**武农商行数据仓库系统**

**ETL部署手册**

**(版本V0.2)**

**二零一三年四月**

**内部资料 注意保密**

**文档修改历史**

| **版本号** | **修改日期** | **编写** | **评审** | **批准** | **修改内容** |
| --- | --- | --- | --- | --- | --- |
| V0.1 | 2013-04-16 | 范佳颍 |  |  |  |
| V0.12 | 2013-04-25 | 王锦裕 |  |  | 增加：2.3.3安装unixODBC  2.3.4配置ODBC访问Greenplum |

**文档确认记录**

| **版本号** | **修改日期** | **编写** | **评审** | **确认签字** | **确认内容** |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |

**文档审核/审批**

**客户签字**

|  |  |
| --- | --- |
| **姓名** | **职务/职称** |
|  |  |
|  |  |
|  |  |
|  |  |

**文档分发**

此文档将分发至如下各人

|  |  |
| --- | --- |
| **姓名** | **职务/职称** |
|  |  |
|  |  |
|  |  |

目录

[**ETL部署手册** 1](#_Toc354691363)

[**(版本V0.1)** 1](#_Toc354691364)

[1 前言 4](#_Toc354691365)

[1.1 编写目的 4](#_Toc354691366)

[1.2 读者对象 4](#_Toc354691367)

[2 系统环境安装和部署 4](#_Toc354691368)

[2.1 网络连接 4](#_Toc354691369)

[2.2 安装LINUX操作系统 4](#_Toc354691370)

[2.2.1 检查perl环境是否正常并确认版本信息 5](#_Toc354691371)

[2.2.2 检查是否能够正常上传文件 5](#_Toc354691372)

[2.2.3 检查数据库服务器网络连接 7](#_Toc354691373)

[2.2.4 检查磁盘空间 7](#_Toc354691374)

[2.2.5 检查C编译环境 8](#_Toc354691375)

[2.3 安装PERL执行环境 8](#_Toc354691376)

[2.3.1 安装DBI模块 8](#_Toc354691377)

[2.3.2 安装DBD模块 10](#_Toc354691378)

[2.3.3 安装unixODBC 11](#_Toc354691379)

[2.3.4 配置ODBC访问Greenplum 12](#_Toc354691380)

[2.4 创建ETL服务器用户群组与用户 16](#_Toc354691381)

[2.5 HOSTS配置 16](#_Toc354691382)

[2.6 ETL Automation 安装 17](#_Toc354691383)

[2.6.1 环境要求 17](#_Toc354691384)

[2.6.2 安装 ETL Automation 服务端程序 17](#_Toc354691385)

[2.6.3 设定 ETL Automation 的环境变量 17](#_Toc354691386)

[2.6.4 ODBC for Teradata的配置 18](#_Toc354691387)

[2.6.5 Automation知识库的创建 20](#_Toc354691388)

[2.6.6 配置Automation数据库连接信息 20](#_Toc354691389)

[2.6.7 运行与停止Automation服务 20](#_Toc354691390)

[2.6.8 查看Automation服务状态 20](#_Toc354691391)

[3 ETL 部署 21](#_Toc354691392)

[4 部署核对方案 21](#_Toc354691393)

# 前言

## 编写目的

编制本文档的目的在于描述武农商行数据仓库系统（以下简称本系统）ETL子系统的部署流程与方法，为本系统ETL开发维护人员提供基础部署维护指导。

## 读者对象

本文档是对ETL上线部署的总体说明，其目标读者是：

* ETL开发人员；
* ETL测试人员；
* ETL维护人员；
* 其他与本系统开发有关的高级人员。

# 系统环境安装和部署

## 网络连接

在安装之前，必须确定所要安装的服务器能够正确的连接到所要使用的 GreenPlum 数据库。如果不确定你的 Linux 服务器是否能够连接到 GreenPlum数据库，请联系网络管理人员。

## 安装LINUX操作系统

本系统服务器使Red Hat LINUX系统。在安装之前，必须确定所要安装服务器的操作系统被正确安装，进行相应检查是否满足部署条件。如果确实条件请联系系统管理人员。

本次安装的操作系统版本如下(RHEL6.3 x64)：

[root@smdw ~]# uname -a

Linux smdw 2.6.32-279.el6.x86\_64 #1 SMP Wed Jun 13 18:24:36 EDT 2012 x86\_64 x86\_64 x86\_64 GNU/Linux

