

用户数据同步说明文档

基于数据交换

临时表说明

- 组织机构表

TMP_ORGANIZATION_ORIGIN

名称	类型	描述	是否必填
ID	varchar(64)	ID，主键，随机生成	必填
DELETED	int(1)	是否删除 0：有效 1：删除	必填
ADD_ACCOUNT	varchar(100)	创建人用户名	
ADD_TIME	timestamp	创建时间	
EDIT_ACCOUNT	varchar(100)	修改人用户名	
EDIT_TIME	timestamp	修改时间	
DELETE_ACCOUNT	varchar(100)	删除人用户名	
DELETE_TIME	timestamp	删除时间	
PARENT_ORGANIZATION_ID	varchar(64)	所属的父级组织机构，顶级组织机构为 0	
CODE	varchar(120)	组织机构代码	必填
NAME	varchar(200)	组织机构名称	必填
DESCRIPTION	varchar(500)	组织机构描述	
TYPE_ID	varchar(64)	组织机构部门类型，字典表ID、代码均可	必填
STATE	int(2)	状态 [0:正常 1:冻结 2:注销];默认正常	必填
ENABLE	int(1)	是否启用 1：启用 0：不启用;默认启用	
IS_DATA_CENTER	int(1)	是否来源数据中心 [0：否 1：是]；默认是	必填

- 帐号表

TMP_ACCOUNT_ORIGIN

名称	类型	描述	是否必填
ID	varchar(64)	ID，主键，随机生成	必填
DELETED	int(1)	是否删除 0：有效 1：删除	必填
ADD_ACCOUNT	varchar(100)	创建人用户名	
ADD_TIME	timestamp	创建时间	

EDIT_ACCOUNT	varchar(100)	修改人用户名	
EDIT_TIME	timestamp	修改时间	
DELETE_ACCOUNT	varchar(100)	删除人用户名	
DELETE_TIME	timestamp	删除时间	
UID	varchar(64)	唯一编号	必填
NAME	varchar(255)	姓名	必填
NAME_SPELLING	varchar(120)	姓名简拼	
FULL_NAME_SPELLING	varchar(255)	姓名全拼	
CERTIFICATE_TYPE_ID	varchar(64)	证件类型，字典表ID、代码均可	必填
CERTIFICATE_NUMBER	varchar(120)	证件号码	必填
PHONE_NUMBER	varchar(50)	手机号码	
EMAIL	varchar(50)	邮箱	
IMAGE_URL	varchar(255)	个人照片地址	
GENDER_ID	varchar(64)	性别，字典表ID、代码均可	
NATION_ID	varchar(64)	民族，字典表ID、代码均可	
COUNTRY_ID	varchar(64)	国家，字典表ID、代码均可	
ADDRESS_ID	varchar(64)	地区，字典表ID、代码均可	
ACCOUNT_NAME	varchar(120)	人员账号	必填， 唯一
ACCOUNT_EXPIRY_DATE	date	账号有效日期[yyyy-MM-dd]	
ORGANIZATION_ID	varchar(64)	所属组织机构	必填
IDENTITY_TYPE_ID	varchar(64)	身份类型，身份表ID、代码均可	必填
ACTIVATION	int(1)	是否激活 [0:未激活 1： 激活]	必填
STATE	varchar(20)	账号状态[NORMAL:正常 FREEZE:冻结 WRITTENOFF:注销] 默认正常状态	必填
IS_DATA_CENTER	int(1)	是否来源数据中心 [0： 否 1： 是]；默认是	必填

