Rays's

Personal Study Note

of

Spring

目 录

[1. 引言 4](#_Toc2848782)

[1.1. 文档说明 4](#_Toc2848783)

[2. 概述 5](#_Toc2848784)

[2.1. 现状 5](#_Toc2848785)

[2.2. 优化思路 5](#_Toc2848786)

[3. 模块接口设计 6](#_Toc2848787)

[3.1. 接口规划 6](#_Toc2848788)

[3.2. JSON格式 6](#_Toc2848789)

[4. 数据模型设计 7](#_Toc2848790)

[4.1. 表结构设计 7](#_Toc2848791)

[4.1.1. 历史表 TI\_F\_SYNC\_TRADE\_HIS 7](#_Toc2848792)

[5. CRM侧模块设计 8](#_Toc2848793)

[5.1. 模块概述 8](#_Toc2848794)

[5.2. 处理流程 8](#_Toc2848795)

[5.3. 依赖配置 8](#_Toc2848796)

[5.4. 异常设计 8](#_Toc2848797)

[6. 实时订购模块设计 9](#_Toc2848798)

[6.1. 模块概述 9](#_Toc2848799)

[6.2. 处理流程 9](#_Toc2848800)

[6.3. 开发设计 9](#_Toc2848801)

[6.3.1. 服务端 10](#_Toc2848802)

[6.3.2. DataTrans 10](#_Toc2848803)

[6.3.3. ITable 10](#_Toc2848804)

[6.3.4. MDB上发 10](#_Toc2848805)

[6.4. 依赖配置 11](#_Toc2848806)

[6.5. 异常设计 11](#_Toc2848807)

# 装配Bean

## 自动化装配

自动化装配的两大要素

* 组件扫描（Component Scan）：自动发现Bean
* 自动装配（Autowiring）：自动处理Bean之间的关系

1.组件扫描

@ComponentScan或者<context:component-scan> 开启自动扫描功能。

使用 @Component 注解标记需要的Bean对应的类（或者使用JDI规范@Named）

2.自动装配

@Autowired 注解表明Bean和其他Bean的依赖关系，可以用于构造函数、set方法和直接用于属性（或者使用JDI规范@Inject）。

## 代码配置

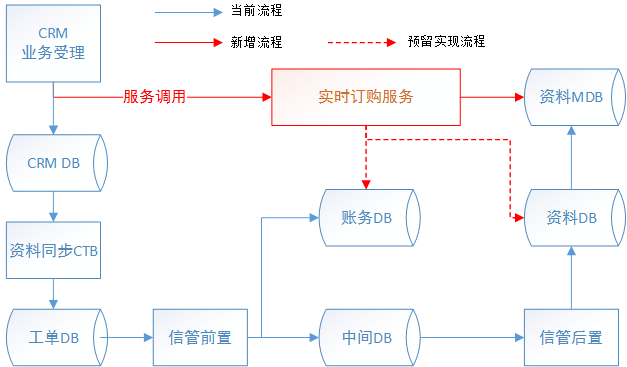
# 概述

## 现状

由于CRM到BOOS的资料同步存在大量异步过程，同步速度较慢。有时用户订购了时效性比较短的优惠，还没有享受到该优惠的内容而优惠已经到期了的情况。或者用户订购了资费优惠，但资料还未同步，用户就已经使用了大量资源，导致计费收取高额费用。本文档针对这些情况给出了解决方案：当crm前台登记业务时，新增一个过滤的配置，调取信管侧的实时接口，解决资料同步不及时的问题。

## 优化思路

信息管理增加一个模块，加速资料的同步，并对外提供实时订购服务接口。CRM侧调用服务接口，接口入参为订购资料信息，信管侧实时订购模块将入参解析为工单，之后进过前置和后置的逻辑转换为资料，然后增量入MDB。程序流程如下：