[root@smdw ~]# cat /etc/redhat-release

Red Hat Enterprise Linux Server release 6.3 (Santiago)

[root@smdw ~]#

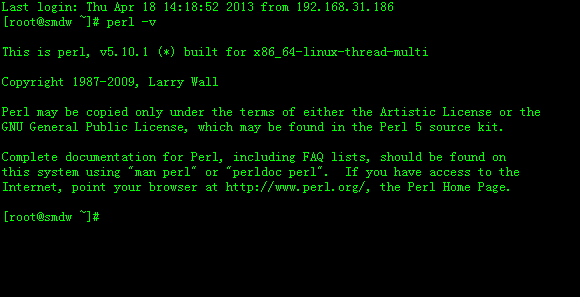
### 检查perl环境是否正常并确认版本信息

在LINUX系统中一般情况下，内定就已安装了 Perl 执行环境。如果你的操作系统中并没有安装 Perl 执行环境，请连络你的系统管理员。

如果你想知道你的系统中是否已安装了 Perl执行环境, 可在Shell 下键入下列指令:

perl -v

如果你的系统中已安装了 Perl执行环境，你会看到在终端机画面上出现 Perl 的程序版本相关信息。



如果你看到是错误讯息则表示你的系统尚未安装 Perl 或版本过低（建议版本在5.8.3 以上），请正确安装 Perl 执行环境。

### 检查是否能够正常上传文件

在LINUX系统中一般情况下，上传文件需要通过telnet的FTP或者SSH，请确认两者是否正常。务必确保其中至少一个可用，如发现两者都不可用的情况，请连络你的系统管理员。

#### 检查FTP

如果你想知道你的系统中是否FTP可用,首先确认telnet正常， 可在Shell 下键入下列指令:

telnet 31.2.1.13

31.2.1.13是本机ip (或者 hostname) 如果配置正确，系统提示输入远程机器的用户名和密码

Login:

Password:

注：默认只允许普通用户。

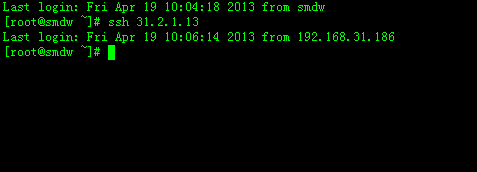
由于telnet登陆时是明文传送密码，所以不安全，建议使用ssh（secure shell）替代！

#### 检查SSH

如果你想知道你的系统中SSH是否可用, 可在Shell 下键入下列指令:

SSH 31.2.1.13

31.2.1.13是本机IP，如果你的SSH可用，你会看到在终端机画面上出现 提示你输入password信息，在输入正确password后提示你最后一次登录信息。



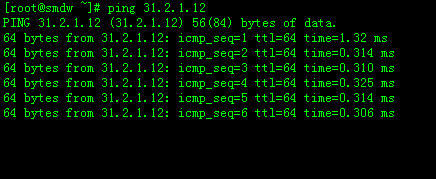
如果你看到是错误信息则表示你的系统SHH尚未可用，请连络你的系统管理员。

### 检查数据库服务器网络连接

如果需要确认当前服务器是否能够正常连接数据库服务器，可在Shell 下键入下列指令:

ping 31.2.1.12

31.2.1.12是数据库服务器IP，如果网络正常连接你会看到在终端机画面上出现连接相应信息。

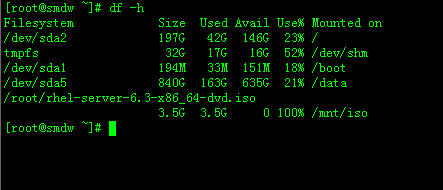


如果你看到是错误讯息则表示你的网络不能连接上数据库服务器或者数据库服务器为启动，请连络你的系统管理员。

### 检查磁盘空间

在程序部署前，请确认当前有足够的空间可以使用，可在Shell 下键入下列指令:

