海颐标识

新疆天富能源营销自动化项目

存储过程开发规范

烟台海颐软件股份有限公司

电力产品部

2016年07月

文件变更记录

\*A - 增加 M - 修订 D - 删除

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 版本号 | 日期 | 变更类型（A\*M\*D） | 修改人 | 变更摘要 | 备注 |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

目 录

[1 命名规范 4](#_Toc397325430)

[2 编写规范 4](#_Toc397325431)

[2.1 函数/过程声明规范 4](#_Toc397325432)

[2.2 参数命名规范 5](#_Toc397325433)

[2.3 注释规范 5](#_Toc397325434)

[2.4 异常处理规范 7](#_Toc397325435)

[2.5 书写规范 9](#_Toc397325436)

[2.6 临时表规范 9](#_Toc397325437)

[2.7 事务管理规范 10](#_Toc397325438)

[3 管控规范 10](#_Toc397325439)

[3.1 创建及变更规范 10](#_Toc397325440)

[3.2 现场同步规范 11](#_Toc397325441)

[4 附件：存储过程完整示例 11](#_Toc397325442)

# 命名规范

用能代码约定如下：

供电 D，供气 Q，供热 R。

各模块代号约定如下：

业扩YK，抄核收CHS，计量JL，损耗SH，用检YJ，服务渠道KF，客户关系KHGX，停能TN，市场建设和市场交易SCJY、需求侧和市场开发SCKF，营销稽查YXJC、系统支撑XT，外部集成JC，银电联网 YDLW；

与业务无关的通用函数、过程和包的名称、用能类别为一体化的，不需要加模块代号，直接用p\_或f\_开头。

1. PROCEDURE命名规范：

电、热、气:

模块代号+下划线+P\_+用能代码+下划线+功能汉语拼音缩写，例如供电的开放抄表数据功能为chs\_p\_d\_kfcbsj。

一体化：

模块代号+下划线+P+下划线+功能汉语拼音缩写，例如开放抄表数据功能为chs\_p\_kfcbsj。

1. FUNCITON命名规范：

模块代号+下划线+F\_+用能代码+下划线+功能汉语拼音缩写，例如供电的开放抄表数据功能为chs\_f\_d\_kfcbsj。

1. 统计报表FUNCITON命名规范

格式：rpt\_+模块代号+下划线+f\_+用能代码+下划线+功能汉语拼音缩写

如：供电的抄核收应收报表 rpt\_chs\_f\_d\_dfysbb;

1. PACKGE和PACKGE BODY命名规范：

PKG\_+模块代号+下划线+用能代码+下划线+功能汉语拼音缩写，例如供电的抄核收处理工作单的包为PKG\_CHS\_D\_GZD。

除统计报表外，其它Function和procedure必须按照业务相关性进行分包，放到合适的package中。

# 编写规范

## 函数/过程声明规范

为了在营销系统中能够对执行函数/过程的结果进行友好提示，因此要求被前台功能调用的函数/过程在声明时必须返回字符串类型参数。执行成功时，返回空值（不是空串，应该return null）；执行失败时，返回有意义的错误提示信息。然后营销系统中接收此返回参数，如果返回值不为空，则根据返回值进行信息提示。

例如：

声明函数：

create or replace function chs\_f\_d\_kfcbsj (

…)

return varchar2

is

begin

end;

声明过程

create or replace procedure chs\_f\_d\_kfcbsj(

…,

v\_ret\_str out varchar2 --返回参数

)

is

begin

end;

## 参数命名规范

1. 传入参数命名为in（或out）\_+有意义英文词汇，并在参数后写注释说明参数的含义，例如：

create or replace function chs\_f\_d\_kfcbsj (

in\_value varchar2, --抄表区段号或户号

in\_mon number, --抄核收年月

….