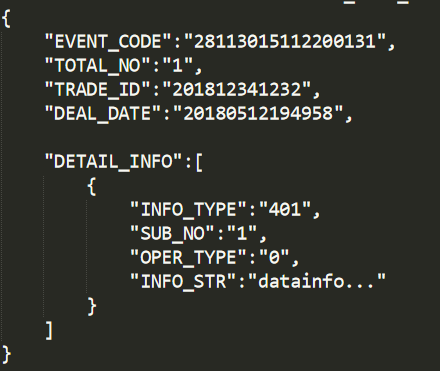
# 模块接口设计

## 接口规划

接口目前规定如下，具体实现通过json字符串保存，然后通过报文方式传递。

|  |  |  |
| --- | --- | --- |
| 接口参数 | 明细含义 | 明细 |
| EVENT\_CODE | 事务号 | 序列规则与C2B增量规则一致，不能重复，在一天之内序列要保证升序排列，原则上是18位，格式如下：  前十位表示时间，精确到分，年以18开始；  后面的6位数为序列  倒数第二位代表数据类型，实时订购服务接口约定为3  最后一位代表数据的库别信息 |
| TRADE\_ID | 业务工单号 | CRM业务工单号 |
| DEAL\_TIME | 业务办理时间 | CRM业务办理时间 |
| SUB\_NO | 子序号 | 编号1 对应上面的个数，也用于标记明细位置，原则上接口数据的顺序即为处理顺序即为子序号顺序 |
| TOTAL\_NO | 总数 | 该工单需要处理的资料条数 |
| INFO\_TYPE | 接口号 | 接口ID 与C2B接口保持一致 |
| OPER\_METHOD | 操作方式 | 0 新增，1 删除，2 更新 |
| INFO\_STR | 资料字符串 | 和C2B接口一致 |

## JSON格式



CRM封装到报文中的，JSON的格式如图，必须包含每个字段。

# 数据模型设计

## 表结构设计

### 历史表 TI\_F\_SYNC\_TRADE\_HIS

实时订购模块的历史表设计如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 类型 | 主键 | 索引 | 注释 |
| EVENT\_CODE | NUMBRE(18) |  | 是 | 事务号 |
| TRADE\_ID | NUMBER(16) |  | 是 | 业务工单号 |
| DEAL\_TIME | DATE |  |  | 业务办理时间 |
| SUB\_NO | NUMBRE(2) |  |  | 子序号 |
| TOTAL\_NO | NUMBER(2) |  |  | 子工单数 |
| INFO\_TYPE | NUMBER(5) |  |  | 接口ID |
| OPER\_METHOD | NUMBER(1) |  |  | 操作方式 |
| INFO\_STR | VARCHAR2(1000) |  |  | 资料字符串 |
| DONE\_TAG | NUMBER(2) |  |  | 标记 |
| DONE\_TIME | DATE |  |  | 完工时间 |
| ERROR\_CODE | VARCHAR2(10) |  |  | 错误码 |
| ERROR\_MSG | VARCHAR2(2000) |  |  | 错误信息 |

# CRM侧模块设计

## 模块概述

在前台业务办理登记流程的最后，新加一个过滤的配置来处理一些时效性比较高的优惠，在用户订购了该优惠时，直接调用实时接口，把资料送给信息管理，信息管理直接入MDB

## 处理流程



## 依赖配置

需要在过滤的配置参数表TD\_B\_PARAM\_FILTER配置一条过滤的数据。该配置主要功能是在业务登记最后执行该配置的类，完成实时接口的调用，把资料送给信息管理。

## 异常设计

无论调用实时接口返回是否异常，都需要把它catch住，不能影响正常的业务流程。

# 实时订购模块设计

## 模块概述

模块横跨信息管理原有两大流程（datatrans、itable）,将其中的核心业务逻辑抽离，跳过了分发逻辑，同时将原本的异步流程整合成为同步的流程。

## 处理流程



模块整体四个部分：服务端、datatrans、itable和mdb上发

服务端：负责接受远程调用并解析为工单，将工单记录后传递到datatrans。

DataTrans：根据原信管的dataTrans流程的配置和业务逻辑，将工单初步处理为中间数据结构，之后将中间数据传递给itable。

ITable：ITable按照原信管ITable流程中的接口Handler逻辑处理接口的中间逻辑，将中间数据解析为物理库实体。处理完毕后将所有实体缓存，并传递给MDB服务对象。