df -h

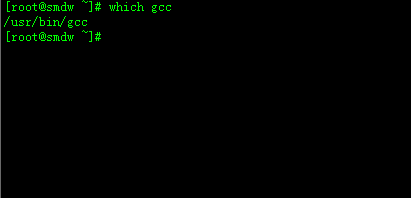


如果空间不足，请连络你的系统管理员。

### 检查C编译环境

当安装部分软件的时候我们需要C编译环境，如果需要确认当前服务器是否安装了gcc，可在Shell 下键入下列指令:

which gcc



如果在终端机画面上显示没有找到gcc，请连络你的系统管理员。

## 安装PERL执行环境

由于 ETL Automation 服务端的程序是由 Perl语言所撰写的，所以必须要先安装 Perl 的执行环境。在确认Perl环境正常的情况下我们还需要单独安装Perl的扩展包来满足我们程序的需要。

扩展包如下：

* + DBI:Perl数据库访问接口；
  + DBD-ODBC:Perl数据库访问ODBC驱动模块；

### 安装DBI模块

Perl 的DBI 接口模块是一个类似 ODBC 的接口程序，Perl 的 DBI 接口提供了一组标准的函数界面让我们可以使用来对于数据库进行运作，而无须担心如果实际上所连接的数据库的不同会使程序的写法有差异。事实上，DBI 只定义了一组标准的函数界面, 不管你实际上所连接的数据库是那一种，对使用 DBI 来说都是一样的。

现在，让我们来看一下如何在Linux 上安装 Perl 的 DBI 接口模块。首先，你必须以系统管理者 (root) 的身份登入进系统中并键入如下的指令：

cd /tmp

mkdir DBI

cd DBI

上面指令会更换目前的目录到 /tmp 下并建立一个新的目录叫做 DBI，接着我们再次更换目前的目录到 /tmp/DBI 下，将 Perl 的 DBI 接口的压缩文件复制到我们刚才建立好的目录，/tmp/DBI 下Perl 的 DBI 接口的压缩文件名称应该是 DBI-1.608.tar.gz，如果你有更新的版本，你可以安装新的版本。

在成功的将**DBI-1.608.tar.gz** 复制到 /tmp/DBI 目录下之后, 键入下列的指令将 **DBI-1.608.tar.gz** 解压缩：

tar -xzvf DBI-1.608.tar.gz

上面的指令会将 DBI-1.608.tar.gz 中所包含的文件解开放在目前的目录, 也就是 /tmp/DBI 下。其中有一个文件叫做 “Readme”，在这个文件里包含了详细的安装步骤，如果你想了解进一步的信息，请参阅这个文件。

接下来请键入下列指令：

perl Makefile.PL

当上列的指令执行完毕后，再键入下列指令：

make

当上列的指令执行完毕后，再键入下列指令：

make test

当上列的指令执行完毕后，再键入下列指令：

make install

注意在执行安装步骤时是否有错误产生，如果没有错误的话，那么你你已完成了 Perl 的 DBI 接口模块的安装了。

如果需要确认当前服务器DBI是否安装成功，可在Shell 下键入下列指令:

perl -e ‘use DBI’



如果在终端机画面上显示没有找到报错，证明DBI已经正常安装，如果出现错误信息请从新检查安装。

### 安装unixODBC

准备好安装介质unixODBC-2.3.0.tar.gz或者更高版本，然后顺序执行下面命令：

tar -zxvf unixODBC-2.3.0.tar.gz

解压完成后进入unixODBC-2.3.0目录

cd unixODBC-2.2.1

./configure --prefix=/usr/local/unixODBC-2.2.1 --includedir=/usr/include --libdir=/usr/lib -bindir=/usr/bin --sysconfdir=/etc

或者

./configure

make

make install

安装成功后，unixODBC所需的头文件都被安装到了/usr/inlucde下，编译好的库文件安装到了/usr/lib下，与unixODBC相关的可执行文件安装到了/usr/bin下，配置文件放到了/etc下。

注意：由于本操作系统是64位,共享库文件可能安装在/usr/lib64下面；程序文件可能位于/usr/local/bin下面。

方法二：

下载rpm包进行安装，我们这里以Red Hat 7.3为例：

unixODBC-2.2.0-5 RPM for i386（安装包及源码包）

（ ftp://speakeasy.rpmfind.net/linux/redhat/7.3/en/os/i386/RedHat/RPMS/unixODBC-2.2.0-5.i386.rpm、 ftp://ftp.rpmfind.net/linux/redhat/7.3/en/os/i386/SRPMS/unixODBC-2.2.0-5.src.rpm）

unixODBC-devel-2.2.0-5 RPM for i386

（ ftp://speakeasy.rpmfind.net/linux/redhat/7.3/en/os/i386/RedHat/RPMS/unixODBC-devel-2.2.0-5.i386.rpm）

