# 一、Linux命令行和window PL/SQL命令兼容性和可重复执行处理

1. **, 000-创建表空间V71.00.sql**

文件修改如下:

创建表空间是添加IF NOT EXISTS语句来做到可以重复执行

例如：

CREATE LARGE TABLESPACE "EMP\_TABLESPACE"

IN DATABASE PARTITION GROUP "IBMDEFAULTGROUP"

PAGESIZE 16K

EXTENTSIZE 256

PREFETCHSIZE 192

BUFFERPOOL "MWSMSBP"

OVERHEAD 12.67

TRANSFERRATE 0.18

AUTORESIZE YES INCREASESIZE 15G

DROPPED TABLE RECOVERY OFF;

改成

BEGIN

IF NOT EXISTS (SELECT TBSPACE FROM SYSCAT.TABLESPACES WHERE TBSPACE='EMP\_TABLESPACE') THEN

BEGIN ATOMIC

EXECUTE IMMEDIATE '

CREATE LARGE TABLESPACE "EMP\_TABLESPACE"

IN DATABASE PARTITION GROUP "IBMDEFAULTGROUP"

PAGESIZE 16K

EXTENTSIZE 256

PREFETCHSIZE 192

BUFFERPOOL "MWSMSBP"

OVERHEAD 12.67

TRANSFERRATE 0.18

AUTORESIZE YES INCREASESIZE 15G

DROPPED TABLE RECOVERY OFF

';

END;

END IF;

COMMIT;

END &&

备注：语句的分割符号由;修改成&&

1. **, 001-EMP网关实时库DB2初始化表结构全新脚本V73.00.sql文件**

修改如下:

1.1所有语句的分割符号由;修改成&&。（备注：其中存储过程，触发器，函数，begin end语句块里面语句，还是保留;号）

1.2 所有的CREATE PROCEDURE都替换为CREATE OR REPLACE PROCEDURE

1.3 建表语句是添加IF NOT EXISTS语句来做到可以重复执行，该文件涉及的表有：

AGENTACCOUNT,ATTIME\_TASK,A\_CHANNELMAP,A\_CMDQUE,A\_CMDQUE\_HIS,A\_GWACCOUNT,A\_GWDBCONF,A\_GWLOGS,A\_GWPARAMCONF,A\_GWPARAMTMPL,A\_GWPARAMVALUE,A\_GWSPBIND,A\_PROINFO,A\_PROTOCOLTMPL,BATCH\_MT\_REQ,BATCH\_MT\_REQ\_HIS,GT\_PORT\_USED,KEYWORD\_BLACK,KF\_CORPBASE,KF\_PARAMS,MMS\_PTMSGID\_SEQ,MMS\_SEND\_HISTORY,MMS\_SEND\_TASK,MO\_DATAREPORT,MO\_TASK,MO\_TASK\_HIS,MO\_WAIT\_A,MO\_WAIT\_C,MR\_SPGATE\_WATCH,MT\_DATAREPORT,MT\_DATAREPORT\_SP,MT\_LEVEL0\_QUEUE,MT\_LEVEL1\_QUEUE,MT\_LEVEL2\_QUEUE,MT\_LEVEL3\_QUEUE,MT\_LEVEL4\_QUEUE,MT\_LEVEL5\_QUEUE,MT\_LEVEL6\_QUEUE,MT\_LEVEL7\_QUEUE,MT\_LEVEL8\_QUEUE,MT\_LEVEL9\_QUEUE,MT\_ROUTETABLE,MT\_TASK,MT\_TASK\_C,MT\_TIMER\_QUEUE,MT\_VERIFY\_TASK,MT\_VERIFY\_WAIT,MT\_WAIT\_A,M\_MON\_BUFINFO,M\_MON\_LOGINFO,M\_MON\_MONLOG,M\_MON\_MONMSG,M\_MON\_PTINFO,M\_MON\_SPGATEINFO,M\_MON\_SYSINFO,M\_MON\_SYSINFO,M\_MON\_THREADINFO,M\_MON\_USERINFO,M\_MON\_BUF\_HIS,M\_MON\_LOG\_HIS,M\_MON\_MONLOG\_HIS,M\_MON\_MONMSG\_HIS,M\_MON\_PT\_HIS,M\_MON\_SPGATE\_HIS,M\_MON\_SYS\_HIS,M\_MON\_THREAD\_HIS,M\_MON\_USER\_HIS,PB\_LIST\_BLACK,PB\_SERVICETYPE,PB\_WEBZZCMD\_LOG,PB\_WEBZZCMD\_QUEUE,PROCESSINGSTATUS,RPT\_WAIT\_A,RPT\_WAIT\_B,RPT\_WAIT\_C,USERDATA,USERFEE,VERSION\_CMPP,XT\_GATE\_QUEUE,MMS\_TEMPLATE,MMS\_BMTREQ,MMS\_BMTREQHIS,MMS\_LEVEL\_QUEUE,MMS\_TASK,MMS\_RPT,MMS\_MO,MMS\_DATAREPORT,MMS\_MODATAREPORT,A\_CMD\_ROUTE,A\_CMD\_PORT,A\_SMSFWDCONF,GW\_MT\_TASK\_BAK,GW\_MT\_TASK\_ERR,GW\_MTSDOK,GW\_MTSDOK\_ERR,GW\_RPTRVOK,GW\_RPTRVOK\_ERR,

