# 交互中心维护说明书

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 项 目 号 |  | | | |
| 文档编号 |  | | | |
| 工程编号 |  | | | |
| 版 本 号 |  | | | |
| 保密级别 | 一般 | 内部公开 | 秘密 | 机密 |

修订记录

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 日期 | 作者 | 版本 | 修改原因 | 主要修改内容 |
| 2017/1/6 | Clark | 1.00 | 初始创建 |  |
| 2017/2/3 | Clark | 1.01 | 完善内容 | 对部分措辞就行修改 |
|  |  |  |  |  |

目录

[交互中心维护说明书 1](#_Toc473885049)

[1.交互中心职责 5](#_Toc473885050)

[2.交互中心功能 5](#_Toc473885051)

[2.1外部业务系统开发人员 5](#_Toc473885052)

[2.2交互中心开发人员 7](#_Toc473885053)

## 1.交互中心职责

作为欧冶金融结算系统的子系统，主要用于承接不同的业务系统的业务数据，然后再转化为结算系统能够处理的结算单据。为了应对种类繁多的业务系统，交互中心采用类似模版的思想，构建灵活的基于数据库配置的纯配置型交互中间件。

## 2.交互中心功能

功能一、承接业务数据。

功能二、转换业务数据。

### 2.1外部业务系统开发人员

承接外部系统数据的时候，需要我们的不同的业务系统严格按照交互总心的数据传递规则。

首先，交互中心采用的交互方式是http restful的交互模式。传递的数据格式为json格式。具体的交互接口地址，由交互中心配置好后告知业务系统。业务系统需要将待传递的数据表内容告知交互中心开发人员。交互中心开发人员会根据业务系统表，在交互中心建立落地表，用于存储业务系统发送的json数据。

例1（业务系统单表传递）：如果业务系统A想要传递待结算的业务数据（对应的表为 A\_BUSI）到交互中心，然后交由结算系统去结算。 A\_BUSI 的表结构如下：

Name Null Type

---------------- -------- -------------

SERIALNO NOT NULL VARCHAR2(20)

TYPE VARCHAR2(2)

FRAMEID VARCHAR2(50)

FINANCEID VARCHAR2(50)

REDEEMID VARCHAR2(50)

CUSTOMERID VARCHAR2(50)

CUSTOMERNAME VARCHAR2(100)

CUSTOMERBANKNAME VARCHAR2(100)

CUSTOMERACCNO VARCHAR2(50)

CUSTOMERBANKNO VARCHAR2(50)

BANKID VARCHAR2(50)

BANKNAME VARCHAR2(100)

BANKBANKNAME VARCHAR2(100)

GOODSAMTUPDATE VARCHAR2(50)

PANELADDRESS VARCHAR2(50)

FINANCETYPE VARCHAR2(10)

FINANCEPRODUCT VARCHAR2(20)

CCY VARCHAR2(10)

FINANCEMONEY NUMBER(20,2)

CAPITAL NUMBER(20,2)

INTERSET NUMBER(20,2)

SPFEE NUMBER(20,2)

PAYDATE DATE

EXPFLAG VARCHAR2(10)

PAYTYPE VARCHAR2(10)

PROCESSMODE VARCHAR2(10)

OFFSETFLAG VARCHAR2(10)

TURNONTIME DATE

DOCDATE VARCHAR2(36)

RATE NUMBER

BILLENDDATE VARCHAR2(36)

RELATRIONID VARCHAR2(50)

ISHANDLE CHAR(1)

外部业务系统需要将这张表的表结构传给交互中心开发人员。

那么传递的json数据应该为:

{

“SERIALNO” :1 ,

“TYPE” :”2”,

“FRAMEID”: “123456”,

……

“ISHANDLE “ : “12”

}

当然也可以传递List:

[

{

“SERIALNO” :1 ,

“TYPE” :”2”,

“FRAMEID”: “123456”,

……

“ISHANDLE “ : “y”

},

{

“SERIALNO” :2 ,

“TYPE” :”3”,

“FRAMEID”: “1234567”,

……

“ISHANDLE “ : “n”

}

]

注意交互中心允许传递的JSON数据只有表格中的部分字段。但是不允许传递的JSON，Key不存在给的表中的。如果业务方需要添加字段，请告知交互中心开发人员，添加相应的字段到交互中心表中。

