|  |  |  |  |
| --- | --- | --- | --- |
|  | **文 档 编 号** | **产品版本** | **密级** |
|  |  |  |
|  | | **共 页** |

清结算二期概要设计说明书

**北京瑞金麟网络技术服务有限公司**

**Rkylin Network Technology & Service Co. Ltd.**

**二○一五年十二月**

改 版 履 历 表

|  |  |  |  |
| --- | --- | --- | --- |
| **版数** | **发行年月** | **作成者** | **变 更 内 容 概 要** |
| V0.0 | 2015-11-05 | 孙锐斌 | 初版作成 |
| V1.0 | 2016-1-5 | 游宇 | 对文档做补充和完善 |

目录

[1．引言 4](#_Toc438485518)

[1.1编写目的 4](#_Toc438485519)

[1.2项目背景 4](#_Toc438485520)

[1.3定义 4](#_Toc438485521)

[1.4参考资料 4](#_Toc438485522)

[2．任务概述 5](#_Toc438485523)

[2.1目标 5](#_Toc438485524)

[2.2运行环境 6](#_Toc438485525)

[2.3需求概述 6](#_Toc438485526)

[2.4 条件与限制 6](#_Toc438485526)

[3．总体设计 7](#_Toc438485528)

[3.1处理流程 7](#_Toc438485529)

[3.2总体结构和模块外部设计 8](#_Toc438485530)

[3.3功能分配 8](#_Toc438485531)

[4．接口设计 13](#_Toc438485532)

[4.1外部接口 13](#_Toc438485533)

[4.2外部接口 13](#_Toc438485534)

[5．数据结构设计 13](#_Toc438485535)

[5.1表列表 13](#_Toc438485536)

[5.2物理结构设计 13](#_Toc438485537)

[5.3数据结构与程序的关系 15](#_Toc438485538)

[6．运行设计 17](#_Toc438485539)

[6.1运行模块的组合 17](#_Toc438485540)

[6.2运行控制 18](#_Toc438485541)

[6.3运行时间 18](#_Toc438485542)

[7．出错处理设计 18](#_Toc438485543)

[7.1出错输出信息 18](#_Toc438485544)

[7.2出错处理对策 18](#_Toc438485545)

[8.安全保密设计 18](#_Toc438485546)

[9.维护设计 18](#_Toc438485547)

[10.风险评估 18](#_Toc438485548)

[11.工作量评估 19](#_Toc438485549)

[11.附录： 20](#_Toc438485550)

# 1．引言

## 1.1编写目的

融数金服清结算系统加入代收付交易结算功能，使清结算系统功能更加完善，系统划分更加清晰。

## 1.2项目背景

在一期清结算系统上线稳定后，加入一期缺失的功能，完善现有功能，目前代收付交易结算功能由账户系统与代收付系统完成，二期清结算系统完成后将由清结算系统与代收付系统完成。