)

1. 函数/过程中用到的变量命名
2. 建议采用v\_有意义的英文词汇，并在变量后写注释说明变量的含义，当变量实际上表示数据库表的某列数据时，为避免数据库结构修改对变量的影响，应统一使用%TYPE属性对变量命名。

例如v\_user\_no kh\_ydkh.yhbh%type; --用户编号

1. 游标的命名：游标统一用前缀 “c\_cur\_” 来命名，例如c\_cur\_user。

## 注释规范

1. 在函数/过程的传入参数声明结束后，需要对本函数/过程的实现做简要说明，如功能、算法，以及其它有利于自己或他人以后查看的。例如：

|  |
| --- |
| create or replace function CHS\_F\_D\_YDLWDBKK (  in\_value varchar2 *--银行入账返回串*  )  return varchar2 is *--成功返回null；异常返回错误提示信息*  */\*===============================================================================*  *功能：银电联网单笔扣款之供电方入账(20)*  *说明：1.首先银行方入账成功，然后执行本程序入账（收费方式：B单笔扣款）*  *2.根据银行方入账总金额，按月次，先抄核收、后违约金次序对入账金额进行分配。*  *3.in\_value格式：RETURN\_CODE{00}*  *4.银行返回对帐唯一标识 WYSB*  *5.目前按 户号+抄核收年月+抄核收次数 进行单笔代扣,不按计量点*  *编写人: 张三*  *编写日期：2014-08-29*  *================================================================================\*/*  v\_msg varchar2(500); *--保存返回提示信息* |

1. 在函数/过程编写过程中，原则上应该每一句SQL都要写注释。

注释编写规范如下：

1. 注释单独成行、放在语句前面；
2. 应对不易理解的分支条件表达式加注释；
3. 对重要的计算应说明其功能；
4. 过长的函数实现，应将其语句按实现的功能分段加以概括性说明；
5. 常量及变量注释时，应注释被保存值的含义，合法取值的范围；
6. 单行注释时必须采用--方式，整段SQL注释时允许使用/\*..\*/方式；

例如：

--插入数据到临时表

Insert into tmp\_d\_hs\_yhda(…) select …. From …

1. 函数/过程后期有变更时，需要对修改的每条SQL都写注释，并且注明修改日期和修改人，建议在SQL开始先写一个总的注释，然后对修改的where条件也要写注释。例如：

1）对于新增的SQL语句

--更新最后一起发行抄核收的总次数、总抄核收及次数抄核收信息。2014/4/4 add by刘某某

Update df\_tmp\_user\_select a

Set a…. = ….

Where ….

2）对于修改原SQL语句的

Insert into df\_money\_files(…)

Select …

From jl\_point\_file b, df\_tmp\_user\_select c

where b.user\_no = c.user\_no

and b.user\_ms\_type <> '6'; --排除参考表 2014/6/2 modify by 刘某某

3）对于注释原来SQL语句的

/\*由于….原因，此处不再处理。 2014/5/5 comment by 刘某某

Update df\_tmp\_user\_select a

Set a…. = ….

Where ….

\*/

## 异常处理规范

1. 对于select into 语句，如果字段未使用聚合函数，必须进行异常捕获。

例如：

Select count(1) into v\_count from kh\_ydkh where …，这种语句不需要捕获异常，类似的聚合函数还有sum()，max()等，这种都不需要捕获异常。

例如下面这个语句就要捕获异常，如果不捕获，传入的户号不存在时就会报错：

begin

select yhmc into v\_user\_name from kh\_ydkh where yhbh = v\_user\_no;

exception

when others then

v\_user\_name := null;

end;

1. 每个函数/过程的开始统一声明两个变量，一个错误代码v\_result integer,一个错误信息v\_msg varchar2(500)。
2. 凡是涉及到表操作(insert,update,select,delete)的sql语句,建议在SQL语句开始对错误代码、错误信息两个变量进行赋值，可以对内容紧密关联的多个sql语句共用一个错误信息，例如：

