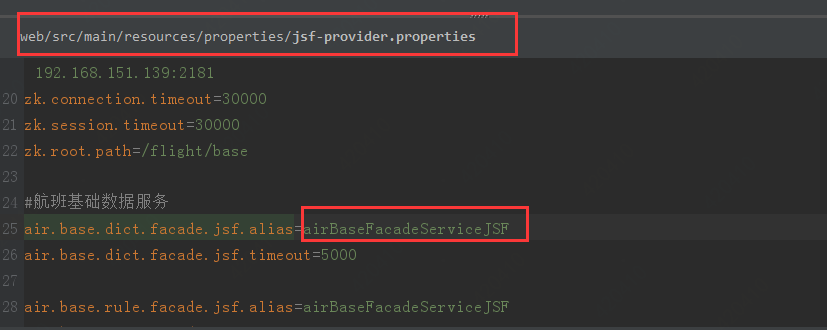


FlightServer—base（通过client的jar包调用jsf服务）

接口： com.jd.airplane.base.service.facade.DictFacadeService

现在存活的别名：





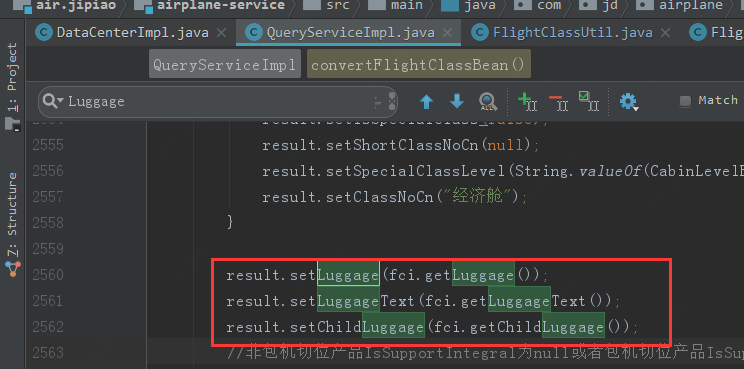
## 行李额

航班查询接口增加两个字段，成人行李额和儿童行李额。

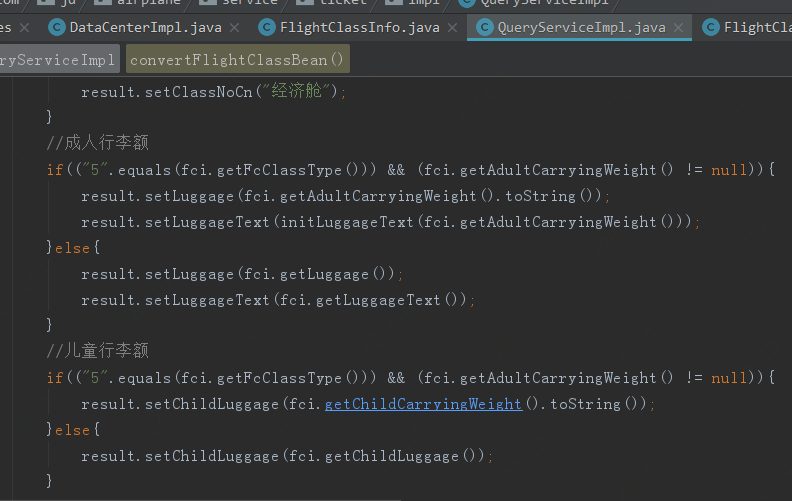
新增两个字段如果返回为空，请从基础数据接口获取展示行李额，如果不为空，则取这两个字段展示。

行李额是调用工单接口返回的，在setLuggage方法中，调用链：DataCenterImpl—flightInfoHandle--FlightClassUtil.*setFlightData*方法，在往前台写行李额数据之前。

现有的往前台传行李额：



改后：



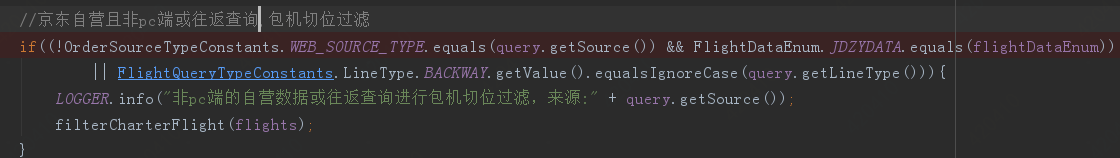
## 发票类型

Jipiao项目中：getAirFlightLimitMap（）方法

## 往返查询过滤掉

原因：配送类型过滤掉不一致的航班和舱位

过滤：



## 上预发的准备

### 考虑是否影响线上用户

设置白名单（白名单不为all， and不在白名单）不展示）

1. 将过滤的代码push到dev上，commom中相应改变

最新的dev，check out到本地分支20180330\_filter\_charter，代码粘贴上，测试没问题，push到远端。在check out到dev，origin/20180330\_filter\_charter来merge到本地dev，将dev取push，完了合master。上线。

1. air-base-model和service包1.0.3

### 退改签别名

预发时，需要将air-flight-sever中调用工单的别名改一下

工单和air-flight-sever需要调用同一个jsf

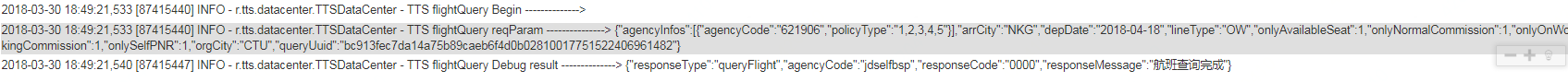
接口：com.flight.manager.service.HandlerService

