

1 UA3101R - 用户信息维护服务组件

本组件提供用户基本信息和补充信息的新增和修改等维护功能, 接收接口参数后, 经过必要的用户存在性、所属机构存在性、签到状态、证件类型、证件代码、性别、身份认证方式等合法性校验后, 将其维护到【用户基本信息表】和【密码信息表】中。

1.1 资源关联情况

1. 数据库表使用情况

编号	表名	表描述	表操作方式 (I/U/A/O)	键字结构名	是否参与事务
1	UAUSBA01	用户基本信息表	U/A	DS_UAUSB_K01	是
2	UAUSPA01	用户密码信息表	A	DS_UAUSP_K01	是

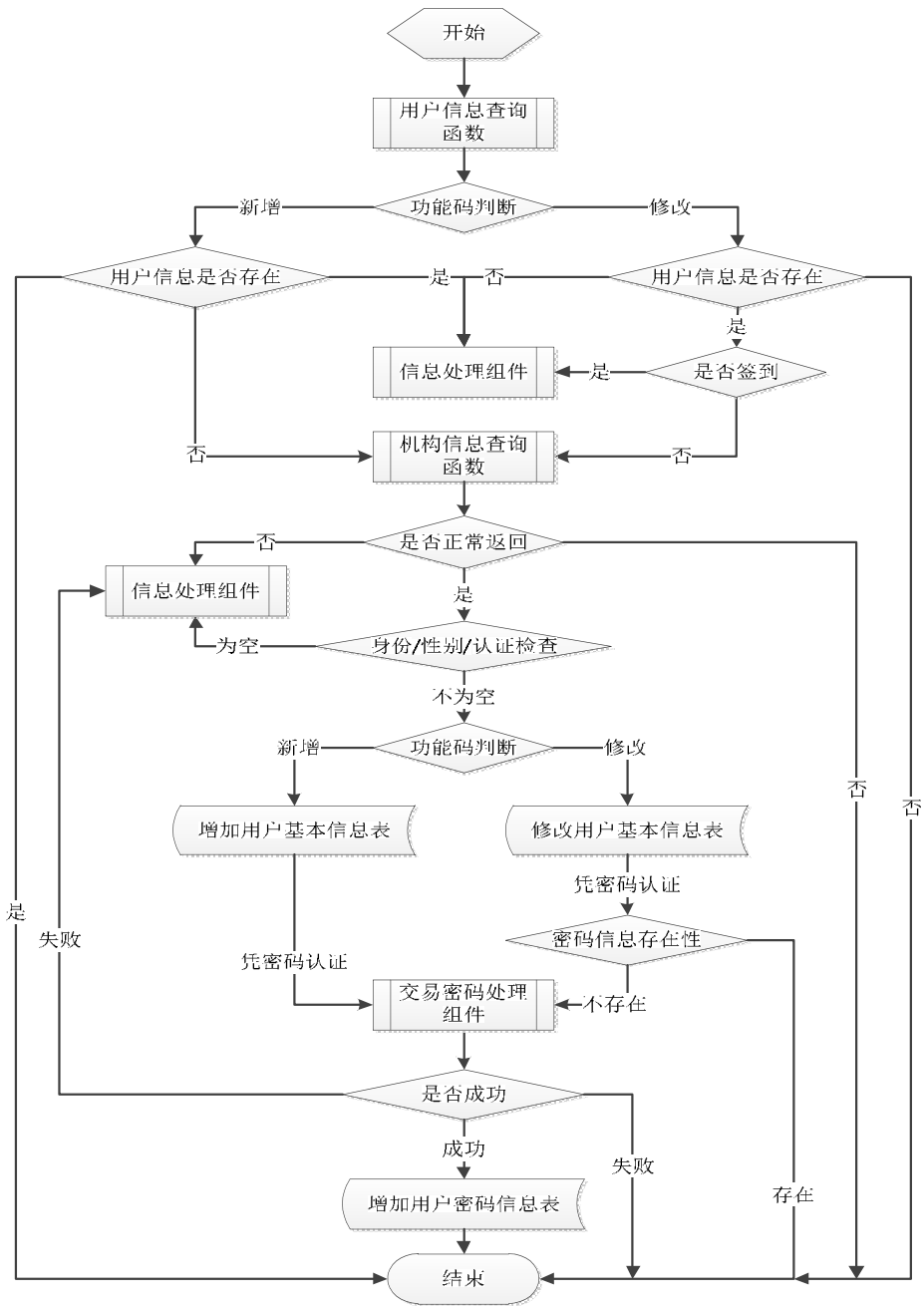
2. 数据结构/变量/常量使用情况

编号	名称	描述	内容
1	SCCTWA	交易工作区	接收联机交易的公有域信息, 参见技术平台
2	UA3101I	组件调用输入接口	接收私有域的信息, 参见本节接口描述
3	UA3101IS	组件调用标准输入接口	参见本节接口描述
4	DS_UAUSB_K01	UAUSBA01 表键字结构	
5	DS_UAUSP_K01	UAUSPA01 表键字结构	
6	UAINQUSR	用户信息查询函数调用接口	获取用户信息并检查, 见接口描述
7	ORINQORG	机构信息查询函数调用接口	获取机构信息并检查, 见机构管理
8	SECTRANPW	交易密码处理组件接口	生成用户初始密码, 参见技术平台
9	DS_UAUSB	UAUSBA01 结构	BASED (PTR_UAUSB) 接收传入参数到 UAUSBA 表
10	DS_UAUSB_IS	UAUSBA01 结构	PREFIX (IS:3)

			接收传入参数
11	L_DIGITS_ALL	0 到 9 的数字	0123456789

1.2 处理描述

1. 处理流程图



2. 主处理描述

1) 初始化

//数据初始化

EXSR @SR_INZDTA ;

2) 输入合法性检查

//输入参数合法性检查

EXSR @SR_CHKDTA ;

3) 主处理

//根据不同操作进行处理

EXSR @SR_MAIN ;

4) 结束处理

EXSR @SR_RETURN ;

3. 子例程描述

1) @SR_INZDTA:数据初始化

BEGSR @SR_INZDTA ;

//获取接口指针

PTR_TWA = %ADDR(PI_TWA) ;

PTR_UA3101I = %ADDR(PI_UA3101I) ;

PTR_UA3101IS = %ADDR(UA3101I.IS) ;

ENDSR ;

2) @SR_CHKDTA:输入参数合法性检查

BEGSR @SR_CHKDTA ;

//用户存在性检查

//调用户信息查询函数，传入用户号，检查用户是否存在

```
UAINQUSRIS.ISUSNO = UA3101IS.ISUSNO ; //用户号

CALLP UAINQUSR UAINQUSR ;

//分不同的操作，对用户信息进行检查

SELECT ;

    WHEN UA3101IS.ISFUNC = 'C' ; //参数操作功能码= '增加'

        //增加时，用户存在性检查

        IF %SUBST(UAINQUSRRES.ESRTCD:3:1) <> 'R' AND

            UAINQUSRRES.ESUSST <> 'C' ; //用户存在并且不是注销的

            XC_MGID = 'UARA005' ; //&1 已存在

            XC_AINF = XC_SYPN + '用户信息' ;

            //@_@抛错误信息 *PGM

        ENDIF ;

    WHEN UA3101IS.ISFUNC = 'U' ; //参数操作功能码= '修改'