|  |
| --- |
| *--更新抄核收处理月份*  v\_msg := '更新换表工作单抄核收处理月份出错';  *--1更新计量点变更记录*  v\_result := -10;  update lc\_jldbgjl a  set (a.jfgzdbh, a.dfny, a.bqcbcs)  = (select b.gzdbh, b.dfny, b.bqcbcs from tmp\_hs\_yhda b where b.yhbh = a.yhbh and b.dqbm = a.dqbm)  where exists (select 1 from tmp\_hs\_yhda b where b.yhbh = a.yhbh and b.dqbm = a.dqbm)  and a.jfgzdbh is null;  *--2更新换表信息记录*  v\_result := -11;  update lc\_hbxxjl a  set (a.jfgzdbh, a.dfny, a.bqcbcs)  = (select b.gzdbh, b.dfny, b.bqcbcs from tmp\_hs\_yhda b where b.yhbh = a.yhbh and b.dqbm = a.dqbm)  where exists (select 1 from tmp\_hs\_yhda b where b.yhbh = a.yhbh and b.dqbm = a.dqbm)  and a.jfgzdbh is null; |

1. 在函数/过程最后必须进行全局异常处理，调用通用的p\_pub\_error\_log过程来记录错误日志信息。

create or replace function gd\_chs\_f\_jlgztb

(in\_app\_no varchar2,

in\_area\_code varchar2

)

return varchar2

is

begin

….

…

--提交

commit;

return null;

**--异常处理**

**exception**

**when others then**

**rollback;**

**pkg\_comm\_util.p\_pub\_error\_log(v\_result,v\_msg,'gd\_chs\_f\_jlgztb',in\_app\_no);**

**return v\_msg;**

end;

1. 通用的记录错误日志过程参数说明

Procedure p\_pub\_error\_log

(in\_errcode number, --自定义错误代码

in\_errtext varchar2, --自定义错误内容

in\_object Varchar2, --发生错误对象

in\_param varchar2 --过程传入参数

)

其中第1和2参数传入的就是函数/过程开始要求定义的两个变量v\_result和v\_msg,第3个参数是函数/过程的名字，第4个参数是函数/过程的执行参数，如果有多个参数，可以拼成字符串或者根据自己个人需要决定需要保存哪个参数。

## 书写规范

1. 函数/过程/包/包体中的所有文字全部采用小写。
2. 对于if end if; begin end；loop end loop等语句块时，编写时要注意语句的缩进，保持段落的层次感。

例如：

if f1 = 1 then

f2 := 2;

elsif f1 = 2 then

f2 := 3;

else

f2 := 4;

end if;

for v\_cur in c\_cur loop

v\_user\_no := v\_cur.yhbh;

…

end loop;

1. 不允许直接使用plsql自带的格式化工具对sql进行排版，主要是因为plsql自带的格式化工具对insert语句、decode语句等格式化后语句太长、不容易看懂；
2. 建议一行代码长度不要超过屏幕的可视显示区，便于阅读代码。
3. 书写WHERE子句当语句另起一行时，以保留字或者连接符开始，连接符右对齐。

例如：

where f1 = 1

and f2 = 2;

1. INSERT语句，必须书写字段，字段之间需要用TAB或空格分开；字段可5个或6个一行。
2. 多表连接时，使用表的别名来引用列，不允许有不带表别名的字段；外层库表与内层库表的别名不允许重复。
3. 函数/过程不宜过长，建议不要超过500行，过长时，可以采用多个子存储过程实现。

## 临时表规范

命名规范：TMP\_用能代码\_模块代号\_有意义的汉语拼音缩写，创建语法如下：

|  |
| --- |
| create global temporary table TMP\_D\_CHS\_YHDA  (  yhbh VARCHAR2(20) not null,  cbqdbh VARCHAR2(20),  yhmc VARCHAR2(128),  yddz VARCHAR2(128),  wyjqsr NUMBER(5),  dwbm VARCHAR2(20),  dfny NUMBER(6),  nljdl NUMBER(14,2)  )  on commit preserve rows; |