## 1.3定义

**清结算系统:**

**清算：**是指按一定的规则和制度安排对经济活动中形成的多重债权债务关系结清的过程，清算是一个核对过程，将多方的资金往来对账，核对，汇总，计算清楚，清算没有资金的转移，只对交易进行核对，计算，汇总，对账等动作；

**结算：**是将清算过程中产生的待结算债权债务，在收付款人金融机构之间进行账务处理、账簿记录，以完成资金最终转移的过程，结算就是清算后对资金进行最终的划拨，有资金的转移动作。

## 1.4参考资料

《瑞金麟金融中心账户系统接口V1.1》

# 2．任务概述

## 2.1目标

1. 完成清结算系统二期的研发、测试、上线。
2. 清结算后端功能修改点：
   1. 新增对代收付文件的导入、解析、数据加工、入库等逻辑；
   2. 新增代收付交易单汇总的逻辑,将机构号+用户ID+交易类型都相同的交易进行汇总后再发送给代收付(加开关控制)；
   3. 新增代收付结果处理接口，接收代收付的返回结果；
   4. 新建ActiveMQ队列QJS\_DSF\_QUEUE ；
   5. 新增对队列QJS\_DSF\_QUEUE的监听逻辑；
   6. 新增将代收付结果更新到交易表、结算表、汇总表的逻辑 ；
   7. 新增调用账户系统读取绑卡信息接口的逻辑；
   8. 新增调用订单系统读取绑卡信息接口的逻辑；
   9. 新增调用账户系统代收付结果通知接口的客户端逻辑；
   10. 新增退票接口，接收代收付系统发起的退票请求;
   11. 新增调用账户系统的退票接口的客户端逻辑;
3. 清结算前端功能修改点：
   1. 新增文件导入功能(机构直接提供文件，不通过账户系统)、查询功能、发送代收付等功能的画面。
   2. 新增对代收付失败的单子增加查询、分页、复选和重发等功能的画面；

说明：（目前会发生由于上游渠道的问题导致商户的代收付失败的情况，而商户侧会有今天必须代收，代付成功的要求，我方就需要在上游渠道调整完之后运营人员通过前端页面手动再次发送之前失败的那些代收，代付等交易），伴随代收付交易结算功能增加部分差错处理（提示：代收付交易如果失败了，默认重发一次，如果2次均失败了，则不再自动发送，这种单子就需要人工干预重发）。

新增两个输入框，手工输入条数、金额，跟选择的做比对；

* 1. 新增结算表的查询、分页、修改状态等功能画面；

## 2.2运行环境

略

## 2.3需求概述

之前代收付交易是由账户系统直接跟代收付系统交互，二期会将此功能转由清结算系统跟代收付系统交互完成。

二期清结算的代收付功能需要账户系统、订单系统、代收付系统的配合；

**之前：**

**二期清结算:**

## 2.4条件与限制

1. 由于这个功能之前由账户系统完成，所以二期清结算提测前，需要账户侧也要进行修改，同时提测。
2. 代收付系统做接口切换，对于代收付结果，由调用账户系统接口改为调用清结算系统接口；

# 3．总体设计

## 3.1处理流程

3.1.1 批量代收付和提现：



3.1.2 文件导入：



3.1.3退票：



## 3.2总体结构和模块外部设计



上图中清结算系统分为两个层级,即一级清算和二级清算,一级清算是指机构间的清算,即支付机构,瑞金麟金融产品支付平台和受理机构之间的数据清结算.二级清算是指受理机构与商户及分润主体之间的清结算。

