文档变更记录

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 变更内容说明 | 版本号 | 版本日期 | 执笔人 |
| 1 | 初稿 | 1.0 | 2013-1-14 | 郑东海 |
| 2 | 新增PL/SQL 中现有数据表操作SQL语句导出。 | 1.1 | 2013-03-19 | 郑东海 |
| 3 | 修改4创建和配置新建domain的配置文件(手机阅读为例)下4.1配置TransConfig.xml和DBCPConfig.properties标题和内容 | 1.2 | 2013-04-02 | 郑东海 |
| 4 | 2数据库配置 高亮需要注意部分  增加2.8 配置t\_t\_routing\_group  4.1.1和4.1.2修改创建方法。 | 1.3 | 2013-05-28 | 郑东海 |
| 5 | 添加“6 刷新对应duns应用内存中的数据（包括数据表和校验文件）” | 1.4 | 2013-06-04 | 郑东海 |
| 6 | 1. 修改创建和配置新建domain的配置文件步骤 2. 添加“6 刷新对应duns应用内存中的数据（包括数据表和校验文件）” 3. 新增 “9 新平台接入测试环境配置文档提交” 4. 新增“注意事项”（重要） 5. 新增附录“11附录” | 1.5 | 2014-03-28 | 郑东海 |
| 7 | 1、修改crital（关键交易）默认值为0(非关键交易) 2、增加6.2 zookeeper与T说明。 | 1.6 | 2014-06-06 | 郑东海 |

# 阅读须知

该操作手册以如若没明确指出凭条，则默认以手机阅读的配置为例进行编写，且只针对于一般性配置。文档中的表格，语句仅供参考。如果新增平台或者省boss需要对语句进行部分修改。数据库操作需要仔细精确，然后再进行操作，特别是delete语句谨慎使用，别把之前配置好的数据弄乱了！语句中红色字体为需要修改的部分，具体含义会在相应处说明。

该手册所有操作以编写时间为准。后续可能更新

## 阅读规范，获取信息。

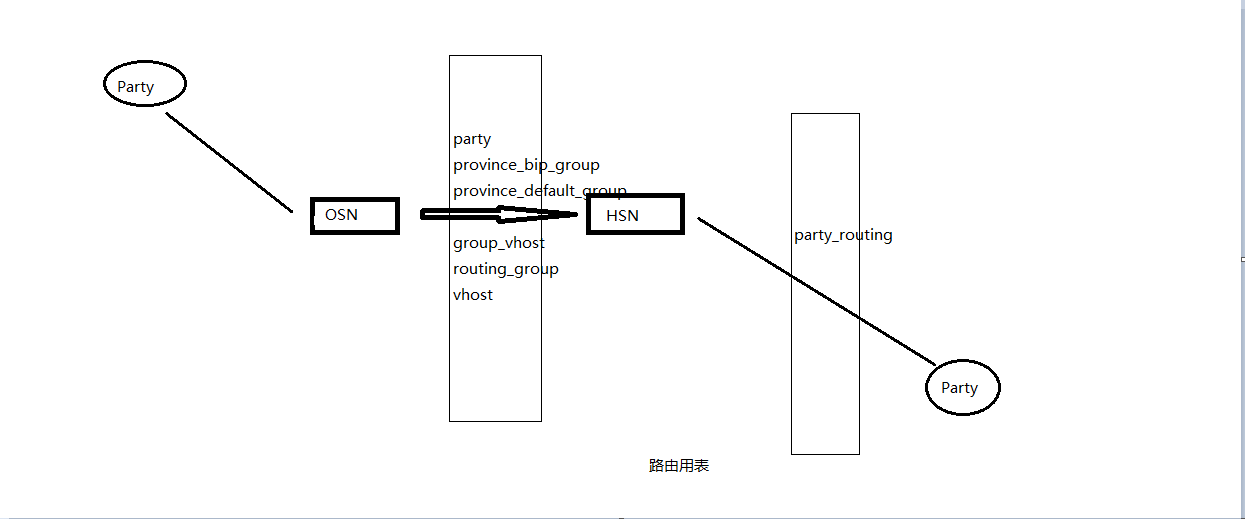
查阅手机阅读规范：“中国移动网状网系统接口规范 - 手机阅读平台分册 V1.0.2.doc”中“网状网与手机阅读平台接口”接口部分。收集数据库配置bip，activity信息。如下表：



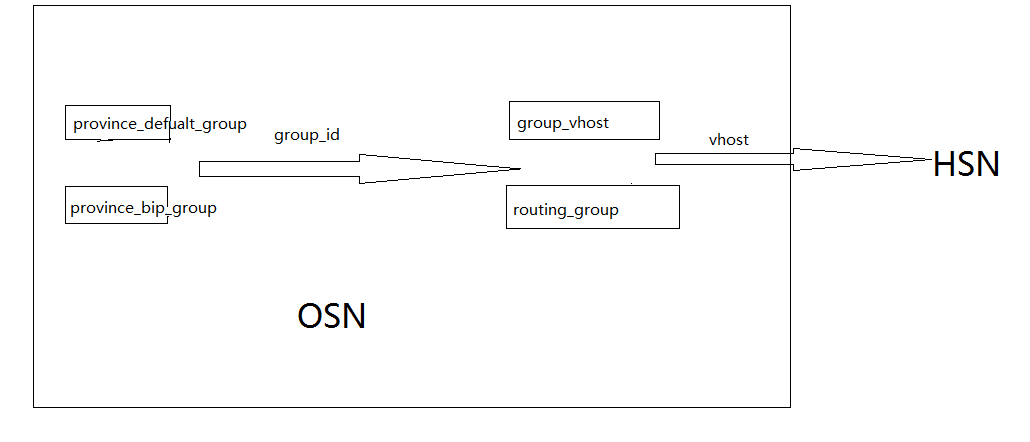
其中need reveral是代表是否冲正，recon\_flag代表是否对账,MEBP为手机阅读缩写

## 数据库配置

路由涉及的表：



具体需找路由过程：



基础表属性表：

Bip，activity，bip\_activity，domain，party，vhost

根据查看文档所得信息，我们需要完成以下几张表的配置：

### 2.1 t\_t\_domain:

T\_t\_domain表中有以下几个字段需要说明：

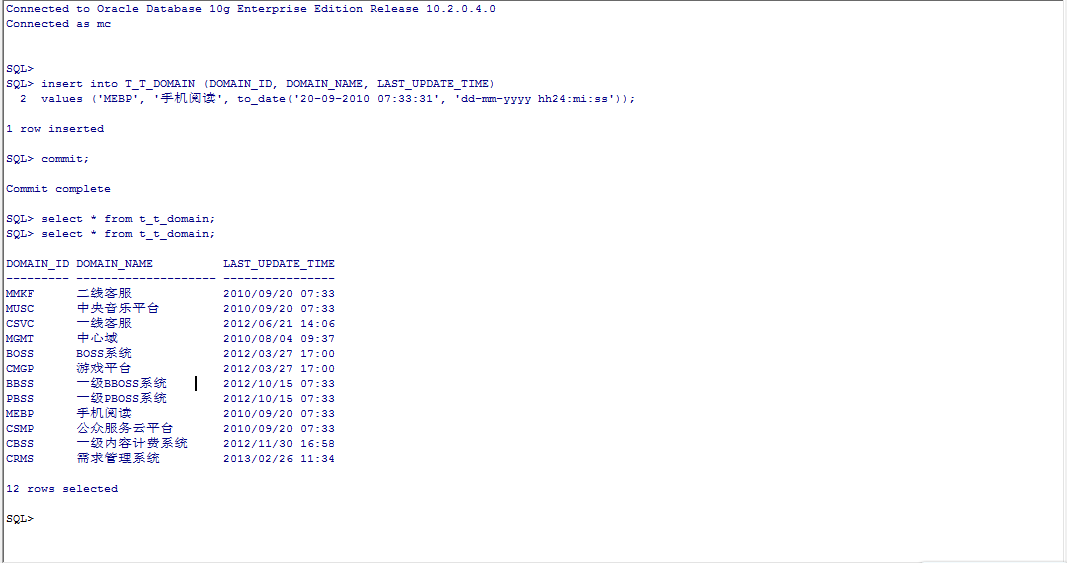
Domain\_id： party所属域名，新平台一般都用平台缩写

Domain\_name：域名描述

以手机阅读为例我们需要利用PL/SQL在MC用户下执行sql语句如下（红色字段为新平台需要修改属性）：

**insert** **into** T\_T\_DOMAIN (DOMAIN\_ID, DOMAIN\_NAME, LAST\_UPDATE\_TIME)  
**values** ('MEBP', '手机阅读', to\_date('20-09-2010 07:33:31', 'dd-mm-yyyy hh24:mi:ss'));  
  
**commit**;

执行结果如图：



### t\_t\_activity

在t\_t\_activity表中，需要明确如下属性意义，新平台配置需要修改。

Activity\_id：交易名或接口名

Activity\_type: 交易类型（系统大圈：0，通知：1，冲正：2）

Activity\_name: 交易描述

Max\_Retry\_count：通知最大重发次数（交易类型为通知时配置为10，大圈配置为0）

Need\_reversal: 是否冲正（0：不需要，1：需要）

Recon\_Flag: 是否对账（0：不需要，1：需要）

Service\_level：是否服务等级（0-9之间，数值越大服务等级越高）

根据规范中收集的信息，我们我们插入新的activity到t\_t\_activity表。在PL/Sql 命令窗口中执行如下命令：

insert into T\_T\_ACTIVITY (ACTIVITY\_ID, ACTIVITY\_TYPE, ACTIVITY\_NAME, SCHEMA\_NAME, TIMEOUT, MAX\_RETRY\_COUNT, NEED\_REVERSAL, BFEE\_DIRECTION, AFEE\_METHOD\_ID, RECON\_FLAG, LAST\_UPDATE\_TIME, RECORD\_TYPE, SERVICE\_LEVEL)

values ('T2001542', 0, '手机阅读业务受理', null, 180000, 0, '0', '0', '0', '1', to\_date('13-08-2012 17:21:06', 'dd-mm-yyyy hh24:mi:ss'), 'N', '1');

insert into T\_T\_ACTIVITY (ACTIVITY\_ID, ACTIVITY\_TYPE, ACTIVITY\_NAME, SCHEMA\_NAME, TIMEOUT, MAX\_RETRY\_COUNT, NEED\_REVERSAL, BFEE\_DIRECTION, AFEE\_METHOD\_ID, RECON\_FLAG, LAST\_UPDATE\_TIME, RECORD\_TYPE, SERVICE\_LEVEL)

values ('T2101546', 1, '手机阅读批量业务受理', null, 180000, 3, '0', '0', '0', '1', to\_date('13-08-2012 17:21:06', 'dd-mm-yyyy hh24:mi:ss'), 'N', '1');

insert into T\_T\_ACTIVITY (ACTIVITY\_ID, ACTIVITY\_TYPE, ACTIVITY\_NAME, SCHEMA\_NAME, TIMEOUT, MAX\_RETRY\_COUNT, NEED\_REVERSAL, BFEE\_DIRECTION, AFEE\_METHOD\_ID, RECON\_FLAG, LAST\_UPDATE\_TIME, RECORD\_TYPE, SERVICE\_LEVEL)

values ('T2101547', 1, '手机阅读批量业务受理结果反馈', null, 180000, 3, '0', '0', '0', '1', to\_date('13-08-2012 17:21:06', 'dd-mm-yyyy hh24:mi:ss'), 'N', '1');

insert into T\_T\_ACTIVITY (ACTIVITY\_ID, ACTIVITY\_TYPE, ACTIVITY\_NAME, SCHEMA\_NAME, TIMEOUT, MAX\_RETRY\_COUNT, NEED\_REVERSAL, BFEE\_DIRECTION, AFEE\_METHOD\_ID, RECON\_FLAG, LAST\_UPDATE\_TIME, RECORD\_TYPE, SERVICE\_LEVEL)

values ('T2101526', 1, '省BOSS手机阅读服务开通', null, 180000, 3, '0', '0', '0', '1', to\_date('13-08-2012 17:21:06', 'dd-mm-yyyy hh24:mi:ss'), 'N', '0');

insert into T\_T\_ACTIVITY (ACTIVITY\_ID, ACTIVITY\_TYPE, ACTIVITY\_NAME, SCHEMA\_NAME, TIMEOUT, MAX\_RETRY\_COUNT, NEED\_REVERSAL, BFEE\_DIRECTION, AFEE\_METHOD\_ID, RECON\_FLAG, LAST\_UPDATE\_TIME, RECORD\_TYPE, SERVICE\_LEVEL)

values ('T2101527', 1, '省BOSS手机阅读服务开通结果反馈', null, 180000, 3, '0', '0', '0', '1', to\_date('13-08-2012 17:21:06', 'dd-mm-yyyy hh24:mi:ss'), 'N', '1');

insert into T\_T\_ACTIVITY (ACTIVITY\_ID, ACTIVITY\_TYPE, ACTIVITY\_NAME, SCHEMA\_NAME, TIMEOUT, MAX\_RETRY\_COUNT, NEED\_REVERSAL, BFEE\_DIRECTION, AFEE\_METHOD\_ID, RECON\_FLAG, LAST\_UPDATE\_TIME, RECORD\_TYPE, SERVICE\_LEVEL)

values ('T2101528', 1, '省BOSS批量手机阅读服务开通', null, 180000, 3, '0', '0', '0', '1', to\_date('13-08-2012 17:21:06', 'dd-mm-yyyy hh24:mi:ss'), 'N', '1');

insert into T\_T\_ACTIVITY (ACTIVITY\_ID, ACTIVITY\_TYPE, ACTIVITY\_NAME, SCHEMA\_NAME, TIMEOUT, MAX\_RETRY\_COUNT, NEED\_REVERSAL, BFEE\_DIRECTION, AFEE\_METHOD\_ID, RECON\_FLAG, LAST\_UPDATE\_TIME, RECORD\_TYPE, SERVICE\_LEVEL)

values ('T2101529', 1, '省BOSS批量手机阅读服务开通结果反馈', null, 180000, 3, '0', '0', '0', '1', to\_date('13-08-2012 17:21:06', 'dd-mm-yyyy hh24:mi:ss'), 'N', '1');

insert into T\_T\_ACTIVITY (ACTIVITY\_ID, ACTIVITY\_TYPE, ACTIVITY\_NAME, SCHEMA\_NAME, TIMEOUT, MAX\_RETRY\_COUNT, NEED\_REVERSAL, BFEE\_DIRECTION, AFEE\_METHOD\_ID, RECON\_FLAG, LAST\_UPDATE\_TIME, RECORD\_TYPE, SERVICE\_LEVEL)

values ('T2101530', 1, '省boss批量服务暂停/恢复', null, 180000, 3, '0', '0', '0', '1', to\_date('13-08-2012 17:21:06', 'dd-mm-yyyy hh24:mi:ss'), 'N', '1');