直接将unixODBC-2.2.0-5.i386.rpm和unixODBC-devel-2.2.0-5.i386.rpm装入系统就可以了，命令如下：

rpm -ivh unixODBC-2.2.0-5.i386.rpm

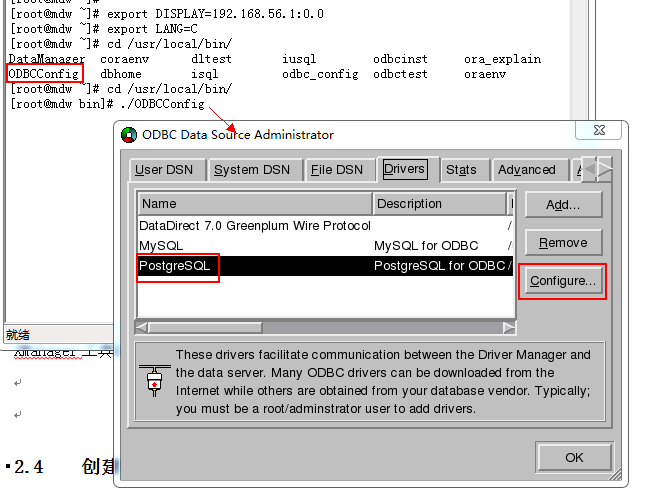
rpm -ivh unixODBC-devel-2.2.0-5.i386.rpm

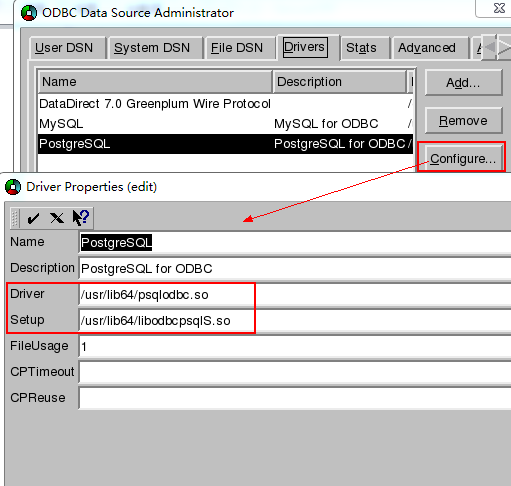
安装好以后，所需的各个部分与上面所列的位置相同。

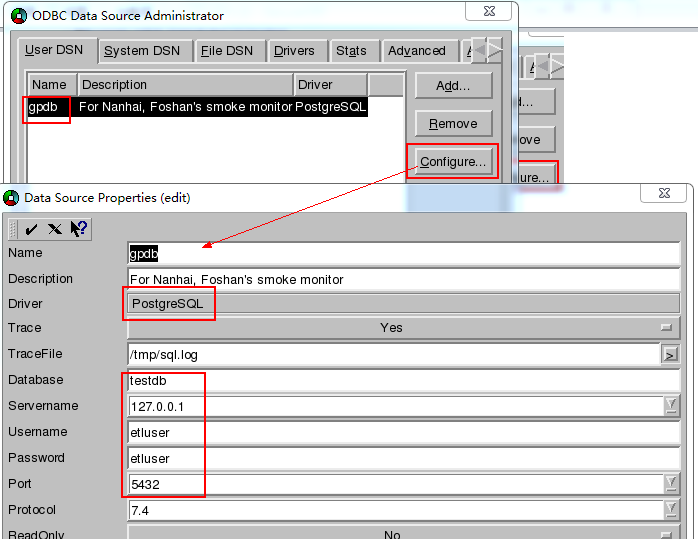
### 配置ODBC访问Greenplum

要想通过ODBC访问Greenplum，前提是本机已经拥有Greenplum的ODBC驱动，就是安装过greenplum-db-4.2.0.0-build-5-RHEL5-x86\_64.zip软件，或是安装了greenplum-connectivity-4.1.0.0-build-5-RHEL5-x86\_64.zip客户端软件，或者是安装过Postgres数据库(因为GP兼容Postgres的驱动)，这部分安装参考有关的产品安装部署文档。