## 3.3功能分配

**代收付交易后台处理:**

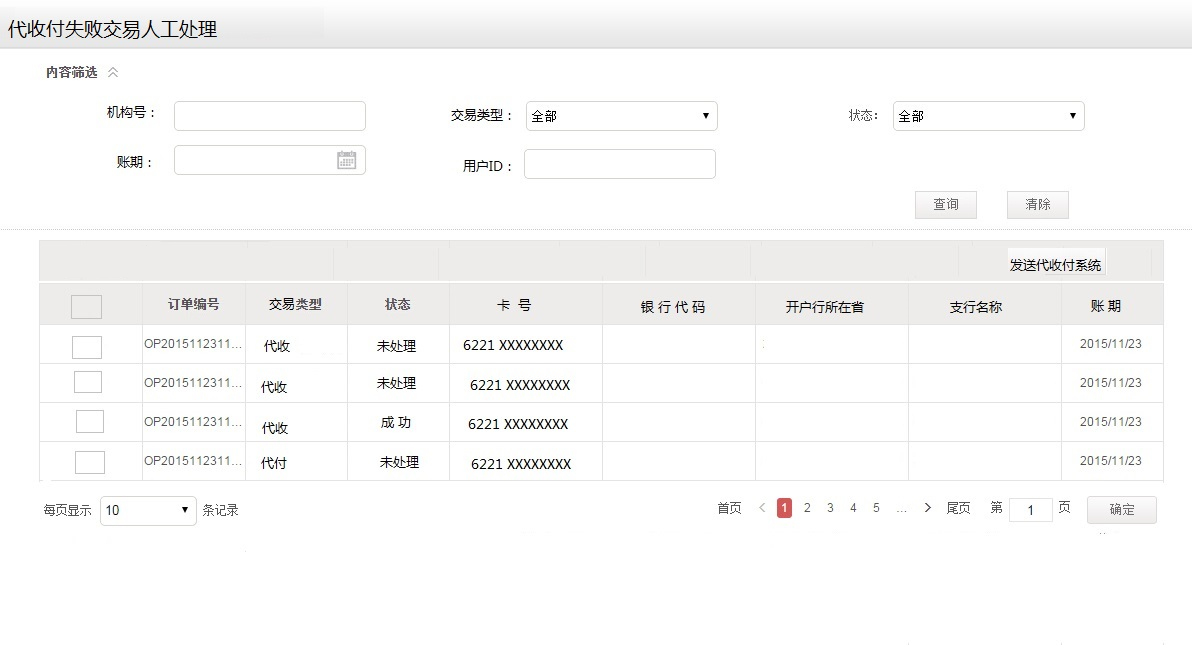
1. 新增对代收付文件的导入、解析、入库等逻辑；
2. 新增代收付交易单汇总的逻辑,将机构号+用户ID+交易类型都相同的交易进行汇总后再发送给代收付(加开关控制)；
3. 新增代收付结果处理接口，接收代收付的返回结果；
4. 新建ActiveMQ队列QJS\_DSF\_QUEUE ；
5. 新增对队列QJS\_DSF\_QUEUE的监听逻辑；
6. 新增将代收付结果更新到交易表、结算表、汇总表的逻辑 ；
7. 新增调用账户系统读取绑卡信息接口的逻辑；
8. 新增调用订单系统读取绑卡信息接口的逻辑；
9. 新增调用代收付接口的逻辑；
10. 新增调用账户系统代收付结果通知接口的客户端逻辑；
11. 新增退票接口，接收代收付系统发起的退票请求;
12. 新增调用账户系统的退票接口的客户端逻辑;