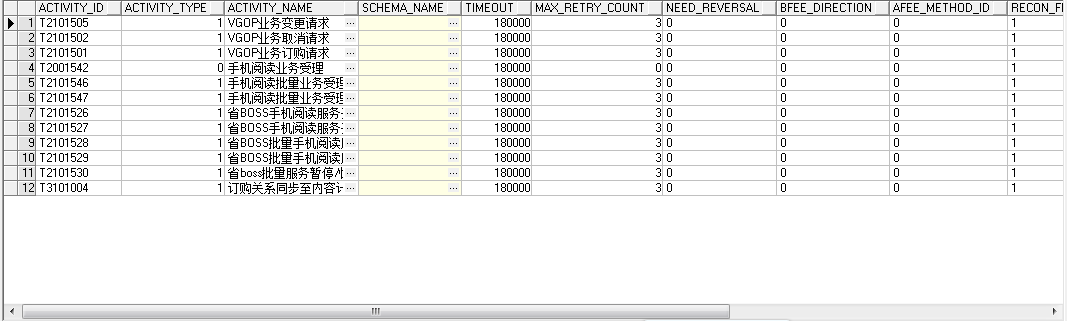
insert into T\_T\_ACTIVITY (ACTIVITY\_ID, ACTIVITY\_TYPE, ACTIVITY\_NAME, SCHEMA\_NAME, TIMEOUT, MAX\_RETRY\_COUNT, NEED\_REVERSAL, BFEE\_DIRECTION, AFEE\_METHOD\_ID, RECON\_FLAG, LAST\_UPDATE\_TIME, RECORD\_TYPE, SERVICE\_LEVEL)

values ('T3101004', 1, '订购关系同步至内容计费接口', null, 180000, 3, '0', '0', '0', '1', to\_date('13-08-2012 17:21:06', 'dd-mm-yyyy hh24:mi:ss'), 'N', '1');

commit;

使用如下命令验证是否插入activity结果如下：

Select \* from t\_t\_activity where activity\_id like 'T21015%' or activity\_id like 'T3101004' or activity\_id like 'T2001542'



### 配置t\_t\_activity\_action

根据规范导航中“交易报文”目录，确定activity应答SvcCont是否为空。如果不为空则该activity在t\_t\_activity\_action中应对应aciton 为0和1 两条记录，否则该activity只对应action为0的一条记录。t\_t\_activity\_activity\_action 中如下属性需要明确，且需要根据规范做相应修改：

BIP\_id：业务ID，参考新平台网状网规范

Activity\_id：交易ID或接口，参考规范

Action\_code：该属性只能填0或1两个值。0代表该接口发起需要做action\_class的操作。1代表该接口应答时需要做action\_class的操作。

Action\_class：一个类名，代表的是一个动作。比如校验报文体或加密

Arguments：action\_class所需要的参数。这里当action \_class为’com.hp.ecip.trans.biz.policy.ValidateAction’（校验报文体）时，需要填写校验文件名。校验文件名命名格式为：activity名\_n.xsd（n为0或者1。1代表应答SvcCont不为空，否则为0）.

其他属性值保持和现有其他activity一致.）。所以MEBP手机阅读的t\_t\_activity\_action表格需要要插入如下sql：

--BIP2B542

insert into T\_T\_ACTIVITY\_ACTION (BIP\_ID, ACTIVITY\_ID, ACTION\_CODE, ACTION\_CLASS, ARGUMENTS, LAST\_UPDATE\_TIME)

values ('BIP2B542', 'T2001542', '0', 'com.hp.ecip.trans.biz.policy.ValidateAction', 'T2001542\_0.xsd', to\_date('01-02-2013 10:20:35', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_ACTIVITY\_ACTION (BIP\_ID, ACTIVITY\_ID, ACTION\_CODE, ACTION\_CLASS, ARGUMENTS, LAST\_UPDATE\_TIME)

values ('BIP4B814', 'T2001542', '1', 'com.hp.ecip.trans.biz.policy.ValidateAction', 'T2001542\_1.xsd', to\_date('01-02-2013 10:20:35', 'dd-mm-yyyy hh24:mi:ss'));

--BIP2B546

insert into T\_T\_ACTIVITY\_ACTION (BIP\_ID, ACTIVITY\_ID, ACTION\_CODE, ACTION\_CLASS, ARGUMENTS, LAST\_UPDATE\_TIME)

values ('BIP2B546', 'T2101546', '0', 'com.hp.ecip.trans.biz.policy.ValidateAction', 'T2101546\_0.xsd', to\_date('01-02-2013 10:20:35', 'dd-mm-yyyy hh24:mi:ss'));

--BIP2B547

insert into T\_T\_ACTIVITY\_ACTION (BIP\_ID, ACTIVITY\_ID, ACTION\_CODE, ACTION\_CLASS, ARGUMENTS, LAST\_UPDATE\_TIME)

values ('BIP2B547', 'T2101547', '0', 'com.hp.ecip.trans.biz.policy.ValidateAction', 'T2101547\_0.xsd', to\_date('01-02-2013 10:20:35', 'dd-mm-yyyy hh24:mi:ss'));

--BIP2B526

insert into T\_T\_ACTIVITY\_ACTION (BIP\_ID, ACTIVITY\_ID, ACTION\_CODE, ACTION\_CLASS, ARGUMENTS, LAST\_UPDATE\_TIME)

values ('BIP2B526', 'T2101526', '0', 'com.hp.ecip.trans.biz.policy.ValidateAction', 'T2101526\_0.xsd', to\_date('01-02-2013 10:20:35', 'dd-mm-yyyy hh24:mi:ss'));

--BIP2B527

insert into T\_T\_ACTIVITY\_ACTION (BIP\_ID, ACTIVITY\_ID, ACTION\_CODE, ACTION\_CLASS, ARGUMENTS, LAST\_UPDATE\_TIME)

values ('BIP2B527', 'T2101527', '0', 'com.hp.ecip.trans.biz.policy.ValidateAction', 'T2101527\_0.xsd', to\_date('01-02-2013 10:20:35', 'dd-mm-yyyy hh24:mi:ss'));

--BIP2B528

insert into T\_T\_ACTIVITY\_ACTION (BIP\_ID, ACTIVITY\_ID, ACTION\_CODE, ACTION\_CLASS, ARGUMENTS, LAST\_UPDATE\_TIME)

values ('BIP2B528', 'T2101528', '0', 'com.hp.ecip.trans.biz.policy.ValidateAction', 'T2101528\_0.xsd', to\_date('01-02-2013 10:20:35', 'dd-mm-yyyy hh24:mi:ss'));

--BIP2B529

insert into T\_T\_ACTIVITY\_ACTION (BIP\_ID, ACTIVITY\_ID, ACTION\_CODE, ACTION\_CLASS, ARGUMENTS, LAST\_UPDATE\_TIME)

values ('BIP2B529', 'T2101529', '0', 'com.hp.ecip.trans.biz.policy.ValidateAction', 'T2101529\_0.xsd', to\_date('01-02-2013 10:20:35', 'dd-mm-yyyy hh24:mi:ss'));

--BIP2B530

insert into T\_T\_ACTIVITY\_ACTION (BIP\_ID, ACTIVITY\_ID, ACTION\_CODE, ACTION\_CLASS, ARGUMENTS, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'T2101530', '0', 'com.hp.ecip.trans.biz.policy.ValidateAction', 'T2101530\_0.xsd', to\_date('01-02-2013 10:20:35', 'dd-mm-yyyy hh24:mi:ss'));

--BIP3B098

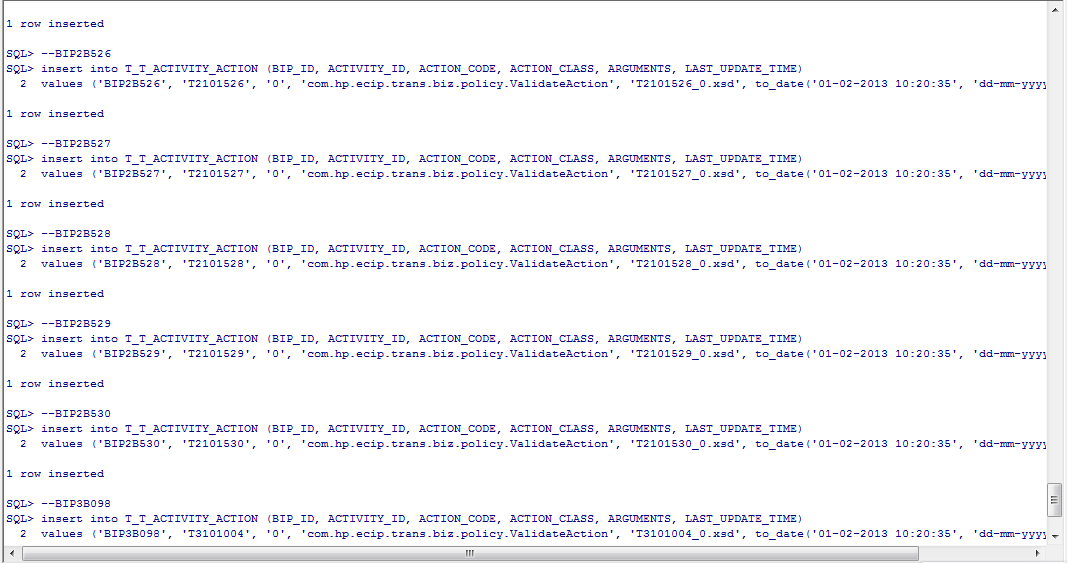
insert into T\_T\_ACTIVITY\_ACTION (BIP\_ID, ACTIVITY\_ID, ACTION\_CODE, ACTION\_CLASS, ARGUMENTS, LAST\_UPDATE\_TIME)

values ('BIP3B098', 'T3101004', '0', 'com.hp.ecip.trans.biz.policy.ValidateAction', 'T3101004\_0.xsd', to\_date('01-02-2013 10:20:35', 'dd-mm-yyyy hh24:mi:ss'));

--提交

commit;

执行结果如下：



### 配置t\_t\_bip\_avtivity

首先明确该表中的部分列的含义：

Bip\_id, activity\_id,请参考规范。

Step：代表activity属于bip的第几步

Critical：代表这个activity是不是关键业务。如果规范没有说明，默认配置为0：非关键交易（0：非关键交易）

Settle flag：代表是不是结算。如果规范未说明请配置为0：不需要结算。（1：需要结算）

Repeat times：目前填1，不需要修改

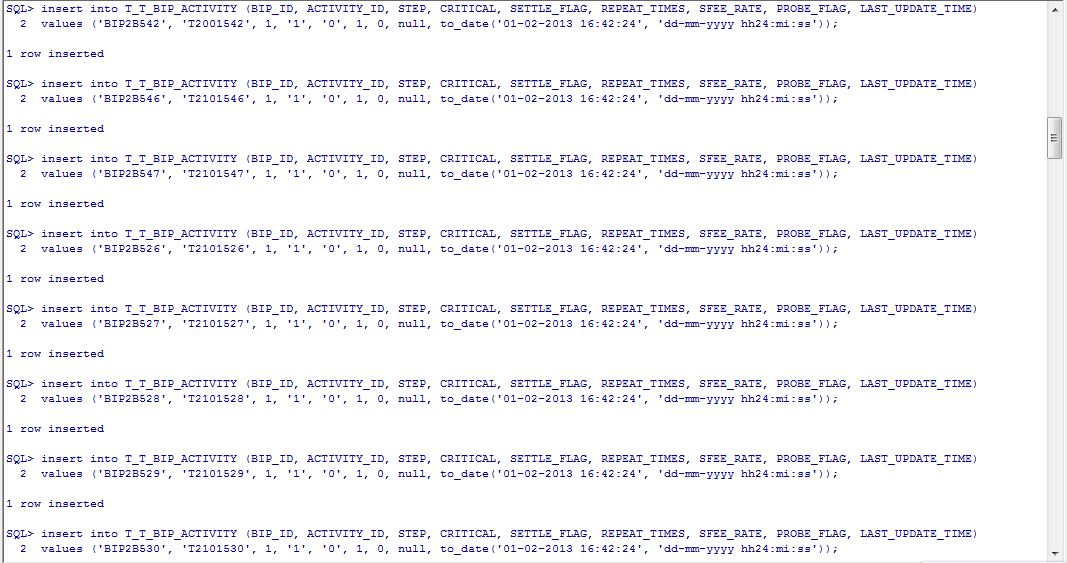
Sfee\_rate:达标结算费率，不需要修改

Probe\_flag:现在都填null，不需要修改。

因此针对MEBP手机阅读我们需要执行如下命令完成t\_t\_bip\_activity配置：

**insert** **into** T\_T\_BIP\_ACTIVITY (BIP\_ID, ACTIVITY\_ID, STEP, CRITICAL, SETTLE\_FLAG, REPEAT\_TIMES, SFEE\_RATE, PROBE\_FLAG, LAST\_UPDATE\_TIME)  
**values** ('BIP2B542', 'T2001542', 1, '1', '0', 1, 0, **null**, to\_date('01-02-2013 16:42:24', 'dd-mm-yyyy hh24:mi:ss'));  
  