        //修改时，用户存在性检查

        IF %SUBST(UAINQUSRRES.ESRTCD:3:1) = 'R' OR

            UAINQUSRRES.ESUSST = 'C' ; //用户不存在或者已注销

            XC_MGID = 'UARA004' ; //&1 不存在

            XC_AINF = XC_SYPN + '用户信息' ;

            //@_@抛错误信息 *PGM

        ELSE ; //用户存在并且未注销时，需要进行以下检查

            //签到状态检查：修改用户信息时，该用户必须已签退

            IF UAINQUSRRES.ESUIFG <> 'F' ; //不是已签退

                XC_MGID = 'UARA010' ; //用户尚未签退，不能修改

                XC_AINF = XC_SYPN ;

                //@_@抛错误信息并继续

            ENDIF ;

        ENDIF ;

    OTHER ; //不是增加和修改的操作码，做错误处理
```

```
XC_MGID = 'UARA006' ; // &1 错误

XC_AINF = XC_SYPN + '操作类型' ;

// @@ 抛错误信息 *PGM

ENDSL ;

// 所属机构检查

// 调机构信息查询函数，传入用户所属机构

ORINQORGIS.ISBRNO = UA3101IS.ISBLBR ; // 所属机构

CALLP ORINQORGR ORINQORG ;

IF %SUBST(ORINQORGES.ESRTCD:3:1) = 'R' ; // 机构信息不存在

XC_MGID = 'UARA004' ; // &1 不存在

XC_AINF = XC_SYPN + '用户所属机构' ;

// @@ 抛错误信息并继续

ENDIF ;

// 柜台用户的证件、性别、身份认证方式检查

IF UA3101IS.ISUSTP = 'T' ; // 用户类型为柜台用户

// 证件类型检查

IF UA3101IS.ISIDTP = *BLANK ; // 证件类型

XC_MGID = 'UARA002' ; // &1, &2 不能为空

XC_AINF = XC_SYPN + '维护柜台用户时&&证件类型' ;

// @@ 抛错误信息并继续

ENDIF ;

// 证件代码检查

IF UA3101IS.ISIDNO = *BLANK ; // 证件号码

XC_MGID = 'UARA002' ; // &1, &2 不能为空

XC_AINF = XC_SYPN + '维护柜台用户时&&证件号码' ;

// @@ 抛错误信息并继续

ENDIF ;

// 性别检查
```

```
IF UA3101IS.ISSEX = *BLANK ; //性别

XC_MGID = 'UARA002' ; //&1, &2 不能为空

XC_AINF = XC_SYPN + '维护柜台用户时&&性别' ;

//@@抛错误信息并继续

ENDIF ;

//身份认证方式检查

IF UA3101IS.ISSEAG = *BLANK ; //用户身份认证方式

XC_MGID = 'UARA002' ; //&1, &2 不能为空

XC_AINF = XC_SYPN + '维护柜台用户时&&用户身份认证方式' ;

//@@抛错误信息并继续

ENDIF ;

ENDIF ;

//合法性检查错误，结束程序

IF SCCTWA.I_MSGT = G_TWA_MSGT_ERR ;

EXSR @SR_RETURN ;

ENDIF ;

ENDSR ;
```