别名：ttsHandlerServiceSoa\_v1\_yf

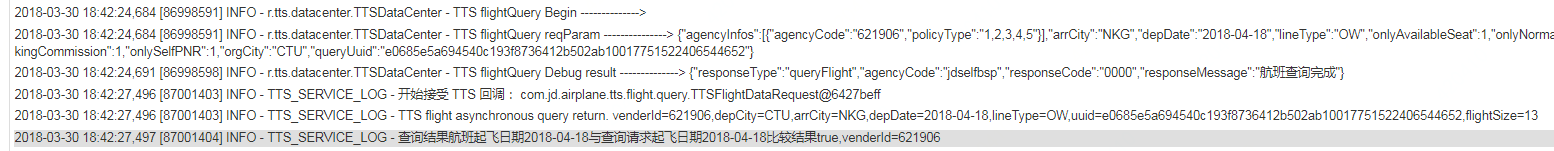


### 打标时有时无

分析：没有打标时，工单没有回调。



正常日志：

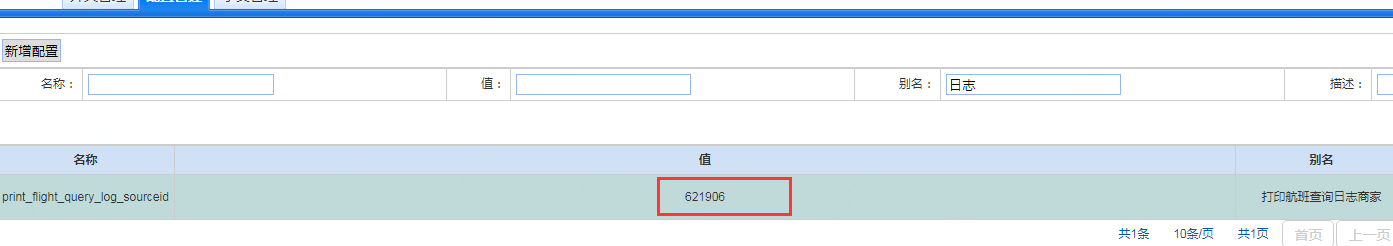






具体原因：

将日志开关打开：print\_flight\_query\_log\_sourceid



解决：

预发两台机器下面接口的别名一样，可能回调到另外一台机器上了。所以tts回调日志时有时无。将172.19.116.142和10.190.30.15两台机器的别名改成不一样。然后j-one中重启机器，重启没用，重新构建发布。



改别名后：

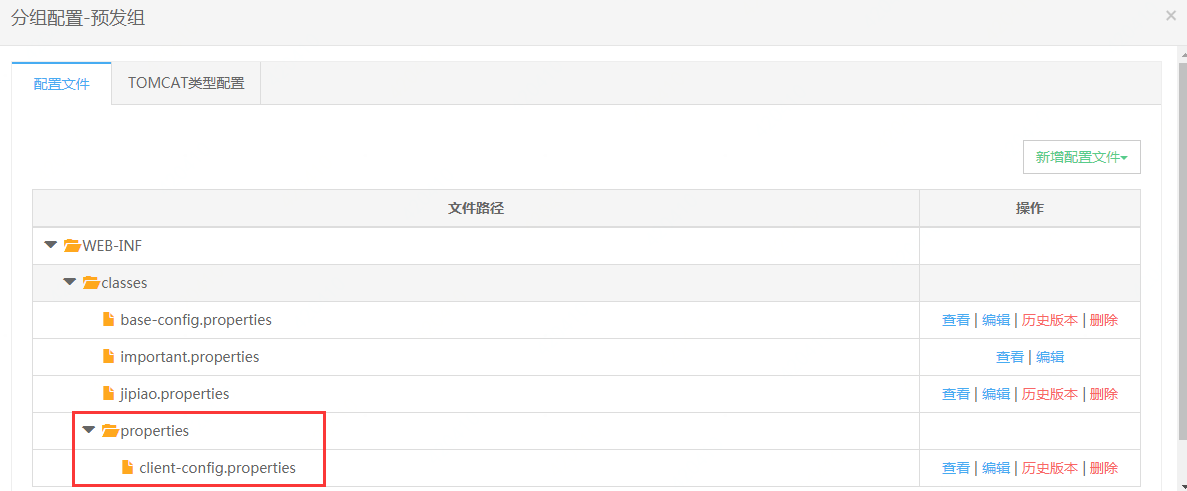


## 上线前

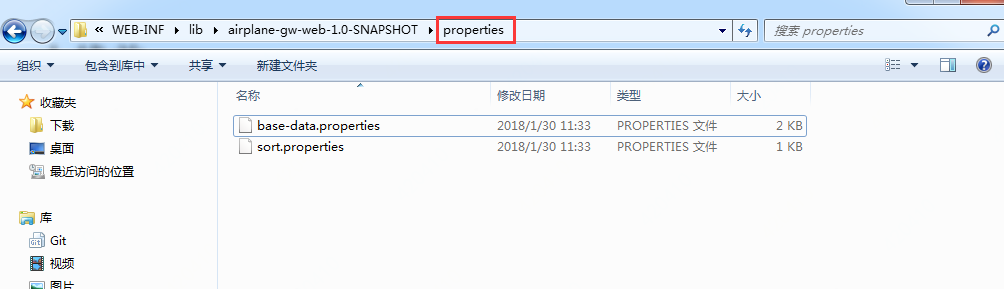
### Gw中部署预发失败：

原因：宾哥建立了一个proterties文件

会覆盖打包的本地文件夹，这样找不到base-data.properties文件



打包后的文件夹



### 白名单问题

使用userlist.contains(query.getUserPin())，当userPin为空时出现空指针异常。导致线上白名单不是all时，未登录用户查询不到航班。

改后，先判断null。

现在：all时所有用户都能看到，非all时只有白名单中能看到