在本次安装中，具体进行如下配置(通过图形化来配置，需要开启本地XManager工具的本地图形接收服务)：







配置好之后，可以查看到相关配置信息：

[root@mdw bin]# cat /etc/odbcinst.ini

# Example driver definitions

# Driver from the postgresql-odbc package

# Setup from the unixODBC package

[PostgreSQL]

Description = ODBC for PostgreSQL

Driver64 = /usr/lib64/psqlodbc.so

Setup64 = /usr/lib64/libodbcpsqlS.so

FileUsage = 1

[root@mdw bin]# cat /etc/odbc.ini

[gpdb]

Description = For Nanhai, Foshan's smoke monitor

Driver = PostgreSQL

Trace = Yes

TraceFile = /tmp/sql.log

Database = testdb

Servername = 127.0.0.1

UserName = etluser

Password = etluser

Port = 5432

Protocol = 7.4

ReadOnly = No

RowVersioning = No

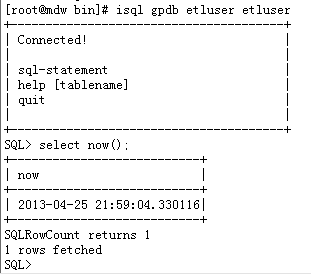
ShowSystemTables = No

ShowOidColumn = No

FakeOidIndex = No

ConnSettings =

测试一下：



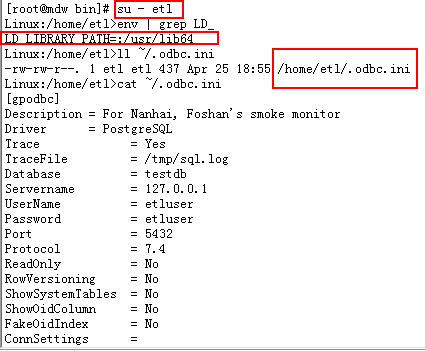
在文档中，后面会专门建立etl用户，只需要把odbc.ini文件复制到etl用户目录下变成 .odbc.ini 就可以，并且设置etl的环境变量:

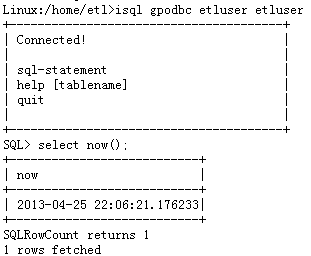
su – etl

export LD\_LIBRARY\_PATH=$LD\_LIBRARY\_PATH:/usr/lib64

cp /etc/odbc.ini ~/.odbc.ini

然后就可以得到下面同样的测试结果：





### 安装DBD模块

光安装了 DBI 并不能成功的使用数据库，你还必须安装实际的 DBD 接口。现在，让我们来看一下如何在Linux 上安装 Perl 的 DBD 接口模块。首先，你必须以系统管理者 (root) 的身份登入进系统中检查如下环境变量是否配置：

ODBCHOME=/usr ##ODBC路径

LD\_LIBRARY\_PATH=/usr/lib64 ##ODBC驱动路径

然后键入如下的指令：

cd /tmp

mkdir DBD

cd DBD

上面指令会更换目前的目录到 /tmp 下并建立一个新的目录叫做 DBD，接着我们再次更换目前的目录到 /tmp/DBD 下，将 Perl 的 DBD 接口的压缩文件复制到我们刚才建立好的目录，/tmp/DBD 下Perl 的 DBD 接口的压缩文件名称应该是 DBD-ODBC-1.23\_2.tar，如果你有更新的版本，你可以安装新的版本。

在成功的将DBD-ODBC-1.23\_2.tar 复制到 /tmp/DBD 目录下之后, 键入下列的指令将 DBD-ODBC-1.23\_2.tar 解压缩：

tar -**-xvf** DBD-ODBC-1.23\_2.tar

上面的指令会将 DBD-ODBC-1.23\_2.tar 中所包含的文件解开放在目前的目录, 也就是 /tmp/DBD 下。其中有一个文件叫做 “Readme”，在这个文件里包含了详细的安装步骤，如果你想了解进一步的信息，请参阅这个文件。