要求临时表必须使用“on commit preserve rows”类型的临时表,在存储过程中使用临时表前必须delete/truncate临时表数据。

因数据库中变更临时表或引用临时表的存储过程时，如果临时表在使用中，变更操作可能会失败，建议大家少用临时表。

临时表不需要申请，与存储过程统一管理。

## 事务管理规范

存储过程自已进行事务控制。

# 管控规范

## 创建及变更规范

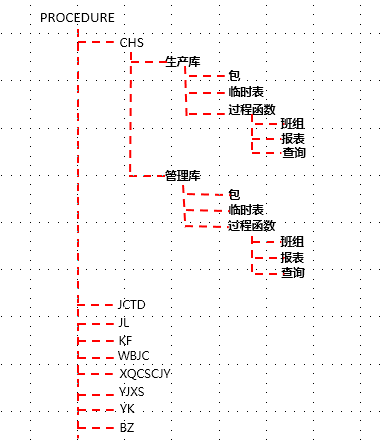
用emms用户来创建存储过程及临时表。

统计报表存储过程必须创建到管理数据库中。

不管是生产库还是管理库，是否需要使用存储过程或临时表，都由各组长负责管控，建议少用存储过程，项目组抽查存储过程质量。

存储过程及临时表对应的创建脚本源文件必须提交到配置库中。创建脚本以配置库中源文件为准，不以数据库中对象为准（也就是说集成部署组和公共资源管控组等可以根据配置库中源文件重建数据库中存储过程及临时表等对象）。

存储过程统一放到\trunk\12Source \集成库\后台\PROCEDURE\下。路径的设置为:



1. 在PROCEDURE文件夹下按照模块建立不同的文件夹
2. 在每个模块文件夹下，新建生产库和集成库两个文件夹。生产库文件夹下存放需要在生产库中创建的存储过程，集成库文件夹下存放需要在集成库中创建的存储过程。
3. 生产库和集成库文件夹下建立包、临时表、存储过程三个文件夹。包文件夹用来存放包脚本，临时表文件夹存放存储过程中用到的临时表的创建脚本。
4. 在存储过程文件夹下建立报表、查询三个文件夹。班组文件夹用来存放给各模块提供给班组用的存储过程脚本。报表文件夹下存放报表用的存储过程脚本。查询文件夹下存放统计查询用的存储过程脚本。
5. 在报表和查询文件夹下不再分月份创建文件夹，所有的存储过程脚本都放在一个目录下，通过svn版本管理。
6. 存储过程文件名规范：函数/过程名\_+函数/过程中文名.sql，临时表文件名规范：表名.sql