**画面:**

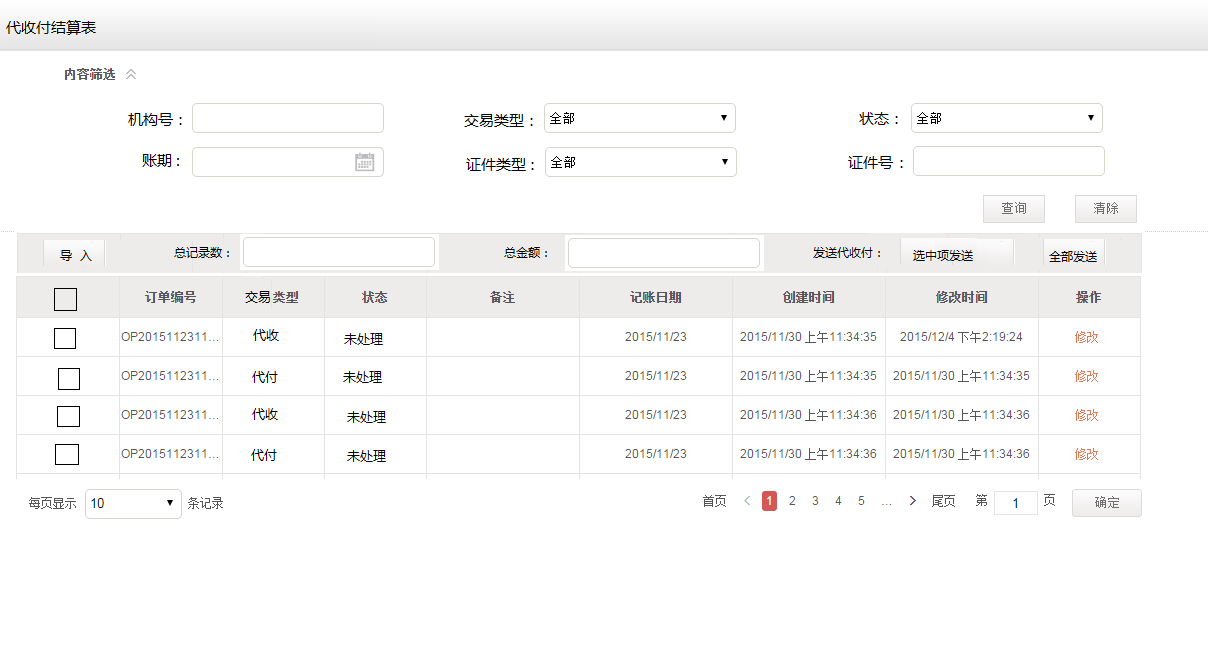
1. 对代收付失败的单子增加查询、分页、复选和重发等功能；

伴随代收付交易结算功能增加部分差错处理。

页面如下图所示：



1. 新增结算表的查询、分页、修改状态等功能，新增两个输入框，手工输入条数、金额，跟选择的做比对。页面如下图所示：

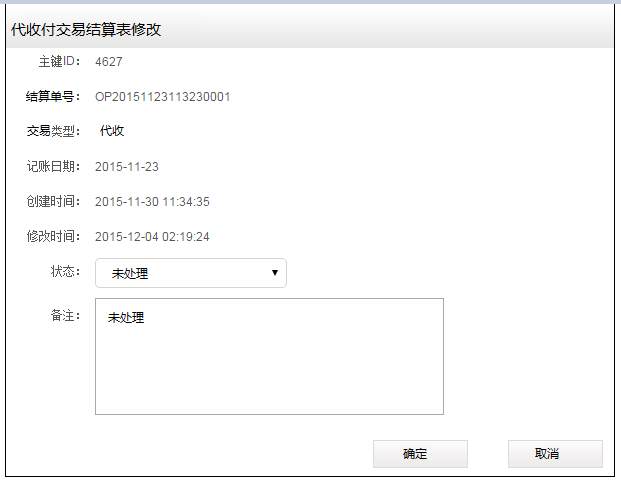


新增导入功能(机构直接提供文件，不通过账户系统)、查询功能、发送代收付等功能。清结算系统准备好Excel模板，运营人员根据模板格式填好信息后上传，清结算系统对文件解析、数据校验后写入结算表。