**insert** **into** T\_T\_BIP\_ACTIVITY (BIP\_ID, ACTIVITY\_ID, STEP, CRITICAL, SETTLE\_FLAG, REPEAT\_TIMES, SFEE\_RATE, PROBE\_FLAG, LAST\_UPDATE\_TIME)  
**values** ('BIP2B546', 'T2101546', 1, '1', '0', 1, 0, **null**, to\_date('01-02-2013 16:42:24', 'dd-mm-yyyy hh24:mi:ss'));  
  
**insert** **into** T\_T\_BIP\_ACTIVITY (BIP\_ID, ACTIVITY\_ID, STEP, CRITICAL, SETTLE\_FLAG, REPEAT\_TIMES, SFEE\_RATE, PROBE\_FLAG, LAST\_UPDATE\_TIME)  
**values** ('BIP2B547', 'T2101547', 1, '1', '0', 1, 0, **null**, to\_date('01-02-2013 16:42:24', 'dd-mm-yyyy hh24:mi:ss'));  
  
**insert** **into** T\_T\_BIP\_ACTIVITY (BIP\_ID, ACTIVITY\_ID, STEP, CRITICAL, SETTLE\_FLAG, REPEAT\_TIMES, SFEE\_RATE, PROBE\_FLAG, LAST\_UPDATE\_TIME)  
**values** ('BIP2B526', 'T2101526', 1, '1', '0', 1, 0, **null**, to\_date('01-02-2013 16:42:24', 'dd-mm-yyyy hh24:mi:ss'));  
  
**insert** **into** T\_T\_BIP\_ACTIVITY (BIP\_ID, ACTIVITY\_ID, STEP, CRITICAL, SETTLE\_FLAG, REPEAT\_TIMES, SFEE\_RATE, PROBE\_FLAG, LAST\_UPDATE\_TIME)  
**values** ('BIP2B527', 'T2101527', 1, '1', '0', 1, 0, **null**, to\_date('01-02-2013 16:42:24', 'dd-mm-yyyy hh24:mi:ss'));  
  
**insert** **into** T\_T\_BIP\_ACTIVITY (BIP\_ID, ACTIVITY\_ID, STEP, CRITICAL, SETTLE\_FLAG, REPEAT\_TIMES, SFEE\_RATE, PROBE\_FLAG, LAST\_UPDATE\_TIME)  
**values** ('BIP2B528', 'T2101528', 1, '1', '0', 1, 0, **null**, to\_date('01-02-2013 16:42:24', 'dd-mm-yyyy hh24:mi:ss'));  
  
**insert** **into** T\_T\_BIP\_ACTIVITY (BIP\_ID, ACTIVITY\_ID, STEP, CRITICAL, SETTLE\_FLAG, REPEAT\_TIMES, SFEE\_RATE, PROBE\_FLAG, LAST\_UPDATE\_TIME)  
**values** ('BIP2B529', 'T2101529', 1, '1', '0', 1, 0, **null**, to\_date('01-02-2013 16:42:24', 'dd-mm-yyyy hh24:mi:ss'));  
  
**insert** **into** T\_T\_BIP\_ACTIVITY (BIP\_ID, ACTIVITY\_ID, STEP, CRITICAL, SETTLE\_FLAG, REPEAT\_TIMES, SFEE\_RATE, PROBE\_FLAG, LAST\_UPDATE\_TIME)  
**values** ('BIP2B530', 'T2101530', 1, '1', '0', 1, 0, **null**, to\_date('01-02-2013 16:42:24', 'dd-mm-yyyy hh24:mi:ss'));  
  
**insert** **into** T\_T\_BIP\_ACTIVITY (BIP\_ID, ACTIVITY\_ID, STEP, CRITICAL, SETTLE\_FLAG, REPEAT\_TIMES, SFEE\_RATE, PROBE\_FLAG, LAST\_UPDATE\_TIME)  
**values** ('BIP3B098', 'T3101004', 1, '1', '0', 1, 0, **null**, to\_date('01-02-2013 16:42:24', 'dd-mm-yyyy hh24:mi:ss'));  
  
  
**commit**;

结果如下图：



### 2.5配置t\_t\_bip

BIP\_ID：业务名，参考规范

BIP\_TYPE ：业务类型，属性值为BIP字符串后面第一位数字。

BIP\_NAME：业务名参考规范

BIP\_NICE ：业务优先级。分1-9个等级，如果规范说明优先级高，请按1-3为敌，4-6为中，7-9为高的依据来填入属性值。

LAST\_UPDATE\_TIME：可以不做修改。

因此可以在数据库中插入如下sql语句：

insert into T\_T\_BIP (BIP\_ID, BIP\_TYPE, BIP\_NAME, BIP\_NICE, LAST\_UPDATE\_TIME)

values ('BIP2B542', '2', '手机阅读业务受理', 3, to\_date('22-08-2012 16:15:57', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_BIP (BIP\_ID, BIP\_TYPE, BIP\_NAME, BIP\_NICE, LAST\_UPDATE\_TIME)

values ('BIP2B546', '2', '手机阅读批量业务受理', 3, to\_date('22-08-2012 16:15:57', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_BIP (BIP\_ID, BIP\_TYPE, BIP\_NAME, BIP\_NICE, LAST\_UPDATE\_TIME)

values ('BIP2B547', '2', '手机阅读批量业务受理结果反馈', 3, to\_date('22-08-2012 16:15:57', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_BIP (BIP\_ID, BIP\_TYPE, BIP\_NAME, BIP\_NICE, LAST\_UPDATE\_TIME)

values ('BIP2B526', '2', '省BOSS手机阅读服务开通', 3, to\_date('22-08-2012 16:15:57', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_BIP (BIP\_ID, BIP\_TYPE, BIP\_NAME, BIP\_NICE, LAST\_UPDATE\_TIME)

values ('BIP2B527', '2', '省BOSS手机阅读服务开通结果反馈', 3, to\_date('22-08-2012 16:15:57', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_BIP (BIP\_ID, BIP\_TYPE, BIP\_NAME, BIP\_NICE, LAST\_UPDATE\_TIME)

values ('BIP2B528', '2', '省BOSS批量手机阅读服务开通', 3, to\_date('22-08-2012 16:15:57', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_BIP (BIP\_ID, BIP\_TYPE, BIP\_NAME, BIP\_NICE, LAST\_UPDATE\_TIME)

values ('BIP2B529', '2', '省BOSS批量手机阅读服务开通结果反馈', 3, to\_date('22-08-2012 16:15:57', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_BIP (BIP\_ID, BIP\_TYPE, BIP\_NAME, BIP\_NICE, LAST\_UPDATE\_TIME)

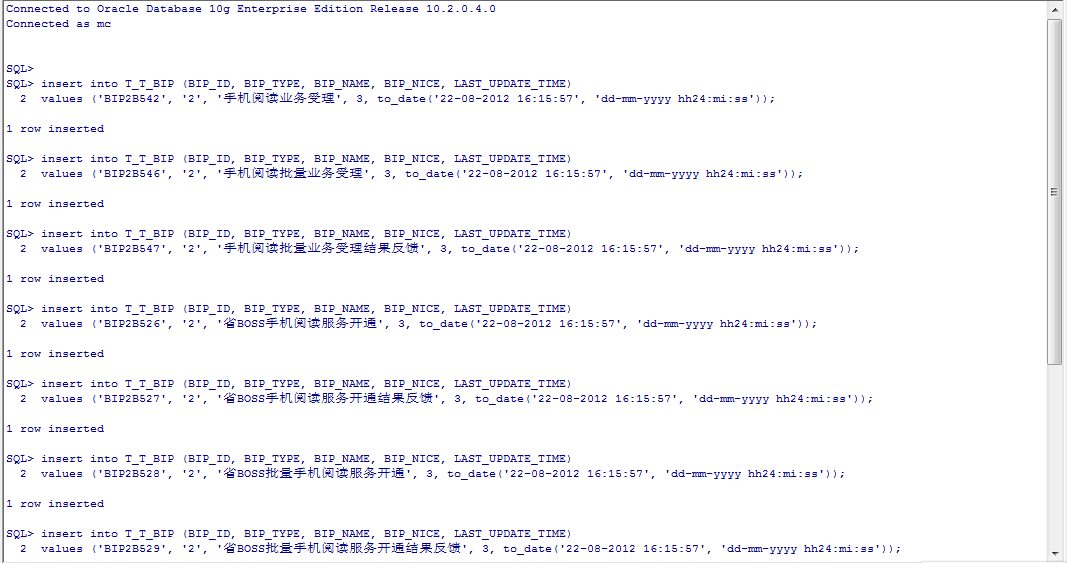
values ('BIP2B530', '2', '省boss批量服务暂停/恢复', 3, to\_date('22-08-2012 16:15:57', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_BIP (BIP\_ID, BIP\_TYPE, BIP\_NAME, BIP\_NICE, LAST\_UPDATE\_TIME)

values ('BIP3B098', '2', '订购关系同步至内容计费接口', 3, to\_date('22-08-2012 16:15:57', 'dd-mm-yyyy hh24:mi:ss'));

commit;

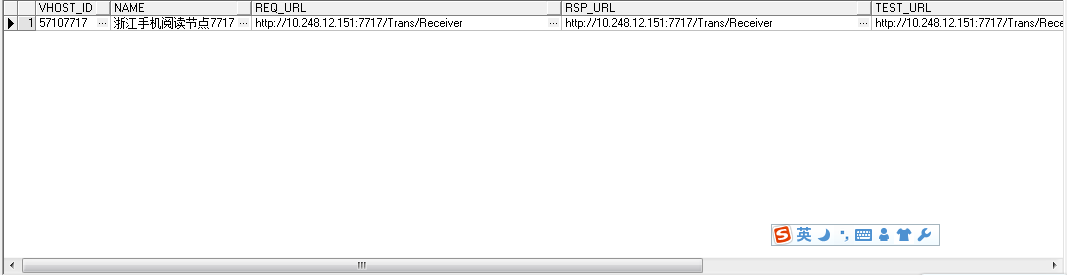
插入结果如下：



然后通过查询t\_t\_bip表，验证手机阅读平台对应的bip是否插入t\_t\_bip表。

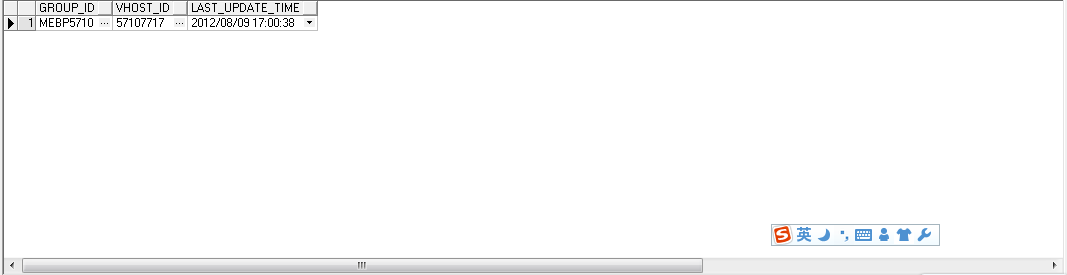
### 2.6配置t\_t\_vhost

检查分配给手机阅读的vhost的地址是否已经存在，存在则不需要配置。现在手机阅读的vhost并没有配置，所以我们需要配置vhost\_id为57107717（duns\_id+端口号）。由于测试环境一个平台配置的vhost少，我们直接使用select update语句，复制黏贴进行修改。修改结果如下：



### 2.7 配置t\_t\_group\_vhost

创建新的vhost以后，我们需要一个group\_id与之对应。Group可以有多个vhost。测试环境我们暂时只在一个group下配置一个vhost。因此我们可以通过select update方法，复制黏贴新建一条记录联系group和vhost，创建结果如下图：



如图MEBP对应的groupid为MEBP5710

### 2.8 配置routing\_group

Group下面可以配置多个vhost，group\_vhost表中配置的是group和vhost的关系。Routing\_group配置的则是group下的vhost按照什么比例来进行分发。因此新建一个group\_id时需要注意更新这张表，否则会出现找不到hsn的情况。在执行语句之前需要明白：