接下来请键入下列指令：

perl Makefile.PL

当上列的指令执行完毕后，再键入下列指令：

make

当上列的指令执行完毕后，再键入下列指令：

make test

当上列的指令执行完毕后，再键入下列指令：

make install

注意在执行安装步骤时是否有错误产生，如果没有错误的话，那么你你已完成了 Perl 的 DBI 接口模块的安装了。

如果需要确认当前服务器DBI是否安装成功，可在Shell 下键入下列指令:

perl -e ‘use DBD::ODBC’



如果在终端机画面上显示没有找到报错，证明DBI已经正常安装，如果出现错误信息请从新检查安装。

## 创建ETL服务器用户群组与用户

接下来我们必须为 ETL Automation 建立专属的群组及使用者。我们先建立一名为 etl 的群组，可在Shell 下键入下列指令:

groupadd -g etl

在成功的加入群组之后，接着就是建立一名为 etl 的使用者，可在Shell 下键入下列指令:

useradd etl

设定用户密码，可在Shell 下键入下列指令:

passwd etl

然后我们把此 etl 的使用者归在 etl 的群组之下，可在Shell 下键入下列指令:

gpasswd -a etl etl

至于 etl 使用者的主目录 (Home Directory) 则设在 /ETL 下，所有 ETL Automation 的工作目录皆位于此目录之下。在此我们假设将 /ETL 设成是 ETL Automation 的主目录，我们在后面的范例及说明也将以 /ETL 当成是在 LINUX 中的主目录。

|  |
| --- |
| 建议：  这里准备做为 etl 使用者主目录的 /ETL 目录最好是单独属于一个文件系统 (File System), 并确定其空间足够存放一段期间的来源数据 (Source Data). |

## HOSTS配置

bteq范围数据库需要通过配置HOSTS文件，直接在HOSTS文件中添加数据库地址信息。

vi /etc/hosts

添加如下信息

31.2.1.12 dbccop1

## ETL Automation 安装

### 环境要求

* Perl, DBI, DBD-ODBC
* unix configure file(/etc/hosts)

### 安装 ETL Automation 服务端程序

现在请用 etl 使用者登录进系统中，接下来我们将进行进行 ETL Automation 服务端程序的安装。

我们先把 Auto261.tar.gz 从安装介质上传送到 在 etl 主目录之。然后执行：

tar -xzvf Auto261.tar.gz

cd install

perl set\_wrkdir.pl

cp $HOME/install/\* $HOME/bin

在完成安装动作之后就可以将$HOME/install 整个目录移除了。

rm -rf install/

### 设定 ETL Automation 的环境变量

在启动 ETL Automation 的服务程序时会从环境变量中读取相关的设定值，有一些变量是必须要设定的，而其它的变量如果没有设定则ETL Automation 会自动指定给这些变量一个内定值。

在 Linux 上要设定环境变量的方式是直接编辑在使用者 etl 的主目录 $HOME 下的 .profile 文件，请用 etl 这个使用者身份登入进系统中并用 vi 或其它的编辑程序来修改 .profile 文件中的内容。

在此我们先列出几个必须要设定的重要变量，至于其它环境变量的意义请参阅后面章节的说明。

export AUTO\_AGENT\_PORT=6346

export AUTO\_DB=PETL

export AUTO\_DSN=Greenplum

export AUTO\_HOME=/home/etl

export AUTO\_JOB\_COUNT=10

export AUTO\_PRIMARY\_SERVER=1

export AUTO\_PRIMARY\_SERVER\_IP=31.2.1.13

export AUTO\_SERVER=ETL1

export AUTO\_SERVER\_IP=31.2.1.13

export AUTO\_SLEEP=20

在上面所列出来的是你必须要加入到 etl 这个使用者的 .profile 中，不过实际上在你的计算机上的设定值与这里的范例会有些出入。

AUTO\_HOME：这个环境变量的内容是 ETL Automation 所安装的主目录，如“/home/etl”，如果你安装的路径名称与这里的范例不一样，那么请你设定成正确的目录路径；