## 现场同步规范

公共资源管控组每天将更新的存储过程脚本查找出来，统一提交到现场。

# 附件：存储过程完整示例

|  |
| --- |
| create or replace function chs\_d\_f\_bbcsh\_hyzjb  (in\_busi\_code varchar2, *--供电单位*  in\_mon number, *--报表月份*  in\_operator varchar2, *--操作员*  in\_area\_code varchar2 *--地区编码*  )  return varchar2  is  */\*功能：报表初始化之生成行业用电中间表*  *功能说明：根据行业用电分类汇总电量抄核收数据*  *编写人：刘玉军*  *编写日期：2014/8/30*  *\*/*  v\_result integer;  v\_msg varchar2(500);  v\_zero\_price\_code hs\_djdm.djdm%type := '00000000'; *--零电价代码*  v\_trade\_type\_code hs\_jldxx.hyfldm%type; *--行业代码*  v\_now\_date date := sysdate; *--当前时间*  v\_dept\_code hs\_jldxx.dwbm%type; *--单位编码*  begin  *--删除已有数据*  v\_result := -10;  delete from gk\_chshyzjb a where a.gddwbm = in\_busi\_code and a.bbny = in\_mon and a.dqbm = in\_area\_code;  *--得到单位编码*  v\_dept\_code := pkg\_comm\_util.f\_get\_gddw\_dwbm(in\_busi\_code);  *--插入当月行业用电分类数据*  v\_result := -15;  v\_msg := '插入各行业用电数据到中间表出错';  insert into gk\_chshyzjb  (gddwbm, bbny, hyfldm, dfny, dwbm, ywlbdm,  dl, df, hs, rl, wqwsdl,  czrbs, czsj, cjsj, dqbm)  select in\_busi\_code, in\_mon, nvl(trim(a.hyfldm),'M0000'), in\_mon, v\_dept\_code, '0000',  sum(nvl(a.jfdl,0)), sum(nvl(a.ysdf,0)), 0, 0, round(sum(nvl(a.jfdl,0))/10000,0),  in\_operator, v\_now\_date, v\_now\_date, in\_area\_code  from zw\_ysdfjl a  where a.gddwbm = in\_busi\_code and a.dfny = in\_mon and a.dqbm = in\_area\_code  and a.djdm <> v\_zero\_price\_code *--去掉零电价的*  group by nvl(trim(a.hyfldm),'M0000'); *--为空的参照V2放到公共管理和社会组织*  *--插入其他没有的行业*  v\_result := -20;  insert into gk\_chshyzjb  (gddwbm, bbny, hyfldm, dfny, dwbm, ywlbdm,  dl, df, hs, rl, wqwsdl,  czrbs, czsj, cjsj, dqbm)  select in\_busi\_code, in\_mon, a.dmbm, in\_mon, v\_dept\_code, '0000',  0, 0, 0, 0, 0,  in\_operator, v\_now\_date, v\_now\_date, in\_area\_code  from xt\_dmbm a  where a.dmfl = 'HYFLDM'  and not exists (select 1 from gk\_chshyzjb b  where b.gddwbm = in\_busi\_code and b.bbny = in\_mon and b.dqbm = in\_area\_code and b.hyfldm = a.dmbm);  *--更新户数和容量*  v\_result := -25;  v\_msg := '更新各行业的户数和容量出错';  update gk\_chshyzjb a  set (a.hs, a.rl)  = (select count(distinct b.yhbh), nvl(sum(nvl(b.ydrl,0)),0)  from kh\_jld b, kh\_ydkh c  where b.yhbh = c.yhbh and b.dqbm = c.dqbm  and c.yhztdm <> '2' *--去掉销户*  and b.djdm <> v\_zero\_price\_code *--去掉考核表*  and b.hyfldm = a.hyfldm and b.gddwbm = a.gddwbm and b.dqbm = a.dqbm)  where a.gddwbm = in\_busi\_code and a.bbny = in\_mon and a.dqbm = in\_area\_code;  *--得到电量最大的行业代码*  select min(a.hyfldm) into v\_trade\_type\_code from gk\_chshyzjb a  where a.gddwbm = in\_busi\_code and a.bbny = in\_mon and a.dqbm = in\_area\_code  and a.wqwsdl = (select max(b.wqwsdl) from gk\_chshyzjb b  where b.gddwbm = in\_busi\_code and b.bbny = in\_mon and b.dqbm = in\_area\_code  and b.hyfldm not in ('Y0100','Y0200')); *--居民电量不动*  *--将电量误差放到电量最大的行业上*  v\_result := -35;  v\_msg := '平衡万千瓦时电量误差出错';  update gk\_chshyzjb a  set a.wqwsdl  = a.wqwsdl + (select round(sum(b.dl)/10000,0) - sum(b.wqwsdl) from gk\_chshyzjb b  where b.gddwbm = in\_busi\_code and b.bbny = in\_mon and b.dqbm = in\_area\_code)  where a.gddwbm = in\_busi\_code and a.bbny = in\_mon and a.dqbm = in\_area\_code  and a.hyfldm = v\_trade\_type\_code;  *--删除数据都为零的*  v\_result := -40;  delete from gk\_chshyzjb a  where a.gddwbm = in\_busi\_code and a.bbny = in\_mon and a.dqbm = in\_area\_code  and a.dl = 0 and a.df = 0 and a.hs = 0 and a.rl = 0;  commit;  return null;  exception  when others then  begin  rollback;  pkg\_comm\_util.p\_pub\_error\_log(v\_result, v\_msg, 'gd\_chs\_f\_bbcsh\_hyzjb', in\_busi\_code||','||in\_mon);  return '生成行业用电中间表数据出错！错误内容：'||v\_msg;  end;  end;  / |