Group\_id：vhost的一个组。一个duns对应一个group\_id。

Name：group\_id对应的名称

Duns：duns号

LB\_Policy：分发机制（比如按权重分配，轮询分配等）

Priority：表示的是权重。当按权重分配时，分发按照权重的比例进行分配。

新加入group\_id可以通过如下语句新建记录。注意红色字体需要根据具体要求配置：

insert into T\_T\_ROUTING\_GROUP (GROUP\_ID, NAME, DUNS, LB\_POLICY, PRIORITY, COMMENTS, LAST\_UPDATE\_TIME, ACTIVE)

values ('BBSS9980', '中心SN', '9980', 'com.hp.ecip.trans.biz.policy.WeightedLBPolicy', 1, null, to\_date('30-12-2011', 'dd-mm-yyyy'), '1');

commit;

### 2.9 配置t\_t\_province\_bip\_group

这张表主要是指定发往平台的bip所要走的group\_id. t\_t\_province\_bip\_routing表中的属性有以下几列：

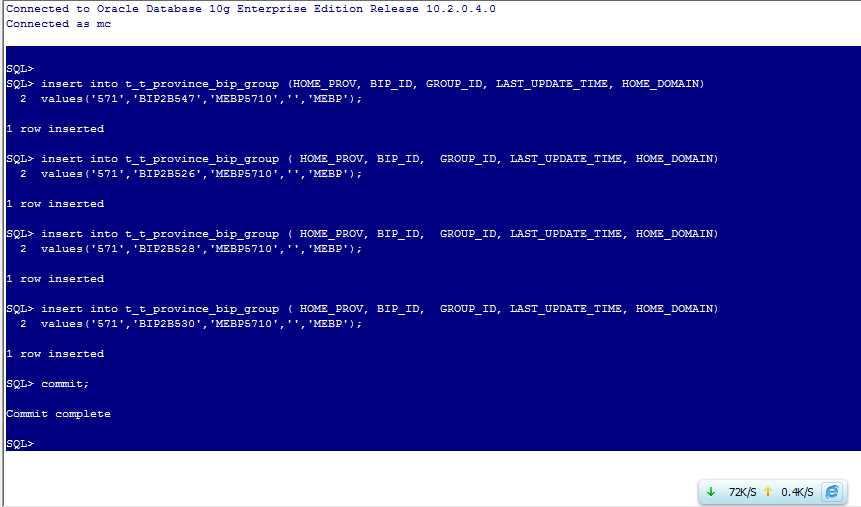
HOME\_PROV BIP\_ID GROUP\_ID LAST\_UPDATE\_TIME HOME\_DOMAIN

落地省代码 业务码 要走的group 最近更新时间 落地domain缩写

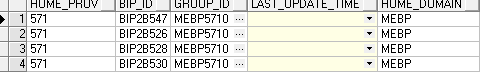
这张表的意思其实就是当遇到该表中配置的bip和落地省代码时，指定往配置的group\_id路由。需要注意的是group\_id可以是任意现成的group\_id。针对手机阅读MEBP，我们需要插入如下语句：

**insert** **into** t\_t\_province\_bip\_group (HOME\_PROV, BIP\_ID, GROUP\_ID, LAST\_UPDATE\_TIME, HOME\_DOMAIN)   
**values**('571','BIP2B547','MEBP5710','','MEBP');  
  
**insert** **into** t\_t\_province\_bip\_group ( HOME\_PROV, BIP\_ID, GROUP\_ID, LAST\_UPDATE\_TIME, HOME\_DOMAIN)   
**values**('571','BIP2B526','MEBP5710','','MEBP');  
  
**insert** **into** t\_t\_province\_bip\_group ( HOME\_PROV, BIP\_ID, GROUP\_ID, LAST\_UPDATE\_TIME, HOME\_DOMAIN)   
**values**('571','BIP2B528','MEBP5710','','MEBP');  
  
**insert** **into** t\_t\_province\_bip\_group ( HOME\_PROV, BIP\_ID, GROUP\_ID, LAST\_UPDATE\_TIME, HOME\_DOMAIN)   
**values**('571','BIP2B530','MEBP5710','','MEBP');  
**commit**;

插入结果如下：



接着使用select语句进行验证是否成功完成插入操作：



至此t\_t\_province\_bip\_group表配置完成；

### 2.10 配置t\_t\_province\_default\_group

这张表配置的是省boss的hsn地址。由于手机阅读之前已经配置过所有省boss的hsn地址，因此，除非多了一个省boss，否则这里不需要操作。

### 2.11 配置t\_t\_party

Party这张表配置的是平台和省boss的属性集合。一个平台对应一条t\_t\_party中的记录。所以我们通过下面语句手动添加一条手机阅读平台的记录。此处需要注意的是根据下表查出平台代码也就是数据库中的short party id.

**select** \* **from** t\_t\_party **for** **update**;



插入的记录如下（一条记录）：





### 2.12 配置t\_t\_party\_bip；

t\_t\_party\_bip这张表配置的是bip和party关系，role和test flag 分别可取0,1两个值。Role代表的是该party是发起或接收bip，发起为0，接受为1。而每个bip无论是party发起或者接收都有正式或测试类别之分。因此根据



表中的信息，我们可以看出什么bip是省作为发起方，什么bip是平台作为发起方。所以我们以BIP2B530为例，BIP2B530是省发起，平台落地接收。因此省boss记录的role属性应配置为0，平台对应的role属性为1。不论role为0或者1，对应的testFlag都应该有0，1两条记录。

Bip\_id：业务名，参考规范。

Party\_ID：party名。可能是BOSS或者平台名。例如：BOSS2000，MEBP5710

ROLE：表示是配置的party能否发起和接受配置的bip。0代表可以发起；1代表可以接受

Test\_Flag：0：非测试交易；1：测试交易。

BIP\_STATUS：bip的签到和签退状态。1：签到注册状态；0：签退状态。

Party\_STATUS: Party的签到和签退状态。1：签到注册状态；0：签退状态。

因此根据规范修改红色字体部分内容，然后插入类似如下语句，以省-MEBP的BIP为例，需要插入如下命令：（注意插入的数据需要精准，不要冗余，否则会占用较多的系统资源）

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS1000', '0', '0', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 07:30:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS1000', '0', '1', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 07:30:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS2000', '0', '0', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 07:30:11', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS2000', '0', '1', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 07:30:11', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS2100', '0', '0', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS2100', '0', '1', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS2200', '0', '0', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS2200', '0', '1', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS2300', '0', '0', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS2300', '0', '1', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS2400', '0', '0', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS2400', '0', '1', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS2500', '0', '0', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:38', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS2500', '0', '1', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:38', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS2700', '0', '0', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS2700', '0', '1', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS2800', '0', '0', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:07:03', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS2800', '0', '1', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:07:03', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS2900', '0', '0', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS2900', '0', '1', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS3110', '0', '0', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS3110', '0', '1', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS3510', '0', '0', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS3510', '0', '1', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS3710', '0', '0', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS3710', '0', '1', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS4310', '0', '0', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS4310', '0', '1', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS4510', '0', '0', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS4510', '0', '1', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS4710', '0', '0', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS4710', '0', '1', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS5310', '0', '0', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS5310', '0', '1', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS5510', '0', '0', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS5510', '0', '1', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS5710', '0', '0', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS5710', '0', '1', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS5910', '0', '0', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS5910', '0', '1', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS7310', '0', '0', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS7310', '0', '1', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS7710', '0', '0', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS7710', '0', '1', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS7910', '0', '0', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS7910', '0', '1', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS8510', '0', '0', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS8510', '0', '1', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS8710', '0', '0', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS8710', '0', '1', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS8910', '0', '0', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS8910', '0', '1', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS8980', '0', '0', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS8980', '0', '1', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS9310', '0', '0', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS9310', '0', '1', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS9510', '0', '0', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS9510', '0', '1', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS9710', '0', '0', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS9710', '0', '1', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS9910', '0', '0', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'BOSS9910', '0', '1', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:06:00', 'dd-mm-yyyy hh24:mi:ss'));

*--平台*

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

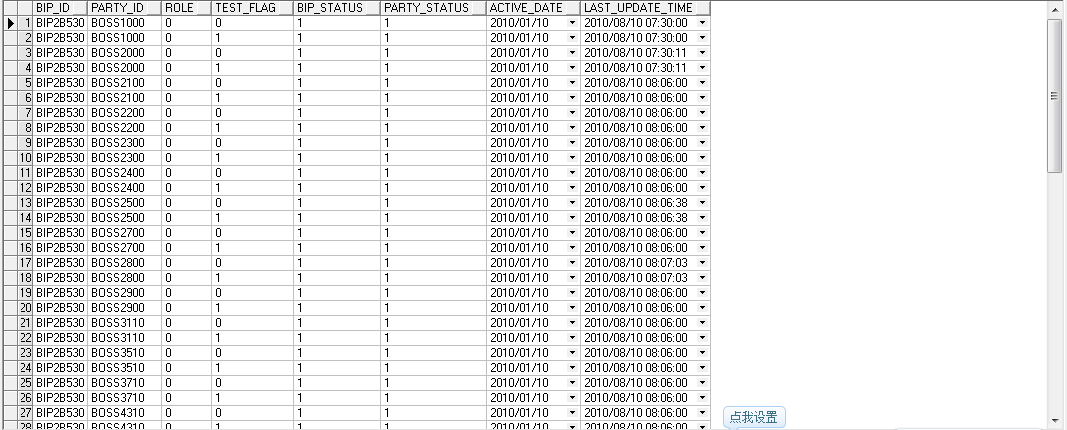
values ('BIP2B530', 'MEBP5710', '1', '0', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:07:32', 'dd-mm-yyyy hh24:mi:ss'));

insert into T\_T\_PARTY\_BIP (BIP\_ID, PARTY\_ID, ROLE, TEST\_FLAG, BIP\_STATUS, PARTY\_STATUS, ACTIVE\_DATE, LAST\_UPDATE\_TIME)

values ('BIP2B530', 'MEBP5710', '1', '1', '1', '1', to\_date('10-01-2010', 'dd-mm-yyyy'), to\_date('10-08-2010 08:07:32', 'dd-mm-yyyy hh24:mi:ss'));

commit;

插入完成后记得验证是否插入成功，select部分结果如下：



### 2.13 配置party\_routing

这张表是配置从hsn找party的地址。需要party提供自己的服务完整地址。需要明确如下属性意思：

Activity\_id: 接口名或交易名。

Party\_id: 省boss或者平台名。

Req\_url: 符合activity\_id和落地party条件发往的落地地址。

RSP\_URL：值同上

TEST\_URL：值同上

因为手机阅读暂时没有提供完整url，这里我以之前的CSMP的地址为例。插入如下语句：（大家需要根据规范的发起方落地方精确的插入数据，不要一个交易所有平台和省都插入记录）

**insert** **into** t\_t\_party\_routing(ACTIVITY\_ID, PARTY\_ID, REQ\_URL, RSP\_URL, TEST\_URL, LAST\_UPDATE\_TIME)  
**values**('T4011065','CSMP9970','http://221.176.53.10:8090/BOSSAgent/BossService','http://221.176.53.10:8090/BOSSAgent/BossService','http://221.176.53.10:8090/BOSSAgent/BossService',to\_date('2010-08-05','yyyy-mm-dd'));  
  
  
**insert** **into** t\_t\_party\_routing(ACTIVITY\_ID, PARTY\_ID, REQ\_URL, RSP\_URL, TEST\_URL, LAST\_UPDATE\_TIME)  
**values**('T4101727','CSMP9970','http://221.176.53.10:8090/BOSSAgent/BossService','http://221.176.53.10:8090/BOSSAgent/BossService','http://221.176.53.10:8090/BOSSAgent/BossService','');  
  
**commit**;

## 创建bes(domain,node,instance)

### 3.1 进入iastool所在目录

iastool是创建新的bes实例的工具。这里需要说明，iastool在38,41和151这三台机器的目录略有不同。以下是三台机器iastool所在绝对路劲：

41：/opt/ecip/T/bin

38：/opt/ecip/bes/bin

151：/1bosstest2/ecip/T/bes/bin/

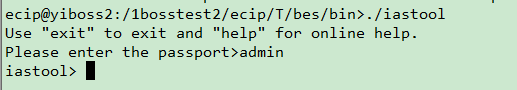
由于手机阅读平台在151上，所以我们将cd /1bosstest2/ecip/T/bes/bin/进入iastool所在目录。

### 3.2打开iastool工具，进行bes实例创建

步骤1：输入命令：./iastool

步骤2：输入密码：admin

过程如下图：



步骤3：开始创建bes实例。这里以手机阅读为例。根据分配的地址我们可以知道如下信息：

Domainname：domain5710 端口号：25710

Node：node5710

Instance：ins257101

Domain和node命名为：domain/node+duns名

Domain的管理端口号：2+duns名

Instance名称命名格式为：ins+端口号+实例号（1,2,3….）

由上面的信息，我们在151上创建手机阅读bes需执行如下命令：

1. 创建domain：（用户和密码都要设置成admin）

create --domain --user admin --password admin --adminport 25710 domain5710

1. 创建node：

create --node --port 25710 --domainname domain5710 node5710

1. 创建instance：

create --instance --node node5710 --port 25710 ins257101

1. 打开domain，node和instance

打开domain：start --domain domain5710

打开node：start --node --domainname domain5710 node5710

打开instance：start --instance --port 25710 ins257101（一般打开node以后instance会自动开启了，但是最好是执行下这个语句）

## 4 创建和配置新建domain的配置文件(手机阅读为例)

### 4.1创建、配置新domain配置文件。

为了提高维护的效率，/1bosstest2/ecip/T/runtime/conf/domain5710目录下除了DBCPConfig.properties，DBCPConfigSata.properties外都为链接，这些链接都是链接到/1bosstest2/ecip/M