3) @SR_MAIN: 主处理

```
BEGSR @SR_MAIN ;

//根据操作功能不同，分别处理

IF UA3101IS.ISFUNC = 'C' ; //参数操作功能码= '增加'

//增加【用户基本信息表】记录

EXSR @SR_WRTUSB ;

IF UA3101IS.ISSEAG = '1' ; //用户身份认证方式= '凭密码'

//初始密码生成

EXSR @SR_GENPWD ;

//增加【用户密码信息表】记录
```

```
        EXSR @SR_WRTUSP ;

    ENDIF ;

ENDIF ;

IF UA3101IS.ISFUNC = 'U' ; //参数操作功能码= '修改'
    //更新【用户基本信息表】记录

    EXSR @SR_UPDUSB ;

    //如果密码信息不存在，还需要增加【密码信息记录】

    IF UA3101IS.ISSEAG = '1' ; //用户身份认证方式= '凭密码'

        //按 '银行号' + '用户名' 键值读取【用户密码信息】表（UAUSPA01）

        K01_USPBKNO = SCCTWA.I_BKNO ; //银行号

        K01_USPUSNO = UA3101IS.ISUSNO ; //用户号

        CHAIN %KDS (DS_UAUSP_K01) UAUSPA01 ;

        IF NOT %FOUND ;

            //初始密码生成

            EXSR @SR_GENPWD ;

            //增加【用户密码信息表】记录

            EXSR @SR_WRTUSP ;

        ENDIF ;

    ENDIF ;

ENDIF ;

ENDSR ;
```

4) @SR_WRTUSB: 增加【用户基本信息表】记录

```
BEGSR @SR_WRTUSB ;

    //按 '银行号' + '用户名' 键值读取【用户基本信息表】记录（UAUSBA01）

    K01_USBBKNO = SCCTWA.I_BKNO ; //银行号

    K01_USBUSNO = UA3101IS.ISUSNO ; //用户号
```

```
CHAIN %KDS (DS_UAUSB_K01) UAUSBA01 ;  
  
//将组件接口参数传到用户基本信息表字段  
  
CLEAR DS_UAUSB_IS ;  
  
EVAL-CORR DS_UAUSB_IS = UA3101IS ;  
  
USBBKNO = SCCTWA.I_BKNO ; //银行号  
  
USBUSST = 'N' ; //用户状态: 正常  
  
USBUIFG = 'F' ; //用户签到标识: 正式签退  
  
USBSGTM = *BLANK ; //签到终端  
  
USBUIDT = *ZEROS ; //用户签到日期  
  
USBUITM = *ZEROS ; //用户签到时间  
  
USBUODT = *ZEROS ; //用户签退日期  
  
USBUOTM = *ZEROS ; //用户签退时间  
  
USBBJFG = *BLANK ; //用户是否完成业务日志勾兑标志  
  
USBCKUS = *BLANK ; //复核用户  
  
USBCRUS = SCCTWA.I_USNO ; //建立用户  
  
USBCRBR = SCCTWA.I_TRBR ; //建立机构  
  
USBCRDT = SCCTWA.I_TRDT ; //建立日期  
  
USBMTUS = SCCTWA.I_USNO ; //维护用户  
  
USBMTBR = SCCTWA.I_TRBR ; //维护机构  
  
USBMTDT = SCCTWA.I_TRDT ; //维护日期  
  
USBTSTM = %TIMESTAMP ; //时间戳  
  
IF %FOUND ; //通过检查后, 如果能找到, 一定是注销的用户信息  
  
    //用户信息存在时, 将组件接口参数更新到【用户基本信息表】  
  
    UPDATE RUAUSBA ;  
  
ELSE ; //找不到用户信息  
  
    //将组件接口参数写入【用户基本信息表】中  
  
    WRITE RUAUSBA ;  
  
ENDIF ;
```