添加触发器

需在临时表上，增加更新触发器，执行以下脚本

```
DROP TRIGGER IF EXISTS after_update_organization_origin;
```

```

delimiter //
create trigger after_update_organization_origin after update on
TMP_ORGANIZATION_ORIGIN for each row
begin
    declare ID1 varchar(100);
    declare ID2 varchar(100);

    -- new 代表 表中新增的数据
    set ID1 = (select ID from TMP_ORGANIZATION_TRANS
    where ((ID is null and new.ID is null) or ID=new.ID)
        and ((PARENT_ORGANIZATION_ID is null and new.PARENT_ORGANIZATION_ID is null) or
PARENT_ORGANIZATION_ID=new.PARENT_ORGANIZATION_ID)
        and ((CODE is null and new.CODE is null) or CODE=new.CODE)
        and ((NAME is null and new.NAME is null) or NAME=new.NAME)
        and ((DESCRIPTION is null and new.DESCRPTION is null) or
DESCRIPTION=new.DESCRPTION)
        and ((TYPE_ID is null and new.TYPE_ID is null) or TYPE_ID=new.TYPE_ID)
        and ((STATE is null and new.STATE is null) or STATE=new.STATE)
        and ((ENABLE is null and new.ENABLE is null) or ENABLE=new.ENABLE)
        and ((IS_DATA_CENTER is null and new.IS_DATA_CENTER is null) or
IS_DATA_CENTER=new.IS_DATA_CENTER)
    );
    -- into @ID1;

    if ID1 is null then
        set ID2 = (select ID from TMP_ORGANIZATION_TRANS where ID=new.ID); -- into @ID2;

        if ID2 is null then
            insert into TMP_ORGANIZATION_TRANS(TRANS_STATUS, TRANS_TIME, PROC_STATUS,
PROC_TIME,
            ID, PARENT_ORGANIZATION_ID,
            CODE, NAME, DESCRIPTION,
            TYPE_ID,
            STATE, ENABLE,
            IS_DATA_CENTER
            )
            values ('1', now(), '0', null,
            new.ID, new.PARENT_ORGANIZATION_ID,
            new.CODE, new.NAME, new.DESCRPTION,
            new.TYPE_ID,
            new.STATE, new.ENABLE,
            new.IS_DATA_CENTER
            )
            ;

        else
            update TMP_ORGANIZATION_TRANS set
            TRANS_STATUS='2',
            TRANS_TIME=now(),

```

```

PROC_STATUS='0',
PARENT_ORGANIZATION_ID=new.PARENT_ORGANIZATION_ID,
CODE=new.CODE,
NAME=new.NAME,
DESCRIPTION=new.DESCRPTION,
TYPE_ID=new.TYPE_ID,
STATE=new.STATE,
ENABLE=new.ENABLE,
IS_DATA_CENTER=new.IS_DATA_CENTER
where ID=new.ID
;

end if;

else

-- 如果数据没变化, 但存在记录, 且被处理, 则标记未 不更新、不处理
update TMP_ORGANIZATION_TRANS set
TRANS_STATUS='0',
TRANS_TIME=now(),
PROC_STATUS='0'
where ID=new.ID
and PROC_RESULT!='0'
;

end if;

end //
delimiter ;

DROP TRIGGER IF EXISTS after_update_account_origin;

delimiter //
create trigger after_update_account_origin after update on TMP_ACCOUNT_ORIGIN for each
row
begin
declare ID1 varchar(100);
declare ID2 varchar(100);

-- new 代表 表中新增的数据
set ID1 = (select ID from TMP_ACCOUNT_TRANS
where ((ID is null and new.ID is null) or ID=new.ID)
and ((UID is null and new.UID is null) or UID=new.UID)
and ((NAME is null and new.NAME is null) or NAME=new.NAME)
and ((NAME_SPELLING is null and new.NAME_SPELLING is null) or
NAME_SPELLING=new.NAME_SPELLING)
and ((FULL_NAME_SPELLING is null and new.FULL_NAME_SPELLING is null) or
FULL_NAME_SPELLING=new.FULL_NAME_SPELLING)