/DataSyncClient/datas/T/conf/domain5710目录下的一个同名链接。随着T的更新，现在不需要链接。直接使用复制命令，从现有domain拷贝所有文件到新建domain5710下即可。

cp ../domain2000 /\* ./

#### 4.1.1配置TransConfig.xml文件。

该文件在 /1bosstest2/ecip/T/runtime/conf/domain5710/目录下。按照如下配置手机阅读domain的TransConfig.xml文件中DUNS\_ID，DBUSER，DBPASSWORD三个字段。其他字段不做变动。

<entry key="DUNS\_ID">5710</entry>

<entry key="DBUSER">sn571</entry>

<entry key="DBPASSWORD">sn571</entry>

#### 4.1.2配置DBCPConfig.properties文件。

修改该文件中的数据库的用户名和密码（其他字段不做变动），DBCPConfig.properties在/1bosstest2/ecip/T/runtime/conf/domain5710/目录下。

username=sn571

password=sn571

#### 4.1.3配置DBCPConfigSata.properties文件

修改该文件中的数据库的用户名和密码（其他字段不做变动）。

修改为：

username= sn571

password= sn571

### 4.2 上发xsd校验文件

1）制作xsd文件，以T2001542为例，如下



这里创建的xsd文件是校验报文体的xsd文件，如果是该activity应答SvcCont不为空，则需要根据规范创建T2001542\_1.xsd文件。

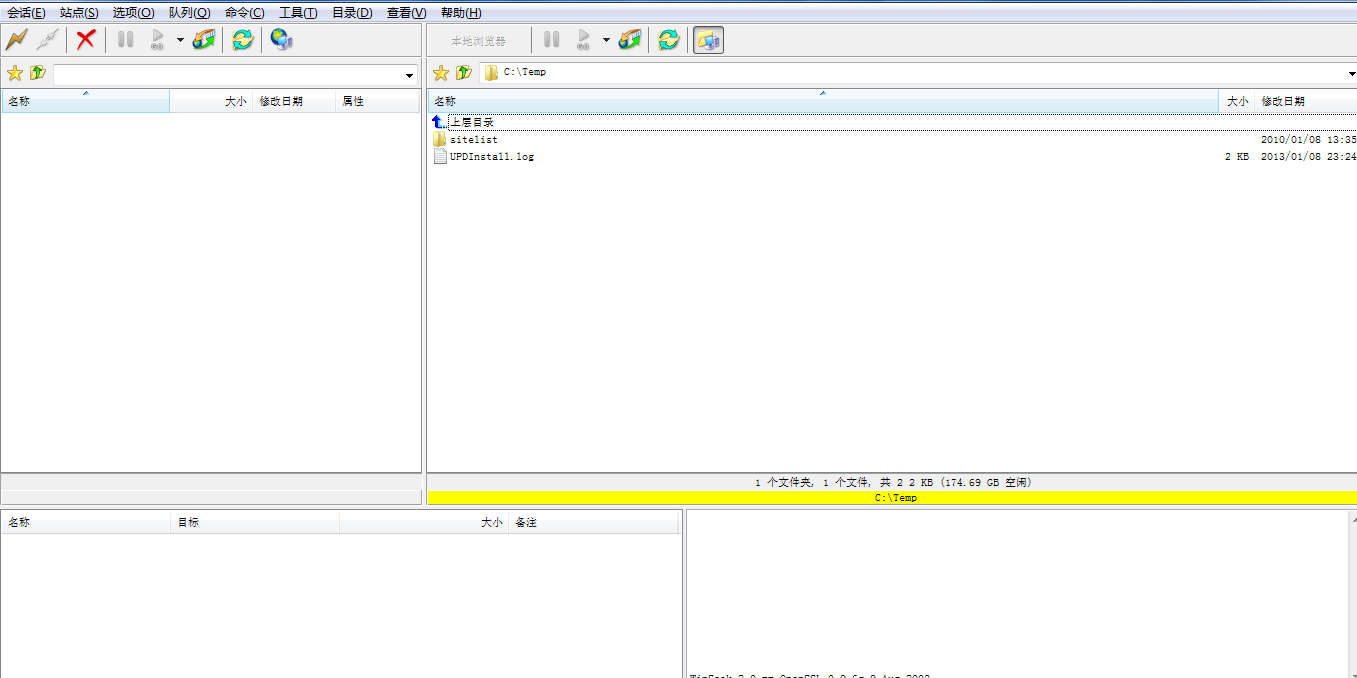
2）上传xsd文件到对应的机器上。

手机阅读平台和各省都有交互，所以我们需要将xsd文件上传到38,41,151三台机器上。这里可以使用FlashXp工具进行上传。

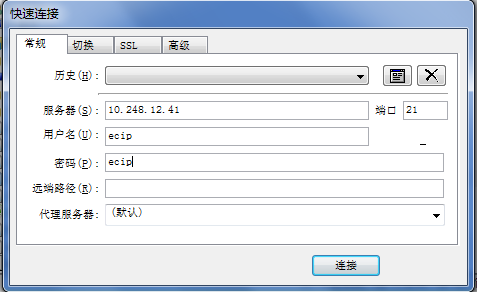


步骤如下：

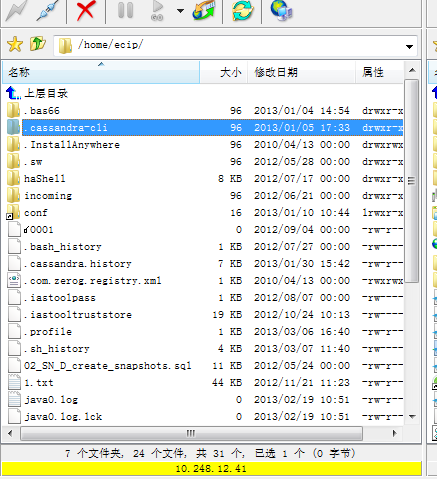
1. 打开FlashXp.exe,设置传输模式为二进制（会话->传输方式->二进制）。



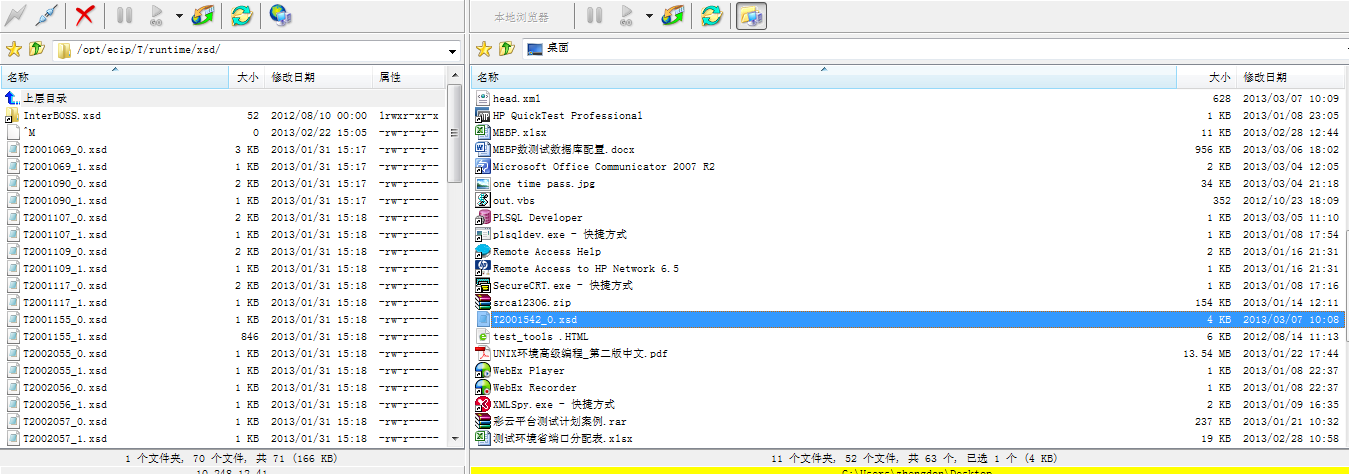
2、会话->快速连接，输入如下信息，连入41.



接入41后的目录。



3、进入41，xsd所在目录（xsd目录在runtime目录下，三台机器有些不同）。右边本地目录窗口选中本地xsd文件所在目录。如下图：



4、将创建好的xsd文件，从本地拉入左边41的xsd目录下就可以了。

注意操作完之后，登入41查看是否成功。

注意：

1. 41,38都可以通用户名/密码：ecip/ecip接入，但是151需要通过cmcbadm/Cmcb123! 接入151。
2. Xsd被上传成功不代表能在内存中生效，需要通过重启或者刷新内存才能生效，详喜操作请参考本文档第六章节。

## 5 部署应用到新建实例。

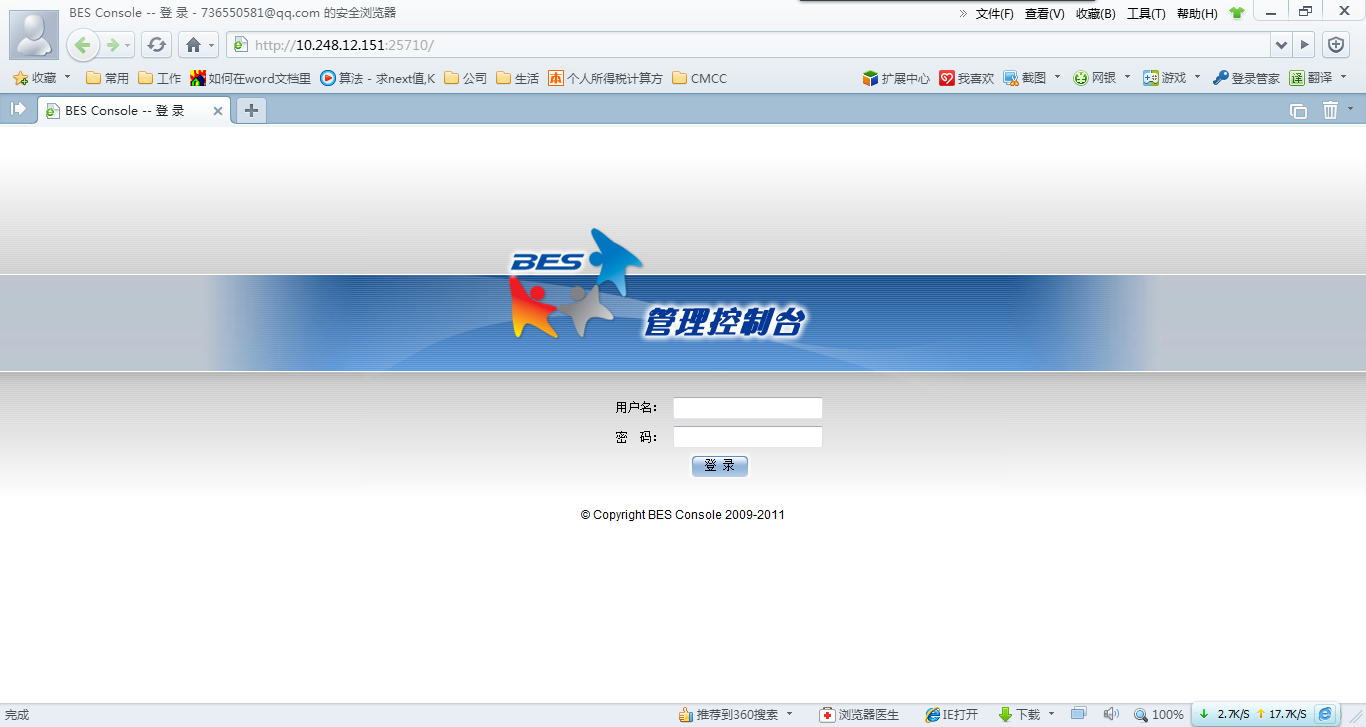
针对一个新建实例，我们部署trans.war应用以后，就能进行业务的正常转发。部署DataSyncClient.war以后就是部署了小M。小M的作用这里就不做详述了！具体的步骤如下：