MDB上发：将 物理库实体转换为内存库的数据格式，然后根据资料的特征值求取路由，不同的资料上发到不同的内存库。

## 开发设计

如上一节所述，本模块需要新建四个部分的逻辑和代码：

1.新增服务端业务，所有代码需要新建

2.新建DataTrans业务，复用部分原DataTrans流程的核心业务处理逻辑和配置

3.新建ITable业务，复用部分原ITable逻辑

4.新建MDB上发业务，服用部分ITable逻辑和配置

### 服务端

1.服务端采用WebService技术，客户端服务端之间通过HTTP协议发送json或xml格式的文本交互。

2.服务端通过外公开服务地址和约定的接口对外服务

3.服务端调用其它业务对象来解析参数，将参数转换为工单对象，保存数据库并专递给后续模块。

### DataTrans

1.DataTrans需要在原ims\_datatrans流程的代码基础上全面改造为新逻辑，只复用核心逻辑。首先DataTrans需要将新建工单解析代码，将工单中的infoStr解析为接口表模型。之后复用ims\_datatrans的转换逻辑，将接口表模型转换为中间表模型。转换完毕后，原有的插入数据库等操作需要改造为传递到后续模块。

2.DataTrans使用原流程的配置表:

UCR\_PARAM.TD\_B\_I\_TABLE\_DEF

UCR\_PARAM.TD\_B\_I\_FIELD\_DEF

UCR\_PARAM.TD\_B\_TRANS\_TABLE\_DEF

UCR\_PARAM.TD\_B\_TRANS \_FIELD\_DEF

UCR\_PARAM.TD\_B\_O\_TABLE\_DEF

UCR\_PARAM.TD\_B\_O\_FIELD\_DEF

来解析工单，这些表中定义了工单到中间数据的转换方式。

### ITable

1.ITable需要建立全新的逻辑，只复用原流程的核心业务逻辑（Handler类）。ITable需要新建获取中间表模型的代码，取消原流程从DB获取中间表模型的逻辑。

2.ITable复用部分业务处理逻辑，将中间表模型转换为资料表模型，取消原业务逻辑中资料上发物理库逻辑，只将资料包模型放至缓存，供后续模块使用。

### MDB上发

1.MDB上发建立新的服务对象来整合MDBJar包提供的接口，通过服务对象调用MDB接口来处理资料表模型。首先改造原有流程的业务逻辑，将资料表模型进行加工分类，准备上发MDB。

2. MDB上发，复用ITable的配置来求取路由。

## 依赖配置

1.需要能够查询信息管理前置配置的数据库参数表：

UCR\_PARAM.TD\_B\_I\_TABLE\_DEF

UCR\_PARAM.TD\_B\_I\_FIELD\_DEF

UCR\_PARAM.TD\_B\_TRANS\_TABLE\_DEF

UCR\_PARAM.TD\_B\_TRANS \_FIELD\_DEF

UCR\_PARAM.TD\_B\_O\_TABLE\_DEF

UCR\_PARAM.TD\_B\_O\_FIELD\_DEF

2.需要ID用户下的配置表，用于求取路由和分表

## 异常设计

调用接口成功的正常返回消息，调用失败则返回错误编码。之后跟新工单表中的记录。

暂定的返回值：

***UNKOWN\_ERROR***("000001"), // 未知异常

***JSON\_FORMAT\_ERROR***("010001"), // JSON 字符串格式错误

***JSON\_CONTENT\_ERROR***("01002"), // JSON 字符串内容错误

***TRADE\_INSERT\_ERROR***("010003"), // 插入工单失败

***TRADE\_SELECT\_ERROR***("010004"), // 查询工单失败

***TRADE\_UPDATE\_ERROR***("010005"), // 更新工地那失败

***INFOSTR\_LEN\_ERROR***("020001"), // info str 长度有问题

***INFOSTR\_VALUE\_ERROR***("020002"), // info str 值有问题

***RELECT\_OBEJCT\_ERROR***("020003"), // 反射实体错误

***RELECT\_FIELD\_ERROR***("020004"), // 反射字段错误

***TRANS\_ANALYZE\_ERROR***("020005"), // 解析表达式错误

***TRANS\_ENUM\_ERROR***("020006"), // 枚举错误

***TRANS\_TO\_ITABLE\_ERROR***("020007"), // 转换为I表出错

***NOT\_SUPPORT\_UPFIELD***("030001"), // 上发位不支持

***HANDLER\_CREATE\_ERROR***("030002"), // 创建Handler失败

***HANDLER\_BUSS\_ERROR***("030003"), // Handler 业务失败

**SYNC\_MDB\_ERROR**("040001"), // 上发MDB失败