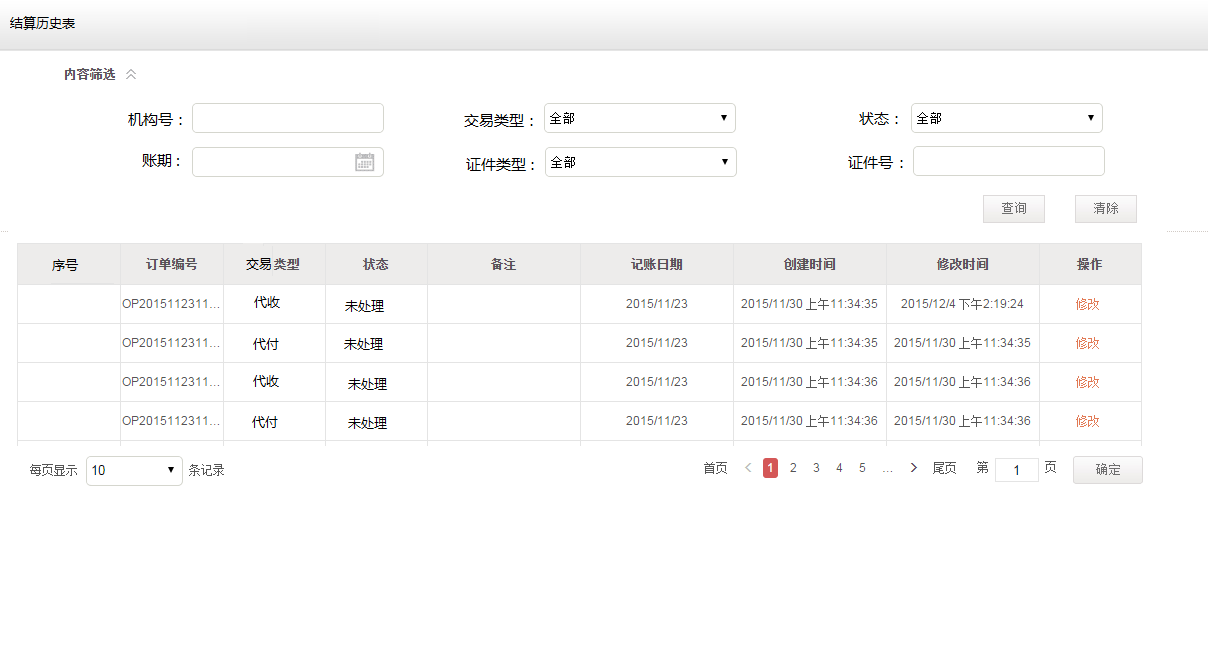
页面如下图所示：



修改状态：



1. 新增结算历史表的查询、分页等功能，画面如下图所示：



# 4．接口设计

## 4.1外部接口（标绿色的接口已经有了）

(1) 账户绑卡信息查询接口：账户系统提供此接口，清结算做客户端调用;

(2) 订单系统绑卡信息查询接口：订单系统提供此接口，清结算做客户端调用;

(3) 代收付批量接口：代收付系统提供此接口，清结算做客户端调用;

(4) 代收付批量结果接收接口：清结算做服务端，代收付做客户端;

(5) 代收付退票接口：清结算做服务端，代收付做客户端;

(6) 账户系统代收付结果处理接口：账户系统做服务端，提供一个代收付结果处理接口，清结算做客户端调用;

(7) 账户系统提供的退票接口：账户系统做服务端，清结算系统做客户端，将退票的交易进行反向充值;

## 4.2内部接口

(1) 解析导入文件的接口（提供给settle-web工程调用）；

(2) 查询结算表的接口（提供给settle-web工程调用）；

(3) 更改结算表状态的接口（提供给settle-web工程调用）；

## 5．数据结构设计

## 5.1表列表

|  |  |  |
| --- | --- | --- |
| 编号 | 表名 | 备注 |
| 1 | SETTLE\_TRANS\_DETAIL | 交易表（改动） |
| 2 | SETTLE\_TRANS\_SUMMARY | 汇总表（新增） |
| 3 | SETTLE\_TRANS\_SUMMARY\_HIS | 汇总历史表（新增） |
| 3 | SETTLE\_TRANS\_INVOICE | 结算单表（新增） |
| 3 | SETTLE\_TRANS\_INVOICE\_HIS（新增） | 结算单历史表（新增） |

## 5.2物理结构设计

本次开发中,数据库表为:

1. 交易表(SETTLE\_TRANS\_DETAIL)，新增字段DFLAG,用于标记代收付交易单状态。

|  |  |  |  |
| --- | --- | --- | --- |
| 编号 | 字段名 | 字段类型 | 备注 |
| 1 | DFLAG | tinyint | 代收付交易单状态：0未处理,1处理中,2成功，3失败，null非代收付交易单 |

1. 汇总表(SETTLE\_TRANS\_SUMMARY)，以纵表的形式存储订单号。

|  |  |  |  |
| --- | --- | --- | --- |
| 编号 | 字段名 | 字段类型 | 备注 |
| 1 | TRANS\_SUM\_ID | int | 汇总交易id |
| 2 | ROOT\_INST\_CD | varchar | 管理机构代码 |
| 3 | USER\_ID | varchar | 用户号 |
| 4 | ORDER\_TYPE | tinyint | 订单类型 |
| 5 | ACCOUNT\_DATE | datetime | 账期 |
| 6 | ORDER\_NO | varchar | 订单号 |
| 7 | BATCH\_NO | varchar | 批次号(跟结算单表的BATCH\_NO关联) |
| 8 | AMOUNT | decimal | 金额(分) |
| 9 | STATUS\_ID | tinyint | 状态, 0未处理,1处理中,2成功，3失败 |
| 10 | CREATED\_TIME | timestamp | 记录创建时间 |
| 11 | UPDATED\_TIME | timestamp | 记录更新时间 |