### 5.1进入bes管理界面。

bes管理访问url在测试环境省端口分布表中有罗列，以手机阅读为例：



图中红色字体部分就是bes管理访问url。点击即可进入bes管理登陆界面，如下图：



用户名/密码：admin/admin

输入用户名密码进入手机阅读bes管理界面，如图：



### 5.2部署解部署应用。

#### 5.2.1部署应用：（前提：应用已经存在本地）

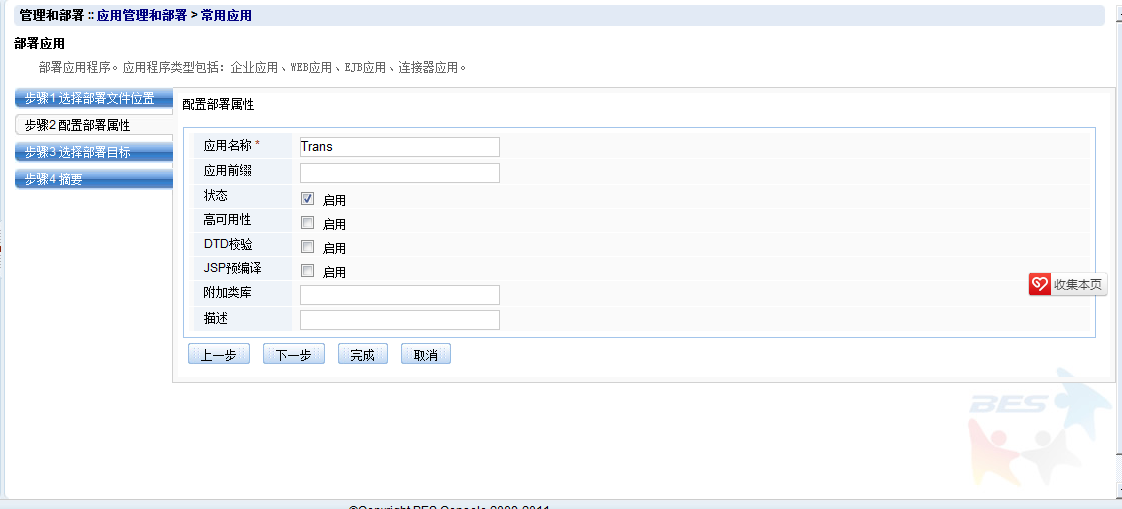
步骤1进入管理和部署-应用管理和部署-常用应用项目下：



步骤2：点击部署，配置如下图所示：类型:WEB应用。位置：选中上传文件，浏览选中要求部署的应用。这里浏览选中的是trans.war文件。点击下一步



步骤3：这一步不需要配置，默认设置就行。继续点击下一步。如图：



步骤4：选择未选择部署目标，点击部署，将其移至已选择部署目标列，点击下一步：具体配置如下图：



步骤5：默认配置就行，点击完成。



在测试环境中应用部署好后自动启动instance，如果出现异常，请查看server.log日志排查问题。

#### 5.2.2解部署：

解部署比较简单，同样在管理和部署-应用管理和部署-常用应用项目下，选中需要解部署的应用，点击解部署就行了。如下图：



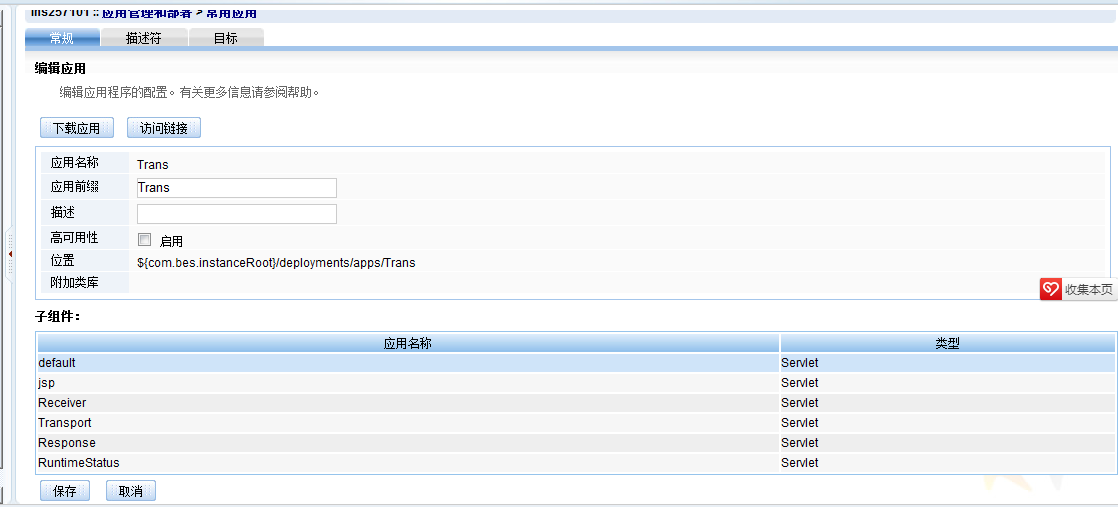
### 5.3 下载应用。

假设已经有已经有你需要的应用在其他实例上运行，那么我们不需要向其他人索取，只要直接下载便可。方法如下：

1在在管理和部署-应用管理和部署-常用应用项目或者ins257101-应用管理和部署-常用应用项目下，点击Trans或者DataSyncClient字段，如图：



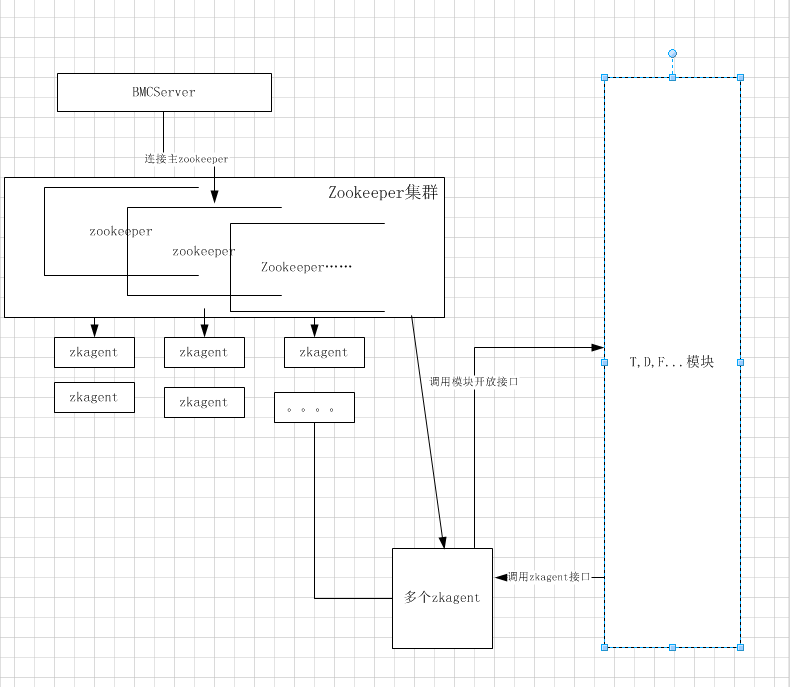
2点击trans字段会调转到如下界面，点击下载应用即可下载。



## 6 刷新对应duns应用内存中的数据（包括数据表和校验文件）

### 6.1 zookeeper与T说明

由于zookeeper取代MC，之前所有T模块与MC模块之间的交互都被取代。此处顺带说明下zookeeper的组织概括和与T之间的功能描述。有兴趣的同事可以参考新版的T模块测试用例，请登录SVN/ecip2下获取资源。下面是我根据历届手工绘制的图纸，仅供参考：

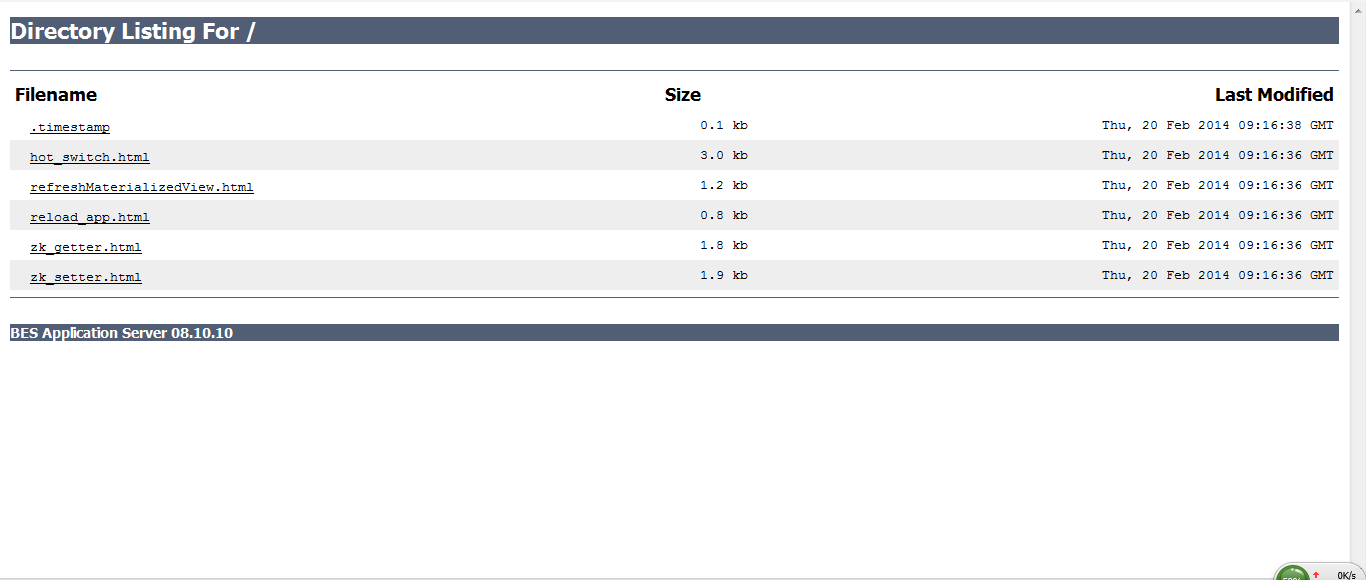


1. zookeeper：zookeeper大家可以看做是类似存储数据的一个模块。通过zookeeper的变动可以触发zkagent对已经提供接口的模块（比如T，D等）进行通知操作。比如内存刷新，日切等。也可以影响BMCServer的监管功能。
2. BMCServer：大家可以看做是以前的MC，
3. 它负责提供接口给外部，执行对zookeeper操作。由于zookeeper的变动可以影响zkagent，因此如果符合逻辑，可以简单的认为，BMC也能够控制zkagent进行操作人员想进行的操作。
4. BMCServer也负责对下属模块的监控，短信告警功能。比如业务处理失败，zkagent异常，数据库异常告警等
5. BMCServer还可以做签到签退，密钥更新操作（需要报文发起触发）和对zookeeper配置数据exe\_para获取和更改等。
6. zkagent：顾名思义，它是具体的操作代理也有自己的定时任务。主要是zkagetn调用模块的对应接口接完成操作。比如get T的runtimestatus接口获取每个T应用的状态，通过post 方式下发全量状态到每个T应用实现状态的监控。

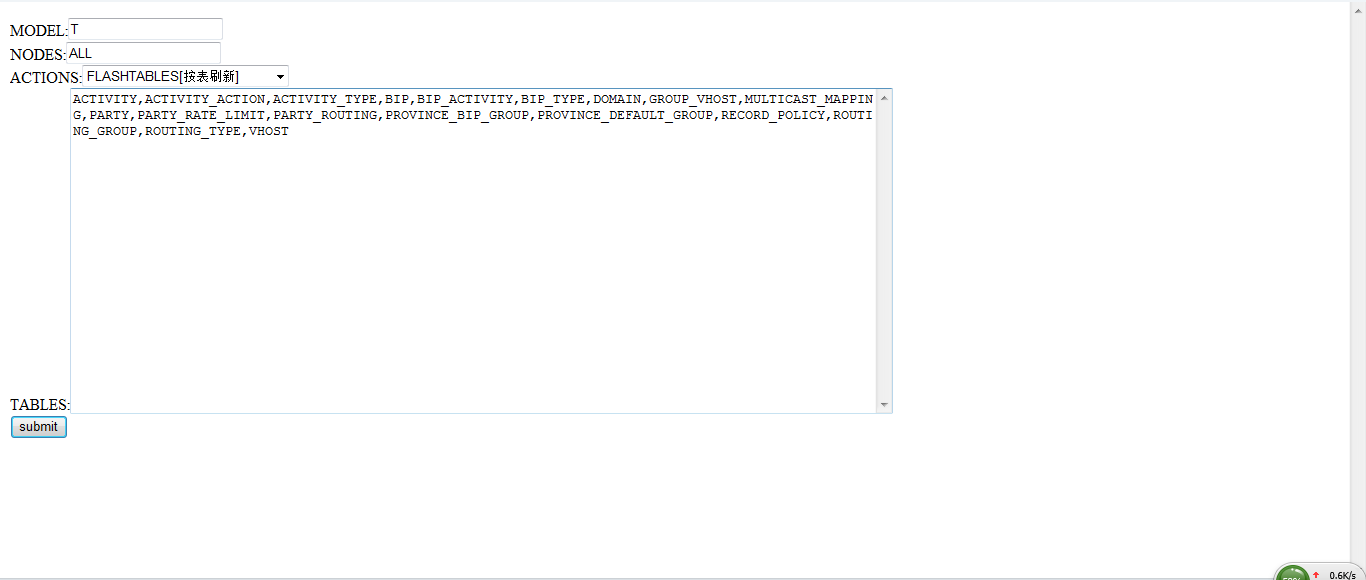
以上是对于三个组成部分的说明，拿内存刷新来说，首先BMCServer提供接口接受请求，如下面6.2第二步，当选择好action等参数，提交给BMCServer之后，BMCServer通过解析，逻辑判断等内部处理之后修改zookeeper中noticeApps 下对应功能的某个字段值（具体字段每个功能不一样而不同）。进而触发zkagent调用下属模块接口，通知应用进行内存刷新操作。其他功能同理。