例2（多表）：如果业务单位传递的业务单据不是存在于单表中，而是存在于多张表中，目前支持的场景是一张头表，一张明细表。如果是这样的情形。业务方同样的需将两张表的表结构给予交互中心开发人员。规则和例1一样。但是注意传递的JSON数据有些不同。

例如：如果业务系统B 有两个表其中头表（B\_H）有字段 Hid 和name 。 明细表（B\_I）有字段Iid 和name， 那么业务方需传递的json应该为

{

"Hid": 1,

"name": "clark",

"items":

[

{

"Iid": 1,

"name": "xiaoming"

},

{

"Iid": 2,

"name": "xiaoming"

}

]

}

1警告：注意明细表的key值的内容必须为items。如果有特殊需求不能写为items的，请告知交互中心开发人员，交互中心开发人员会特殊为你定义一个key值。

至此，业务方需要注意的开发事项全部交代完毕。

### 2.2交互中心开发人员

#### 2.2.1 承接业务数据

首先，交互中心开发人员在拿到业务系统给出的表结构后，需要在这张表中添加一个新的字段用于标志当前这条数据是否已经传递至结算系统。Alter table A\_BUSI add(TO\_CA number default 0); 0 表示未传递，1表示已经传递。

然后， 交互中心数据库中有几张核心表，下面会详细说明，首先要说的是IC\_CFG\_DISPATCHER。

Name Null Type

-------------- -------- -------------

CLZ NOT NULL VARCHAR2(100)

METHOD NOT NULL VARCHAR2(100)

TABLENAME NOT NULL VARCHAR2(200)

CHECKALLCOLUMN NUMBER

PKCOLUMNNAME VARCHAR2(200)

CONDITION NOT NULL VARCHAR2(200)

COMPLETION NOT NULL VARCHAR2(200)

EXPONENT VARCHAR2(200)

对接外部业务系统的时候我们需要在这张表格中配置业务系统传递的URL。地址的格式为[http://ip:port/interaction-center/system/{CLZ}/{METHOD}](http://ip:port/interaction-center/system/%7BCLZ%7D/%7BMETHOD%7D) （最后的两个字段的值是我们根据具体的业务性质填写的）

通常情况如果业务系统单表传递数据。则我们只需要在这张表IC\_CFG\_DISPATCHER中添加一条数据即可。填写好具体的内容后开发人员需要将具体的URL接口地址给予对应的业务系统开发人员。并且进行测试校验数据。

例1：首先我们需要对这个表的每个字段进行一下说明：前两个字段用于配置URL路径。TABLENAME对应的业务系统的表在交互中心的名字。CHECKALLCOLUMN，是否对每个列的值以及类型进行校验。默认是进行校验。PKCOLUMNNAME，表对应的主键，如果是联合主键，用“;”进行隔开。CONDITION，表示状态标示，未完成状态用0表示，已传递状态用1标示，EXPONENT，对当前数据的备注。一般来说，后面的几个参数适用默认值即可。

现在要给A系统配置接口则为 :

insert into IC\_CFG\_DISPATCHER values(“A”,”BUSI”,”IC\_A\_BUSI”,” 1”,”SEREALNO”,” to\_ca=0”,” to\_ca=1”,”A 系统发结算接口”)

然后将<http://ip:port/interaction-center/system/A/BUSI> 发送给A系统开发人员即可。

警告：如果业务单位表没有主键（极少的情况下）我们可以在application.properties文件中进行配置。 NO\_PK\_VALID\_TAB=IC\_A\_BUSI

例2：但是如果业务系统进行多表传递的时候，我们就需要配置多条数据到IC\_CFG\_DISPATCHER中。在多表配置IC\_CFG\_DISPATCHER的时候头的 CLS 必须配置为“list” 并且 明细的CLS必须为头的METHOD。所以配置应该是这样的。

insert into IC\_CFG\_DISPATCHER values(“list”,”B\_H”,”IC\_B\_H”,” 1”,”HiD”,” to\_ca=0”,” to\_ca=1”,”B系统发结算接口”);

insert into IC\_CFG\_DISPATCHER values(“B\_H”,”B\_I”,”IC\_B\_I”,” 1”,”IiD”,” to\_ca=0”,” to\_ca=1”,”B系统发结算接口”)