1. 汇总历史表(SETTLE\_TRANS\_SUMMARY\_HIS)。

|  |  |  |  |
| --- | --- | --- | --- |
| 编号 | 字段名 | 字段类型 | 备注 |
| 1 | TRANS\_SUM\_ID | int | 汇总交易id |
| 2 | ROOT\_INST\_CD | varchar | 管理机构代码 |
| 3 | USER\_ID | varchar | 用户号 |
| 4 | ORDER\_TYPE | tinyint | 订单类型 |
| 5 | ACCOUNT\_DATE | datetime | 账期 |
| 6 | ORDER\_NO | varchar | 订单号 |
| 7 | BATCH\_NO | varchar | 批次号(跟结算单表的BATCH\_NO关联) |
| 8 | AMOUNT | decimal | 金额(分) |
| 9 | STATUS\_ID | tinyint | 状态, 0未处理,1处理中,2成功，3失败 |
| 10 | CREATED\_TIME | timestamp | 记录创建时间 |
| 11 | UPDATED\_TIME | timestamp | 记录更新时间 |

1. 结算单表(SETTLE\_TRANS\_INVOICE)

|  |  |  |  |
| --- | --- | --- | --- |
| 编号 | 字段名 | 字段类型 | 备注 |
| 1 | INVOICE\_NO | bigint | 结算单号 |
| 2 | REQUEST\_NO | varchar | 交易批次号(上游) |
| 3 | BUSSINESS\_CODE | varchar | 业务代码 |
| 4 | ROOT\_INST\_CD | varchar | 管理机构代码 |
| 6 | ORDER\_TYPE | tinyint | 订单类型,代收、代付、提现等 |
| 7 | BATCH\_NO | varchar | 批次号 |
| 8 | GENE\_SEQ | varchar | 记录序号 |
| 9 | USER\_ID | varchar | 用户ID |
| 10 | BANK\_CODE | varchar | 银行代码 |
| 11 | ACCOUNT\_TYPE | varchar | 账户类型 |
| 12 | ACCOUNT\_NO | varchar | 账号 |
| 13 | ACCOUNT\_NAME | varchar | 账号名 |
| 14 | ACCOUNT\_PROPERTY | varchar | 账号属性 |
| 15 | PROVINCE | varchar | 开户行所在省 |
| 16 | CITY | varchar | 开户行所在市 |
| 17 | OPEN\_BANK\_NAME | varchar | 开户行名称 |
| 18 | PAY\_BANK\_CODE | varchar | 支付行号 |
| 19 | AMOUNT | decimal | 金额 |
| 20 | CURRENCY | varchar | 币种 |
| 21 | CERTIFICATE\_TYPE | varchar | 开户证件类型 |
| 22 | CERTIFICATE\_NUMBER | varchar | 证件号 |
|  | DATA\_SOURCE | tinyint | 数据来源：0是文件导入，1系统推送，2手工重付 |
| 23 | PROCESS\_RESULT | varchar | 处理结果 |
| 24 | SEND\_TYPE | tinyint | 发送类型,0正常,1代扣失败,2代扣延迟 |
| 25 | ERROR\_CODE | varchar | 错误码 |
| 26 | SEND\_TIMES | tinyint | 发送次数：如果已经达到2次就不再自动发送，画面部分人工干预重发 |
| 27 | STATUS\_ID | tinyint | 状态： 0未处理,1处理中,2成功，3失败 |
| 28 | ACCOUNT\_DATE | datetime | 记账日期 |
| 29 | REAL\_TIME\_FLAG | tinyint | 实时标识 ：0否，1是 |
| 30 | REMARK | varchar | 用户备注 |
| 31 | CREATED\_TIME | timestamp | 记录创建时间 |
| 32 | UPDATED\_TIME | timestamp | 记录更新时间 |

说明：结算表(SETTLE\_TRANS\_INVOICE) 与汇总表（SETTLE\_TRANS\_SUMMARY）是一对多的关系，结算表中的一个BATCH\_NO值对应汇总表中的多笔订单。