### 6.2 刷新内存详细操作

1、登陆http://10.248.12.41:1199/BMCServer/



1. 选择refreshMaterializedView.html链接



3. 参数值配置

MODULES：选择模块，由于是T模块操作文档，所以此处选择T

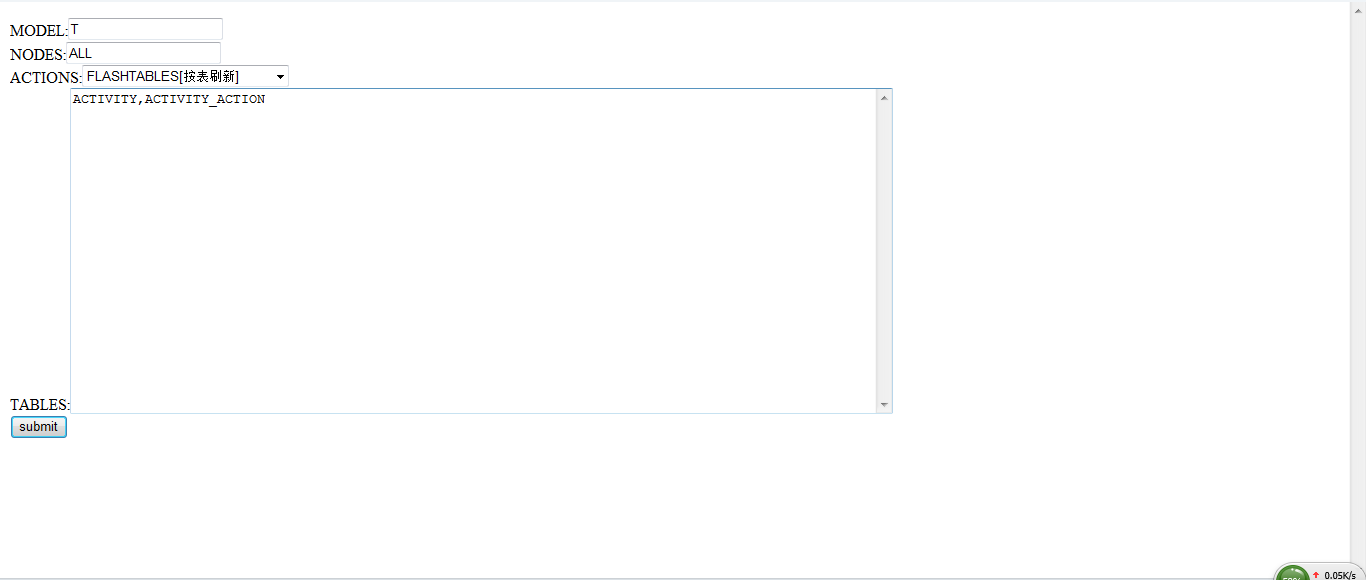
NODES：

这个概念依据zookeeper配置功能reload的exe\_para功能而来。测试环境SN100T代表所有省boss，SN200T代表所有平台。如果后期151配置新端口则需要维护exe\_para，以保证新配置应用被包含。

ACTIONS:

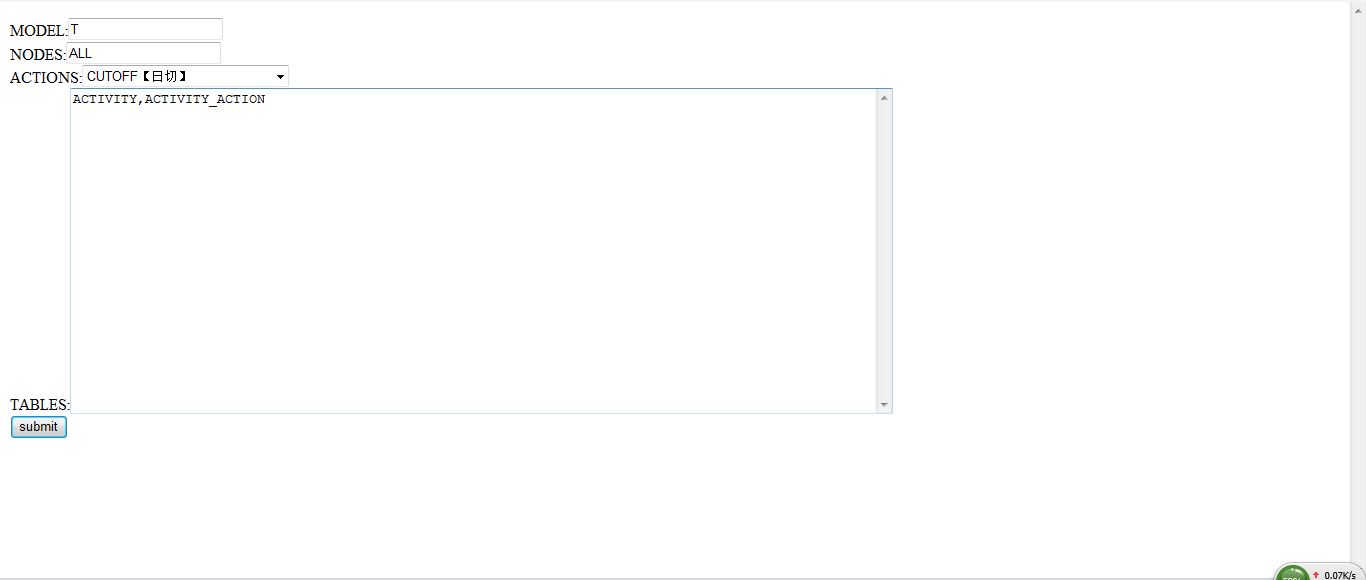
FLASHTABLES[按表刷新]:

当选择该选项时，意味着需要刷新TABLES里面的物化视图（默认全选所有可填物化视图名称）,可以修改默认值，使用逗号分开。如下图：



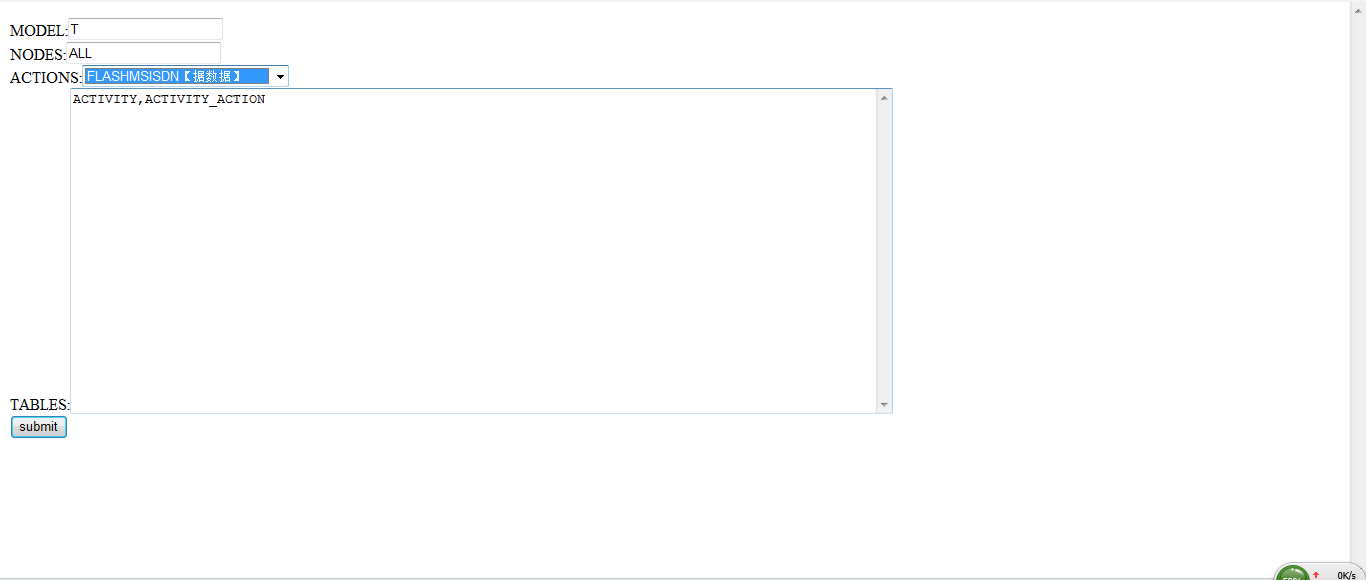
CUTOFF【日切】：

此选项代表刷新日切sys\_para表，一般日切出现异常的时使用该选项。TABLES里面的值不为空即可（该值此时无效）。如下图：



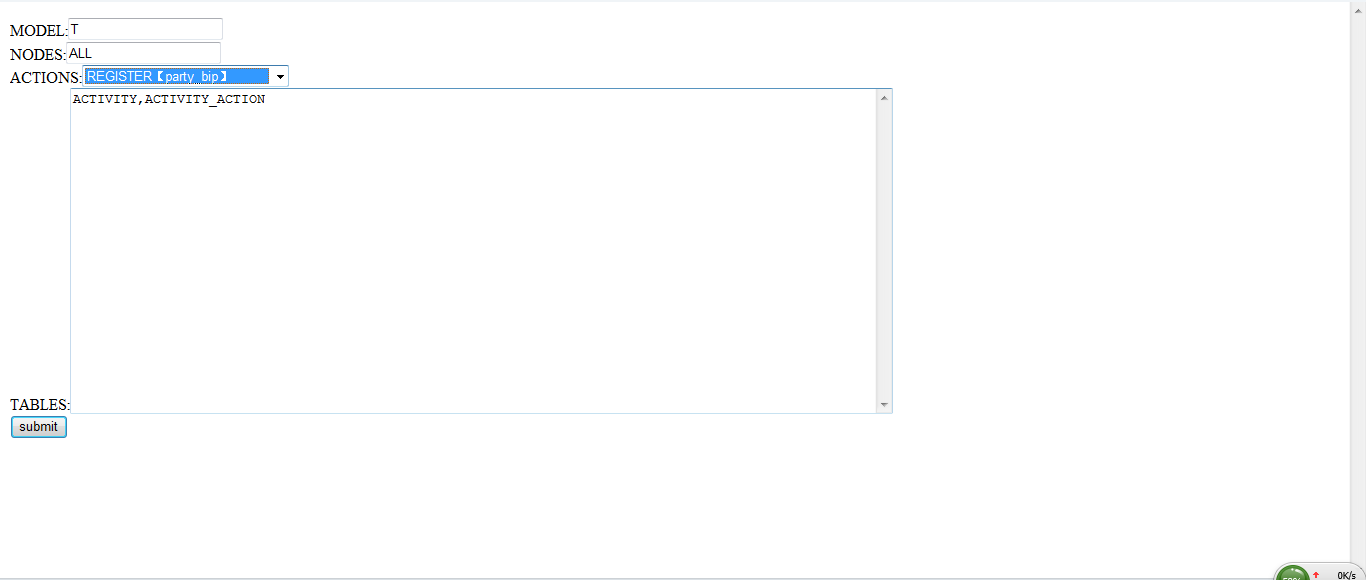
FLASHMSISDN【据数据】：

选择该选项，意味着刷新局数据MSISDN\_AREA\_MAP表。只有局数据出现更新时才执行该选项。TABLES里面的值不为空即可（该值此时无效）。如下图：



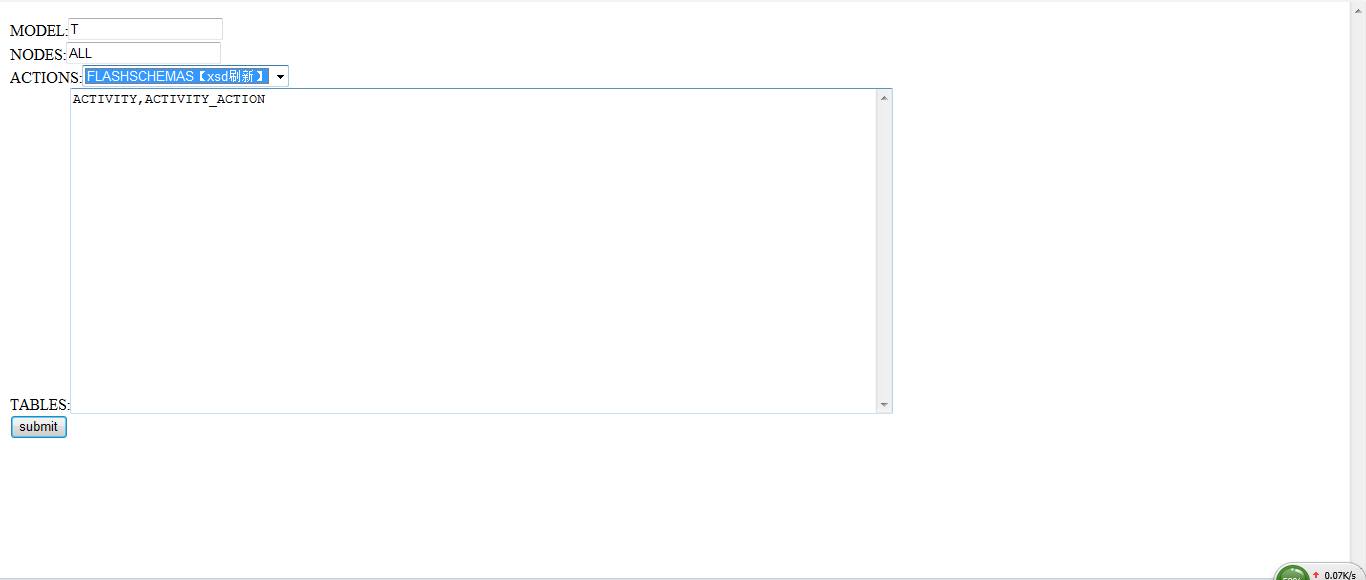
REGISTER【party\_bip】

选择该选项，意味着刷新PARTY\_BIP表。只有该表需要更新时才执行该选项。TABLES里面的值不为空即可（该值此时无效）。如下图：



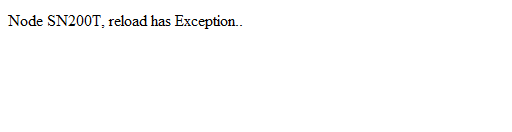
FLASHSCHEMAS【xsd刷新】

选择该选项，意味着刷新xsd校验文件。只有校验文件有改变时才执行该选项。TABLES里面的值不为空即可（该值此时无效）。如下图：

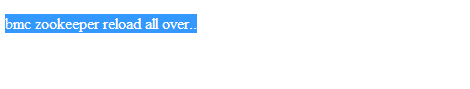


查看页面反馈，由于151资源紧张，刷新所有应用时，部分平台应用处于停止状态，因此会有异常反馈。假如选择SN100T则不会出现异常的现象。

刷新所有应用页面反馈图如下：



刷新SN100T（所有38,41应用）页面反馈图如下：



1. 随机抽取某个省数据库查看是否达到自己的预期效果。

假如只是想刷新某一个省的数据请使用“参数发起zookeeper&T.html”工具选择发送。发起参数类似BMCServer发起，version自动会变更。次方式只时候测试环境。