警告：上面在业务系统开发人员那一节中，我们说明了，如果多表数据发送的时候，使用items作为存储子表的key值，但是如果特殊情况不能用items 作为key那么，开发人员可以通过application.properties中的NODE\_NAME进行配置。然后告诉开发人员。

#### 2.2.2 转换业务数据

至此我们就将外部系统数据全部导入到交互中心。下面我们要做的是将交互中心的业务数据转化为结算系统需要的结算单据。这里面涉及到，我们交互中心的另外几张核心表。IC\_CFG\_MAPTOCA IC\_CFG\_ONETOMANY;

首先我们重点来说明目前，交互中心转到结算系统，有几种方式，是可以进行灵活配置的。

方式1：单表数据对应结算一条单据，一条头数据，一条明细数据。

方式2：单表数据对应两条结算单据，分别拆分为两个单据类型的单据。分别含有一条头数据，一条明细数据。

方式3：多表数据对应产生一条单据，一条头数据，多条明细数据。

详解方式1:

下面我来详细的解释一下IC\_CFG\_MAPTOCA表。

Name Null Type

------------- -------- --------------

CATABLENAME NOT NULL VARCHAR2(200)

CACOLUMNNAME NOT NULL VARCHAR2(200)

ICTABLENAME NOT NULL VARCHAR2(200)

ICCOLUMNVALUE VARCHAR2(1280)

EXPONENT VARCHAR2(200)

* CATABLENAME 对应的结算系统中单据表名字 固定CA\_DOCSCONTENT\_HEADER，和CA\_DOCSCONTENT\_ITEM
* CACOLUMNNAME 为CA\_DOCSCONTENT\_HEADER 对应的列的名字
* ICTABLENAME交互中心要转化的表的名字例如上面中的IC\_A\_BUSI
* ICCOLUMNVALUE为IC\_A\_BUSI 表如何获取到CACOLUMNNAME 对应的值。 注意这个是可编程式的，也就是可以写sql的 包括sql片段， 存储过程，存储函数等等。
* EXPONENT为当前数据注解。

配置完成后，可以通过调用接口 <http://ip:port/interaction-center/system/test> 将交互中心数据导入到结算系统中。注意这个是一个POST请求，可以使用curl命令进行测试。

详解方式2:

和方式1类似。只不过这个时候我们需要对另外的一个配置文件进行更近一步的配置。因为这个时候是将一个单据类型拆分为另外一种单据类型。所以我们需要进行配置。IC\_CFG\_ONETOMANY

Name Null Type

---------- -------- -------------

RULENO NOT NULL NUMBER(38)

DOCTYPEIDS VARCHAR2(128)

REMARK VARCHAR2(128)

* RULENO表示规则编号。标记规则的顺序。
* DOCTYPEIDS 需要1拆2的单据号，用“，”进行分割。
* REMARK 注解。

然后在application.properties 中配置单据拆分的单据类型匹配。例如211单据类型和222的单据类型相匹配。则应该用211=222进行配置 并且在IC\_CFG\_ONETOMANY 中配置一条规则。Insert into IC\_CFG\_ONETOMANY values(10,211,”….”);最后我们需要写一个实现类实现交互中心定好的规则接口。ARuleService， 在这个实现类中我们可以具体的对1对多的一些字段的改变，进行处理。最后的最后，我们需要将我们的实现类的全限定名配置到application.properties中的 ruleImplClass字段上。具体的可以参照已经完成的例子。

详解方式3:

和方式1类似，只是配置不同。首先这个配置中CATABLENAME 当他的值为CA\_DOCSCONTENT\_HEADER时，它的配置也只能从业务头表中的字段进行编程式配置 。CA\_DOCSCONTENT\_ITEM时，它的配置也只能从业务对应的明细表中的字段进行编程式配置 。 如果对这句话不是很理解，可以看看已经做过的一些配置。看看多表时如何配置的。然后需要我们进行application.properties的内容进行配置。

IC\_B\_H=IC\_B\_I最后在CONTINUE\_TAB键中配置值IC\_B\_I

至此我们完成了对交互中心最重要的内容的解析。当然虽然交互中心写的很灵活。可以满足我们百分之90的业务场景的需求。但是需求永远在改变，如果交互中心的逻辑不再满足需求那么就需要新的开发者，在原有功能代码的基础上进行开发。