ENDSR ;

5) @SR_UPDUSB: 更新【用户基本信息表】记录

BEGSR @SR_UPDUSB ;

//按‘银行号’+‘用户名’键值读取【用户基本信息表】记录 (UAUSBA01)

K01_USBBKNO = SCCTWA.I_BKNO ; //银行号

K01_USBUSNO = UA3101IS.ISUSNO ; //用户号

CHAIN %KDS (DS_UAUSB_K01) UAUSBA01 ;

//用户信息存在时，将组件接口参数更新到【用户基本信息表】

IF %FOUND ;

EVAL-CORR DS_UAUSB_IS = UA3101IS ;

USBBKNO = SCCTWA.I_BKNO ; //银行号

USBMTUS = SCCTWA.I_USNO ; //维护用户

USBMTBR = SCCTWA.I_TRBR ; //维护机构

USBMTDT = SCCTWA.I_TRDT ; //维护日期

USBTSTM = %TIMESTAMP ; //时间戳

UPDATE RUAUSBA ;

ENDIF ;

ENDSR ;

6) @SR_GENPWD: 初始密码生成

BEGSR @SR_GENPWD ;

//获取证件后 6 位

IF %LEN(%TRIM(UA3101IS.ISIDNO)) >= 6 ; //证件号码长度多于 6

DS_IDNO = %SUBST (%TRIM (UA3101IS.ISIDNO):

(%LEN (%TRIM (UA3101IS.ISIDNO)) - 5): 6) ;

ELSE ; //证件号码长度不足 6 位

DS_IDNO = %TRIM (UA3101IS.ISIDNO) ;

```
ENDIF;

//非零字符处理

P_IDNUM = %CHECK(L_DIGITS_ALL : DS_IDNO) ;

DOU P_IDNUM = 0 ;

DS_IDNO(P_IDNUM) = '0' ;

P_IDNUM = %CHECK(L_DIGITS_ALL : DS_IDNO) ;

ENDDO ;

//调用技术平台<交易密码处理>组件，生成初始密码

CLEAR SECTRANPW ;

//密码组件操作类型

SECTRANPW.I_PCOT = '5' ; //初始密码的增加

//密码组件账号类型

SECTRANPW.I_PCAT = '2' ; //柜员号/用户号

//前端加密用的主账号

SECTRANPW.I_ENCV = UA3101IS.ISUSNO ;

//密码 PIN BLOCK 加密数据

SECTRANPW.I_PW16 = DS_IDNO ; //证件后 6 位

SECTRANPW.I_PECT = *ZERO ;

XPGM_SETTRANPSW(SCCTWA:SECTRANPW) ;

IF %SUBST(SECTRANPW.I_RTCD:3:1) = 'R' ; //交易密码处理组件返回码

    //@@抛错误信息 PGM (SECTRANPW)

ENDIF ;

ENDSR ;
```

7) @SR_WRTUSP: 增加【用户密码信息表】记录

```
BEGSR @SR_WRTUSP ;

//在增加或修改用户信息时，需要将用户密码信息写入【密码信息表】

CLEAR RUAUSP ;
```

```
USPBKNO = UA3101IS.ISBKNO ; //银行号

USPUSNO = UA3101IS.ISUSNO ; //用户名

USPPSWD = SECTRANPW.I_PW16 ; //用户当前密码

USPPWDT = SCCTWA.I_ACDT ; //用户密码设置日期为会计日期

USPPECT = 0 ; //用户密码重试次数

USPPSFL = 'Y' ; //用户密码修改标志

USPCRUS = SCCTWA.I_USNO ; //建立用户

USPCRBR = SCCTWA.I_TRBR ; //建立机构

USPCRDT = SCCTWA.I_TRDT ; //建立日期

USPMTUS = SCCTWA.I_USNO ; //维护用户

USPMTBR = SCCTWA.I_TRBR ; //维护机构

USPMTDT = SCCTWA.I_TRDT ; //维护日期

USPTSTM = %TIMESTAMP ; //时间戳

WRITE RUAUSPA ;

ENDSR ;
```

8) @SR_RETURN: 结束处理

```
BEGSR @SR_RETURN ;

UNLOCK UAUSBA01 ;

UNLOCK UAUSPA01 ;

RETURN ;

ENDSR ;
```

1.3 测试要点

- 1) 用户已存在，做新增，是否报错；
- 2) 用户不存在，做修改，是否报错；
- 3) 已签到用户，最修改，是否报错；

- 4) 柜台用户，不输入证件类型、证件代码、性别、用户身份认证方式，是否报错；
- 5) 做修改时，需要注意没有修改的要素是否也做了修改。