AUTO\_DSN：这个环境变量的内容是指你所要连接的 GreenPlum 数据库系统的Data-Source-Name名称，如“Greenplum”；

AUTO\_DB：这个环境变量的内容是指 ETL Automation 所使用的数据库名称，如“PETL”；

AUTO\_SERVER：这个环境变量的内容是指这部执行 ETL Automation 服务程序计算机的名称，当你的环境中有多部机器同时执行ETL Automation 服务程序时，你必须将每一部机器设定一个唯一的名称，在此我们假设是叫做 “ETL1”；

AUTO\_SERVER\_IP：这个环境变量的内容是指这部执行 ETL Automation 服务程序计算机的网络地址，在此我们假设成“31.2.1.13”，你必须设定成你的计算机的实际网络地址；

AUTO\_AGENT\_PORT：这个环境变量的内容是指 etlagent.pl 将会使用的网络端口号 (Port Number)，内定值为 6346。

### ODBC for Teradata的配置

#### 配置.odbc.ini

使用 etl 用户登录，把/opt/teradata/client/13.0/ODBC\_64 下的odbc.ini文件复制到 $HOME/.odbc.ini。

vi $HOME/.odbc.ini

把 default 换成etldb。 (:1,$s/default/etldb/g)，并把文件中

DBCName 填写成实际的Teradata 节点IP-Address。

如下：

DBCName=192.168.1.222

#### 配置odbcinst.ini

使用root用户复制:/opt/teradata/client/ODBC\_64/cat odbcinst.ini中的内容然后粘贴到/etc/unixODBC /odbcinst.ini中

#### 验证ODBC配置

创建测试程序odbc\_gp.pl，然后运行查看ODBC是否正常，测试程序模板如下：

|  |
| --- |
| #!/usr/bin/perl  use strict;  use DBI;  use DBD::ODBC;  #建立数据库连接  print "visit----\n";  my $dbh = DBI->connect("dbi:ODBC:Greenplum ","gpadmin"," gpadmin ",{AutoCommit=>1,PrintError=>0,RaiseError=>0})|| die "Error : $DBI::errstr\n";  #建立会话  my $sth = $dbh->prepare("select count(\*) from columns") || die "Error : $DBI::errstr\n";  #执行SQL语句  $sth->execute() || die "Error : $DBI::errstr\n";  my @row;  #$sth->fetchrow() : 返回结果  while(@row = $sth->fetchrow())  {  print "@row\n";  }  $sth->finish;  #关闭数据库连接  $dbh->disconnect(); |

### Automation知识库的创建

使用Greenplum工具运行EtlRepository.sql，在运行前表名前边指定Automation知识库的库名。

### 配置Automation数据库连接信息

在/home/etl/bin目录下运行EncodeLogon.pl，根据提示生成数据库连接信息。

etl@ETL:~/bin> perl EncodeLogon.pl

|  |
| --- |
| Please input the logon file name:ETL\_LOGON  Please input the user name:cmoms  Please input the password:cmoms  Output the logon file...  Logon file was generated. |

### 运行与停止Automation服务

启动服务：在/home/etl/bin目录下运行start.sh

etl@ETL:~/bin> sh start.sh

停止服务：在/home/etl/bin目录下运行stop.sh

etl@ETL:~/bin> sh start.sh

### 查看Automation服务状态

通过如下语句查看当前Automation服务状态：

etl@ETL:~/bin> ps -ef | grep etl



# ETL 部署

# 部署核对方案

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 核对项 | 操作步骤 | 核对结果 |
| 1 | odbc配置 | 查看odbc.ini文件  cd /home/etl/  more odbc.ini |  |
| 2 | profile文件是否生效 | 执行一下命令查看环境变量  echo $AUTO\_HOME  echo $AUTO\_DB | 能够启动，并且变量替换没问题 |
| 3 | ETL用户是否更新成功 | 查看ETL\_LOGON文件  cd /home/etl/bin  more ETL\_LOGON |  |
| 4 | automation中server信息是否更新成功 | 在queryman中执行  select ETL\_Server  from dwpetl.etl\_job  where ETL\_System =‘EDW’  ; |  |
| 5 | automation是否可以正常调用 | 暂时更新profile文件中AUTO\_EDWPDATA变量，然后使用ETL\_admin调用任一个任务（去掉依赖关系），执行命令如下：  vi .profile  export AUTO\_EDWPDATA=--根据具体情况填写 |  |
|  |  |  |  |