```

```

    and ((CERTIFICATE_TYPE_ID is null and new.CERTIFICATE_TYPE_ID is null) or
CERTIFICATE_TYPE_ID=new.CERTIFICATE_TYPE_ID)
    and ((CERTIFICATE_NUMBER is null and new.CERTIFICATE_NUMBER is null) or
CERTIFICATE_NUMBER=new.CERTIFICATE_NUMBER)
    and ((PHONE_NUMBER is null and new.PHONE_NUMBER is null) or
PHONE_NUMBER=new.PHONE_NUMBER)
    and ((EMAIL is null and new.EMAIL is null) or EMAIL=new.EMAIL)
    and ((IMAGE_URL is null and new.IMAGE_URL is null) or IMAGE_URL=new.IMAGE_URL)
    and ((GENDER_ID is null and new.GENDER_ID is null) or GENDER_ID=new.GENDER_ID)
    and ((NATION_ID is null and new.NATION_ID is null) or NATION_ID=new.NATION_ID)
    and ((COUNTRY_ID is null and new.COUNTRY_ID is null) or COUNTRY_ID=new.COUNTRY_ID)
    and ((ADDRESS_ID is null and new.ADDRESS_ID is null) or ADDRESS_ID=new.ADDRESS_ID)
    and ((ACCOUNT_NAME is null and new.ACCOUNT_NAME is null) or
ACCOUNT_NAME=new.ACCOUNT_NAME)
    and ((ACCOUNT_EXPIRY_DATE is null and new.ACCOUNT_EXPIRY_DATE is null) or
ACCOUNT_EXPIRY_DATE=new.ACCOUNT_EXPIRY_DATE)
    and ((ORGANIZATION_ID is null and new.ORGANIZATION_ID is null) or
ORGANIZATION_ID=new.ORGANIZATION_ID)
    and ((IDENTITY_TYPE_ID is null and new.IDENTITY_TYPE_ID is null) or
IDENTITY_TYPE_ID=new.IDENTITY_TYPE_ID)
    and ((ACTIVATION is null and new.ACTIVATION is null) or ACTIVATION=new.ACTIVATION)
    and ((STATE is null and new.STATE is null) or STATE=new.STATE)
    and ((IS_DATA_CENTER is null and new.IS_DATA_CENTER is null) or
IS_DATA_CENTER=new.IS_DATA_CENTER)
);
-- into @ID1;

if ID1 is null then
    set ID2 = (select ID from TMP_ACCOUNT_TRANS where ID=new.ID); -- into @ID2;

    if ID2 is null then
        insert into TMP_ACCOUNT_TRANS(TRANS_STATUS, TRANS_TIME, PROC_STATUS,
PROC_TIME,
        ID, UID,
        NAME, NAME_SPELLING, FULL_NAME_SPELLING,
        CERTIFICATE_TYPE_ID, CERTIFICATE_NUMBER,
        PHONE_NUMBER, EMAIL,
        IMAGE_URL,
        GENDER_ID, NATION_ID, COUNTRY_ID, ADDRESS_ID,
        ACCOUNT_NAME, ACCOUNT_EXPIRY_DATE, ORGANIZATION_ID, IDENTITY_TYPE_ID,
        ACTIVATION, STATE,
        IS_DATA_CENTER
        )
        values ('1', now(), '0', null,
        new.ID, new.UID,
        new.NAME, new.NAME_SPELLING, new.FULL_NAME_SPELLING,
        new.CERTIFICATE_TYPE_ID, new.CERTIFICATE_NUMBER,
        new.PHONE_NUMBER, new.EMAIL,
        new.IMAGE_URL,

```

```

        new.GENDER_ID, new.NATION_ID, new.COUNTRY_ID, new.ADDRESS_ID,
        new.ACCOUNT_NAME, new.ACCOUNT_EXPIRY_DATE, new.ORGANIZATION_ID,
new.IDENTITY_TYPE_ID,
        new.ACTIVATION, new.STATE,
        new.IS_DATA_CENTER
    )
;