1. 结算单历史表(SETTLE\_TRANS\_INVOICE\_HIS)

|  |  |  |  |
| --- | --- | --- | --- |
| 编号 | 字段名 | 字段类型 | 备注 |
| 1 | INVOICE\_NO | bigint | 结算单号 |
| 2 | REQUEST\_NO | varchar | 交易批次号(上游) |
| 3 | BUSSINESS\_CODE | varchar | 业务代码 |
| 4 | ROOT\_INST\_CD | varchar | 管理机构代码 |
| 6 | ORDER\_TYPE | tinyint | 订单类型,代收、代付、提现等 |
| 7 | BATCH\_NO | varchar | 批次号 |
| 8 | GENE\_SEQ | varchar | 记录序号 |
| 9 | USER\_ID | varchar | 用户ID |
| 10 | BANK\_CODE | varchar | 银行代码 |
| 11 | ACCOUNT\_TYPE | varchar | 账户类型 |
| 12 | ACCOUNT\_NO | varchar | 账号 |
| 13 | ACCOUNT\_NAME | varchar | 账号名 |
| 14 | ACCOUNT\_PROPERTY | varchar | 账号属性 |
| 15 | PROVINCE | varchar | 开户行所在省 |
| 16 | CITY | varchar | 开户行所在市 |
| 17 | OPEN\_BANK\_NAME | varchar | 开户行名称 |
| 18 | PAY\_BANK\_CODE | varchar | 支付行号 |
| 19 | AMOUNT | decimal | 金额 |
| 20 | CURRENCY | varchar | 币种 |
| 21 | CERTIFICATE\_TYPE | varchar | 开户证件类型 |
| 22 | CERTIFICATE\_NUMBER | varchar | 证件号 |
|  | DATA\_SOURCE | tinyint | 数据来源：0是文件导入，1系统推送，2手工重付 |
| 23 | PROCESS\_RESULT | varchar | 处理结果 |
| 24 | SEND\_TYPE | tinyint | 发送类型,0正常,1代扣失败,2代扣延迟 |
| 25 | ERROR\_CODE | varchar | 错误码 |
| 26 | SEND\_TIMES | tinyint | 发送次数：如果已经达到2次就不再自动发送，画面部分人工干预重发 |
| 27 | STATUS\_ID | tinyint | 状态： 0未处理,1处理中,2成功，3失败 |
| 28 | ACCOUNT\_DATE | datetime | 记账日期 |
| 29 | REAL\_TIME\_FLAG | tinyint | 实时标识 ：0否，1是 |
| 30 | REMARK | varchar | 用户备注 |
| 31 | CREATED\_TIME | timestamp | 记录创建时间 |
| 32 | UPDATED\_TIME | timestamp | 记录更新时间 |

## 5.3数据结构与程序的关系

(1) 定时任务定时将交易表(SETTLE\_TRANS\_DETAIL)中DFLAG值为0(未处理的批量代收付交易或提现交易)的数据查询出来，管理机构代码+用户ID+订单类型+账期相同的订单打上相同的批次号，写入汇总表(SETTLE\_TRANS\_SUMMARY)。

（2）定时任务将汇总表(SETTLE\_TRANS\_SUMMARY)中状态是未处理的数据查询出来，同时调用账户系统的接口查询相关的绑卡信息（如果是提现，则优先从订单系统查询绑卡信息，没查到再从账户系统查），然后将二者的信息组织起来插入结算表(SETTLE\_TRANS\_INVOICE)中。

（3）定时任务扫描结算表(SETTLE\_TRANS\_INVOICE)中状态是未处理的且数据来源是系统推送的批量代收、批量代付或提现的单子,机构号相同的打上同一个交易批次号，每次将同一个交易批次号的数据，通过dubbo调用代收付系统的批量接口，将数据发送给代收付系统。