GW\_RPTSDOK,GW\_RPTSDOK\_ERR,GW\_TRAN\_DEL\_LOG,GW\_UPD\_DEL\_LOG,A\_MOBILEAREA,

A\_IPCOMINFO,A\_SIMINFO,A\_MNP,A\_MNPERRCODE,A\_SPE\_PHONE,A\_AREACODE,A\_GWROUTE,GW\_CLUSPBIND,GW\_CLUSTATUS,GW\_CLUDECISION,GW\_DBMONITOR,A\_SMSMSGIDMAP,A\_SMSPUSHBIND,A\_SMSPUSHCONF,HTTPERRCODE,A\_PROVINCECITY,TRANS\_LOG,GW\_MSGTAIL,GW\_TAILBIND,GW\_TAILCTRL,GW\_RMSRPT\_RVOK,GW\_RMSRPT\_RVOK\_ERR,GW\_RMSRPT\_RVOK\_BAK,

建表语句重复执行的例子：

CREATE TABLE "AGENTACCOUNT"

("LOGINID" CHARACTER(11) NOT NULL DEFAULT '',

"DESCRIPTION" VARCHAR(100) NOT NULL DEFAULT '代理帐号',

"PRIVILEGE" INTEGER NOT NULL DEFAULT 11,

"USERFEEFLAG" INTEGER NOT NULL DEFAULT 1,

"PRIVILEGEINFO" VARCHAR(1000) NOT NULL DEFAULT '具有代理帐号权限',

"ISTRANSIT" SMALLINT NOT NULL DEFAULT 2

)

DATA CAPTURE NONE

IN "TPACCMGR";

ALTER TABLE "AGENTACCOUNT"

ADD CONSTRAINT "PK\_AGENTACCOUNT" PRIMARY KEY

("LOGINID"

);

修改为

BEGIN

IF NOT EXISTS (SELECT TBSPACE FROM SYSCAT.TABLES WHERE TABNAME='AGENTACCOUNT') THEN

BEGIN ATOMIC

EXECUTE IMMEDIATE '

CREATE TABLE "AGENTACCOUNT"

("LOGINID" CHARACTER(11) NOT NULL DEFAULT '''' constraint PK\_AGENTACCOUNT primary key,

"DESCRIPTION" VARCHAR(100) NOT NULL DEFAULT ''代理帐号'',

"PRIVILEGE" INTEGER NOT NULL DEFAULT 11,

"USERFEEFLAG" INTEGER NOT NULL DEFAULT 1,

"PRIVILEGEINFO" VARCHAR(1000) NOT NULL DEFAULT ''具有代理帐号权限'',

"ISTRANSIT" SMALLINT NOT NULL DEFAULT 2

)

DATA CAPTURE NONE

IN "TPACCMGR"

';

END;

END IF;

备注一：单引号'' 修改为双引号''''，不然会报错。

备注二：添加主键语句ALTER TABLE "AGENTACCOUNT" ADD CONSTRAINT "PK\_AGENTACCOUNT" PRIMARY KEY("LOGINID");

修改为constraint PK\_AGENTACCOUNT primary key，集成到建表语句里面。

备注三：修改之后的建表语句就可以做到重复执行。

1.4主键也可以添加IF NOT EXISTS语句来做到可以重复执行。

ALTER TABLE "MT\_DATAREPORT\_SP"

ADD CONSTRAINT "PK\_MTDATAREPORT\_SP" PRIMARY KEY

("IYMD",

"USERID",

"SPGATE",

"SPISUNCM"

);

修改为

IF NOT EXISTS (SELECT CONSTNAME FROM syscat.tabconst WHERE TABNAME='MT\_DATAREPORT\_SP' AND CONSTNAME='PK\_MTDATAREPORT\_SP') THEN

BEGIN ATOMIC

EXECUTE IMMEDIATE 'ALTER TABLE "MT\_DATAREPORT\_SP"

ADD CONSTRAINT "PK\_MTDATAREPORT\_SP" PRIMARY KEY

("IYMD",

"USERID",

"SPGATE",

"SPISUNCM"

)';

END;

END IF;

备注：修改完之后，添加主键语句就可以重复执行

1.5 索引也是添加IF NOT EXISTS语句来做到可以重复执行。

1.6 约束也是添加IF NOT EXISTS语句来做到可以重复执行。

1.7 函数，存储过程，触发器都是通过CREATE 改成CREATE OR REPLACE实现可以重复执行。

1. **, 002-EMP网关实时库DB2初始化数据全新脚本V73.00.SQL**

修改如下：

所有语句的分割符号由;修改成&&

1. **, 003-EMP-WEB实时库DB2初始化全新脚本V73.00.sql**

1，结构修改：

003-EMP-WEB实时库DB2初始化全新脚本V73.00.sql文件里面的结构修改语句和001-EMP网关实时库DB2初始化表结构全新脚本V73.00.sql文件基本一致。

2，数据修改：

由于初始化数据的时候，很多表有主键或者唯一索引的约束，导致对于dml语句不能做到重复执行。操作之前需要加相关判断是否存在。

例如：

insert into LF\_WEIX\_EMOJI (ID, SUNICODE, SUTF8, SUTF16, SSBUNICODE, FILENAME)

values (26, 'U+1F602', '0xF0 0x9F 0x98 0x82', '0xD83D 0xDE02', 'E412', 'smiles\_03\_04.png') ;

修改

IF NOT EXISTS(SELECT \* FROM LF\_WEIX\_EMOJI WHERE ID = 26) THEN

insert into LF\_WEIX\_EMOJI (ID, SUNICODE, SUTF8, SUTF16, SSBUNICODE, FILENAME)

values (26, 'U+1F602', '0xF0 0x9F 0x98 0x82', '0xD83D 0xDE02', 'E412', 'smiles\_03\_04.png') ;

END IF ;

备注一：表LF\_WEIX\_EMOJI中ID为有唯一约束，需要先判断表中是否有该条记录，没有就插入。