# Centos7.4 Oracle 11g GoldenGate 12 安装文档

## 下载 <http://www.oracle.com/technetwork/cn/middleware/goldengate/downloads/index.html>

## 静默安装

1. 解压
2. 编辑响应文件：

vi ./response/oggcore.rsp

|  |
| --- |
| INSTALL\_OPTION=ORA11g  SOFTWARE\_LOCATION=/ogg的安装目录  DATABASE\_LOCATION=/u01/app/oracle/product/11.2.0/dbhome\_1  INVENTORY\_LOCATION=/u01/app/oraInventory  UNIX\_GROUP\_NAME=oinstall |

1. 启动OUI程序指定响应文件进行安装:

|  |
| --- |
| ./runInstaller -silent -responseFile /全路径/response/oggcore.rsp |

1. 配置环境变量

vi ~/.bash\_profile

|  |
| --- |
| export GG\_HOME=xxx  export PATH=$GG\_HOME |

## 部署OGG并测试

一、准备OGG的运行用户

在这里，我直接使用oracle用户作为OGG的运行用户。

二、确认OGG的安装目录

在源端，即Linux环境下，我选择/u01/app/goldengate作为OGG的安装目录，注意：/u01/app/goldengate目录的属主必须为OGG的运行用户，在这里，该目录的属主必须为oracle。

在目标端，即Win环境下，我选择G:\app\goldengate作为OGG的安装目录

三、打开数据库的归档模式

|  |
| --- |
| SQL> archive log list  Database log mode No Archive Mode  Automatic archival Disabled  Archive destination USE\_DB\_RECOVERY\_FILE\_DEST  Oldest online log sequence 5  Current log sequence 7  SQL> shutdown immediate  SQL> startup mount  SQL> alter database archivelog;  SQL> alter database open; |

**四、打开源端数据库最小附加日志**

因为这里搭建的只是单向复制，OGG捕捉的是源端数据库的更改，所以只需对源端数据库开启最小附加日志

|  |
| --- |
| SQL> select supplemental\_log\_data\_min from v$database;  SUPPLEME  --------  NO  SQL> alter database add supplemental log data;  Database altered.  SQL> alter system switch logfile; --切换日志使附加日志生效  System altered. |

**五、在数据库中建立OGG用户并赋予响应的权限**

为了简化授权，直接将dba角色赋予给gg用户。

|  |
| --- |
| SQL> create user gg identified by gg;  SQL> grant dba to gg; |

**六、**在ggsci环境下创建GoldenGate子目录

|  |
| --- |
| [oracle@node1 goldengate]$ ggsci  GGSCI (node1.being.com) 1> create subdirs  Creating subdirectories under current directory /u01/app/goldengate  Parameter files /u01/app/goldengate/dirprm: already exists  Report files /u01/app/goldengate/dirrpt: created  Checkpoint files /u01/app/goldengate/dirchk: created  Process status files /u01/app/goldengate/dirpcs: created  SQL script files /u01/app/goldengate/dirsql: created  Database definitions files /u01/app/goldengate/dirdef: created  Extract data files /u01/app/goldengate/dirdat: created  Temporary files /u01/app/goldengate/dirtmp: created  Stdout files /u01/app/goldengate/dirout: created |

**七、配置Manager进程**

在ggsci环境下配置Manager进程

源端配置如下：

|  |
| --- |
| [oracle@node1 goldengate]$ ggsci  GGSCI (node1.being.com) 1> edit param mgr  GGSCI (node1.being.com) 2> start mgr  Manager started.  GGSCI (node1.being.com) 3> info all  Program Status Group Lag at Chkpt Time Since Chkpt  MANAGER RUNNING --可见MGR进程已启动 |

**八、在源端上添加表级附加日志**

为了验证搭建的效果，在这里我们新建一个test用户，并新建一张test表。

|  |
| --- |
| SQL> create user test identified by test;  User created.  SQL> grant connect,resource to test;  Grant succeeded.  SQL> conn test/test  Connected.  SQL> create table test (id number);  Table created. |

在ggsci命令行中登录数据库，为所有需要复制的表添加trandata

|  |
| --- |
| GGSCI (node1.being.com) 1> dblogin userid gg,password gg  Successfully logged into database.  GGSCI (node1.being.com) 2> add trandata test.\*  2015-08-24 18:14:59 WARNING OGG-00869 No unique key is defined for table 'TEST'. All viable columns will be used to represent the key, but may not guarantee uniqueness. KEYCOLS may be used to define the key.  Logging of supplemental redo data enabled for table TEST.TEST. |

**九、在源端上添加Extract进程**

GGSCI (node1.being.com) **5**> add ext exta,tranlog,begin now

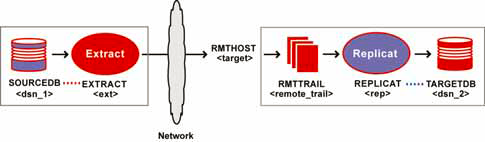
其中，exta为进程名，一般为ext开头表示是extract进程，后面可以加1-2位字符标识；

         tranlog表示要抽取数据库日志；

         begin now表示从当前时间点开始抽取

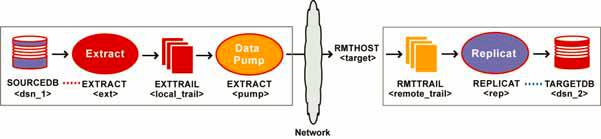
注意：添加完Extract进程后，我们有两种方式来传输数据。

第一种，exta进程直接将捕捉到的有效数据写入到远程队列，网络拓扑如下：



不难看出，该方式对网络要求较高，在实际生产环境中很少使用该种方式。

第二种，首先将数据抽取到本地，然后再由Pump进程传输到远程队列。网络拓扑如下：



在这里，我们采用第二种方式

https://www.cnblogs.com/ivictor/p/4747887.html