```

else

```

update TMP_ACCOUNT_TRANS set
    TRANS_STATUS='2',
    TRANS_TIME=now(),
    PROC_STATUS='0',
    UID=new.UID,
    NAME=new.NAME,
    NAME_SPELLING=new.NAME_SPELLING,
    FULL_NAME_SPELLING=new.FULL_NAME_SPELLING,
    CERTIFICATE_TYPE_ID=new.CERTIFICATE_TYPE_ID,
    CERTIFICATE_NUMBER=new.CERTIFICATE_NUMBER,
    PHONE_NUMBER=new.PHONE_NUMBER,
    EMAIL=new.EMAIL,
    IMAGE_URL=new.IMAGE_URL,
    GENDER_ID=new.GENDER_ID,
    NATION_ID=new.NATION_ID,
    COUNTRY_ID=new.COUNTRY_ID,
    ADDRESS_ID=new.ADDRESS_ID,
    ACCOUNT_NAME=new.ACCOUNT_NAME,
    ACCOUNT_EXPIRY_DATE=new.ACCOUNT_EXPIRY_DATE,
    ORGANIZATION_ID=new.ORGANIZATION_ID,
    IDENTITY_TYPE_ID=new.IDENTITY_TYPE_ID,
    ACTIVATION=new.ACTIVATION,
    STATE=new.STATE,
    IS_DATA_CENTER=new.IS_DATA_CENTER
where ID=new.ID
;

```

end if;

else

-- 如果数据没变化，但存在记录，且被处理，则标记未 不更新、不处理

```

update TMP_ACCOUNT_TRANS set
    TRANS_STATUS='0',
    TRANS_TIME=now(),
    PROC_STATUS='0'
where ID=new.ID
    and PROC_RESULT!='0'
;

```

```
end if;

end //
delimiter ;
```

字典说明

若组织机构数据、人员数据中使用的字典与以下现有字典 不一致的，须进行重新初始化

- 组织机构类型

代码	名称	是否可调整
10000	根组织机构	不要删除、调整
10001	学校	
10002	部门	
10003	院系	
10004	专业	
10005	班级	

注，除 10000 外，其他组织机构类型代码都可以视项目情况，删除，然后新增

- 证件类型

采用标准代码

代码	名称
1	居民身份证
2	军官证
3	士兵证
4	文职干部证
5	部队离退休证
6	香港特区护照/身份证明
7	澳门特区护照/身份证明
8	台湾居民来往大陆通行证
9	境外永久居住证
A	护照
B	户口簿
C	外国人永久居留证
Z	其他

● 性别

采用标准代码

代码	名称
0	未知的性别
1	男
2	女
9	未说明的性别

● 民族

采用标准代码

代码	名称
01	汉族
02	蒙古族
03	回族

04	藏族
05	维吾尔族
06	苗族
07	彝族
08	壮族
09	布依族
10	朝鲜族
11	满族
12	侗族
13	瑶族
14	白族
15	土家族
16	哈尼族
17	哈萨克族
18	傣族
19	黎族
20	傈僳族
21	佤族
22	畲族
23	高山族
24	拉祜族
25	水族
26	东乡族
27	纳西族
28	景颇族
29	柯尔克孜族
30	土族
31	达斡尔族
32	仫佬族

33	羌族
34	布朗族
35	撒拉族
36	毛难族
37	仡佬族
38	锡伯族
39	阿昌族
40	普米族
41	塔吉克族
42	怒族
43	乌孜别克族
44	俄罗斯族
45	鄂温克族
46	德昂族
47	保安族
48	裕固族
49	京族
50	塔塔尔族
51	独龙族
52	鄂伦春族
53	赫哲族
54	门巴族
55	珞巴族
56	基诺族
81	穿青人族
97	其他
98	外国血统中国籍人士

● 国家

采用标准代码

代码	名称
156	中国

- 地区

采用标准代码，代码过多，此处只列出若干，详细可见系统的字典管理功能

代码	名称
110000	北京市
110100	市辖区
110101	东城区
110102	西城区
120000	天津市
120100	市辖区
120101	和平区
120102	河东区
120103	河西区
310000	上海市
310100	市辖区
310101	黄浦区