## 7 发起报文验证平台配置是否成功。

通过xmlspy工具根据xsd文件生产报文样本，使用test\_tool.html发起对应activity的报文到落地方，验证所有配置是否成功。如果失败请检查1-5是否都已正确完成。

## 8 PL/SQL 中现有数据表操作SQL语句导出。

PL/SQL可以对现有的数据表操作SQL语句的导出。这样就能够保证数据一致地从测试数据库迁移到生产数据库的数据。这里主要是对与上线平台数据相关的插入SQL语句的导出。下面我们以彩云平台为例进行操作。首先我们需要明确彩云平台的一些信息：

Domain：MCLD

Party：MCLD2000

Bip： **BIP2B076，BIP2B077**

Activity：**T2001069，T2101075，T2101076**

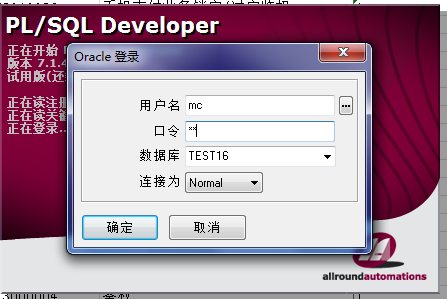
这些信息在导出SQL语句的时候作为限制条件。

在新平台上测试环境的时候，我们配置过数据库。现在我们需要逆向的取出这些我们插入过的语句。但是由于生产环境已经上线部分平台以及生产和测试的差异，我们只要对下面几张表的数据进行导出，保留以备生产上线使用。（这里最好是在联调完成以后再导出SQL）。

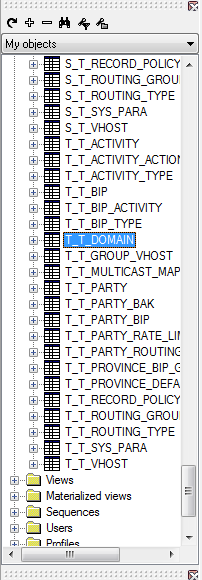
### 8.1 导出t\_t\_domain 表插入语句。

PL/SQL 如何导出t\_t\_domain的具体步骤和图解如下：

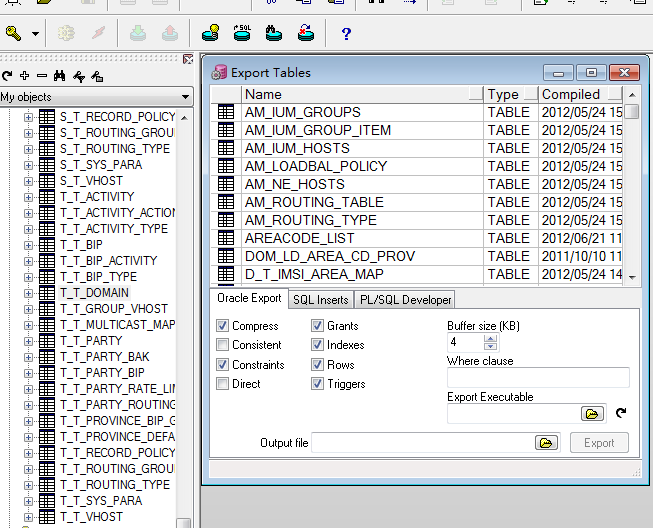
1. 打开PL/SQL，使用MC用户登录测试数据库test16.如下图所示：



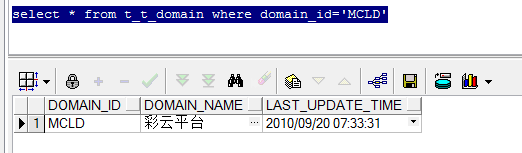
1. 登入数据库以后，在浏览器中设置显示My object.（这步可以省略），然后选中table项下t\_t\_domain表。

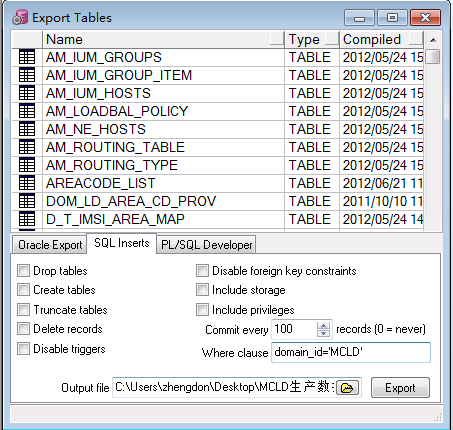


1. 敲击右键选择export data（导出数据），如下图所示

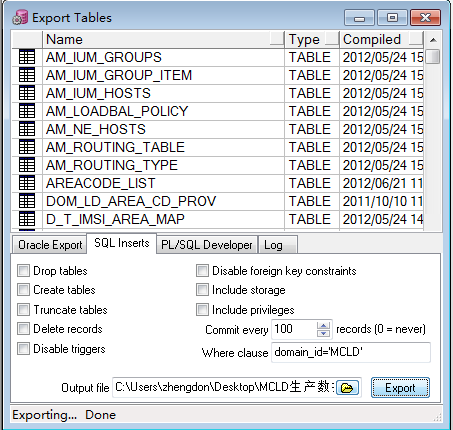


1. 选中SQL insert，在where clause中输入 domain\_id=’MCLD’(限制条件的确定，最好使用select语句对表进行查询以后再确定where clause)，在output file中选择所要导出的位置并命名SQL文件。其他都不需要勾选。如下图：

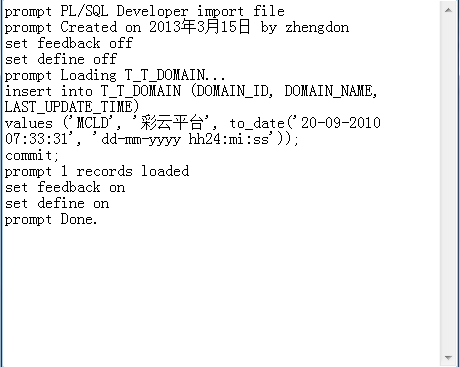




1. 点击export按钮，PL/SQL 提示导出完成。



6)打开输出目录下的sql文件，检查插入语句和数值是否有问题。否则检查之前步骤是否正确执行。



### 8.2 导出t\_t\_activity中关于彩云平台MCLD的相关数据。

导出activity表需要知道平台涉及到的所以activity，所以导出t\_t\_activity中彩云平台数据时，我们设置的where clause 应该为：

activity\_id in ('T2101075','T2001069','T2101076')



### 8.3 导出t\_t\_activity\_action关于彩云平台MCLD的相关数据。

同t\_t\_activitywhere clause 应该设置为：

activity\_id in ('T2101075','T2001069','T2101076')

### 8.4 导出t\_t\_bip关于彩云平台MCLD的相关数据。

根据MCLD所涉及的bip，我们设置where clause 为：

bip\_id in ('BIP2B076','BIP2B077')，如图



### 8.5 导出t\_t\_bip\_activity 关于彩云平台MCLD的相关数据。

导出SQL语句的限定条件同7.4.。

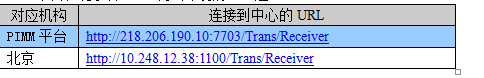
### 8.6导出t\_t\_party\_bip关于彩云平台MCLD的相关数据。

导出SQL语句的限定条件同7.4.

## 9 新平台接入测试环境配置文档提交

由于统计工作量和配合集团检查工作，针对业支负责平台需要提供相关的开发部署文档和内部测试计划文档。对于不同的平台这两个文档需要注意一下几项：

1. 撰写时间
2. 撰写人
3. 参考文档
4. 业务表
5. 校验文件名
6. 平台名包括缩写
7. 测试上发端口：存在业务来往的party。大部分平台只和省boss交互，但是某些特殊平台会存在平台和平台交互，省和省交互的现象。请根据具体情况完成测试地址的编写。如果存在平台与平台交互那么应该有两个平台URL，如果存在省与省交互请配置两个省的URL。



## 11、注意事项

为了规范测试环境的操作以及方便管理。请大家注意如下几项：

1. 新平台配置和检查请大家尽量仔细，因为后续平台分配给个人后，之后的变更都会由这个人负责。主要是为了防止多人变更，最终找不到了解的人。所以对于负责人来说，检查配置仔细，会减少后续自己的工作量。对于其他同事来说，发现问题的时候有个方向找人解决。计划将来上线，配置生产数据之后的检查由负责人检查。
2. 联调过程中，大家发现问题时，如果问题紧急，可以向上报，解决之后再自己分析和总结。如果问题不紧急，请找同事讨论，自己分析和解决，从而达到进步的过程。
3. 对于SVN来说，大家在有xsd，文档，源码，版本，规范等规划到SVN上的资源更新时，请及时上发到SVN上，方便其他同事取用。规范，联调案例，上线计划是测试组负责上传。开发部署文档，内部测试计划文档，xsd以及生产上线统计结果由平台负责人上传。（上线结果一般由上线的人保存，负责人可以让上线人上传或者自己索取再上传）。
4. 大家被分配工作时，如果实在是没时间，请反馈到讨论组找其他同事帮帮忙。但是完事后记得自己检查下哦，毕竟后续都是找负责人的。

希望大家都能轻松愉快的工作。

## 12、附录

个人理解，可能有错误。希望新同事对于沟通时不清楚的概念或者刚入手时不明白的问题提出来，方便文档的更新和维护，为将来的新同事提供一份热心。

|  |  |
| --- | --- |
| 名词 | 解释 |
| 38机器 | 10.248.12.38 测试机。部分省boss应用部署其上,具体请参考“测试环境省端口分配表.xlsx” |
| 41机器 | 10.248.12.41 测试机。部分省boss应用部署其上，具体请参考“测试环境省端口分配表.xlsx” |
| 151机器 | 10.248.12.151 测试机。除了BBOSS，其他所有平台部署其上 |
| server日志 | 每个Instance都有自己的日志，该日志记录了instance本身的日志以及所有部署其上的应用的日志。 比如Trans应用的日志。Server日志在3台机器的路劲基本相似。 38：/opt/ecip/bes/var/domains/domainxxxx/nodes/nodexxxx/instances/insxxxx/logs 41：/opt/ecip/T/var/domains/domainxxxx/nodes/nodexxxx/instances/insxxxx/logs 151：/1bosstest2/ecip/T/bes/var/domains/domainxxxx/nodes/nodexxxx/instances/insxxxxx/logs xxxx:代表duns号 |
| oltp日志 | oltplogs，一般位于T目录下.路径为：$ECIP\_HOME/oltplogs。分别分osn和hsn两类日志，前者是该应用发起报文日志记录，后者是该应用接受报文日志记录 |
| domain | 依照BES而来，如果新的duns出现，并且需要给该duns安装一个T的应用。那么就需要新建一个domain，domain下面可以安装多个node，以现在T来说只有一个node。 当domain创建之后，就可以使用浏览器进行访问了。Domain命名：domain+duns号。 |
| node | 创建T的node时，注意名称为：node+duns号 |
| instance | node下面应用安装的载体。一个应用对应一个http端口。接受报文的端口可配置。省端口分配表“报文发送URL”列中的端口即是instance的http端口 |
| group | group是逻辑上的概念，这是数据库设计中的概念。一个group中包含多个vhost，以达到数据的平台和管控。 |
| vhost | vhost其实可以理解为hsn的url，url代表的是Trans应用。由于一个应用只有一个Trans，所以可以理解为vhost和Instance是对应一对一关系 |
| duns | 省代码+0 |
| bes | 在项目中，bes是web应用安装的载体，具体可以百度下 |
| BMC配置文件 | 41机器。路径：/opt/ecip/BMC/bmcServerConf/config.conf。BMCServer主要实现对各个模块状态的监控，管理，告警等功能。 |
| BMC日志 | 41机器，路径：/opt/ecip/BMC/bmclogs/bmcLog.log |
| zkagent日志 | 进入agent安装目录，一般在MC目录下的zkagent目录下。测试环境38,151各一个agent。Agent日志显示agent运行日志，可以设置info和debug两个级别。 |
| 启动zkagent | 进入agent目录，运行run.sh。 |
| zookeeper启动 | 38机器，/opt/ecip/BMC/zookeeper/bin/zkCli.sh 回车即可 |
| grep | grep 关键字 文件名。类似于在某个文件查询关键字，将结果显示出来 |
| vi | 编辑文件。具体用法man vi查看，或者百度，谷歌 |
| cd | cd 目录。进入目录。 |
| exe\_para | exe\_para是zokeeper配置的功能的配置信息。比如刷新数据集功能的exe\_para就包含了每个T应用的api接口的url。 |