（4）清结算系统提供接口,接收代收付系统的的返回结果，并将结果放入ActiveMQ的队列QJS\_DSF\_QUEUE,清结算系统监听队列QJS\_DSF\_QUEUE ，取得代收付的处理结果，根据结果信息更新结算表(SETTLE\_TRANS\_INVOICE)的状态、更新汇总表(SETTLE\_TRANS\_SUMMARY)的状态、和交易表(SETTLE\_TRANS\_DETAIL)的DFLAG字段的值，注意：如果代收付返回结算表中的某条记录处理失败，则要将结算表的该条记录的批次号(BATCH\_NO)对应的汇总表(SETTLE\_TRANS\_SUMMARY)的所有数据和交易表(SETTLE\_TRANS\_DETAIL)的DFLAG字段的值都改为处理失败。对结算表和汇总表中成功的数据移入历史表。再将结果通知账户系统。



# 6．运行设计

## 6.1运行模块的组合

无

## 6.2运行控制

## 6.3运行时间

# 7．出错处理设计

## 7.1出错输出信息

## 7.2出错处理对策

（1）容错处理：如果代收付系统返回处理结果时，同一条信息返回了多次，清结算系统只处理一次，其余的不做处理。

（2）人工干预：如果是因为一些原因造成单子有问题，可以通过清结算画面进行人工干预，如：可以改单子的状态等，这笔单子就不会进行后续的处理；

（3）如果清结算系统宕机了，会导致代收付系统无法将结果返回，建议代收付系统提供一个代收付结果查询接口，清结算系统可以随时查询代收付结果（因为代收付系统已经有画面可以重发，优先级低）;

(4) 如果清结算系统调用代收付系统失败了，会再请求一次，如果两次都失败了，则不再发送，由人工干预重发;

# 8.安全保密设计

# 9.维护设计

略

# 10.风险评估

（1）清结算二期的改造进度受其他系统（如：账户系统、订单系统、代收付系统）的影响，尤其是接口联调;

# 11.工作量评估

|  |  |  |  |
| --- | --- | --- | --- |
| **工作概要** | **改造点** | **工作量（人日）** | **备注** |
| 准备 | 需求调研+概要设计 | 10 |  |
| 后台改造（开发+自测） | 新增对代收付文件的导入、解析、数据加工、入库等逻辑 | 2 |  |
| 新增代收付交易单汇总的逻辑,将机构号+用户ID+交易类型都相同的交易进行汇总后再发送给代收付 | 1 |  |
| 新增代收付结果处理接口，接收代收付的返回结果并写入MQ队列QJS\_DSF\_QUEUE | 2 |  |
| 新增对队列QJS\_DSF\_QUEUE的监听逻辑 | 1 |  |
| 新增将代收付结果更新到交易表、结算表、汇总表的逻辑 | 1 |  |
| 新增调用账户系统读取绑卡信息接口的逻辑 | 3 |  |
| 新增调用账户系统操作热点账户接口的逻辑 |
| 新增调用订单系统读取绑卡信息接口的逻辑； |
| 新增调用账户系统代收付结果通知接口的客户端逻辑 |
| 代收付系统提供接口，清算做客户端可以主动查询代收付结果(应对异常情况) | 1 |  |
| 新增退票接口，接收代收付系统发起的退票请求 | 2 |  |
| 新增调用账户系统的退票接口的客户端逻辑 | 1 |  |
| 前端改造（开发+自测） | 新增文件导入功能(机构直接提供文件，不通过账户系统)、查询功能、发送代收付等功能 | 2 |  |
| 对代收付失败的单子增加查询、分页、复选和重发等功能； | 2 |  |
| 新增结算表的查询、分页、修改状态等功能 | 2 | 优先级低 |
| **合计** |  | 30 |  |

# 12.建议：

(1) 画面部分可以考虑引入成熟的UI框架(如：ligerui、easyui等)，既可以增加画面部分的美观程度，同时可以提高开发效率；

# 13.附录：

（1）．发代收付系统的时候，机构号要做转换：

（a）课栈的机构号要转成丰年的机构号；

（b）棉庄和食全食美的机构号要转成融数的机构号;