

# Oracle Golden Gate (Para replicar ambientes productivos)

Documento generado por



Para el sitio



## Índice

1.	Introducción.....	2
2.	Descarga de utilitario Golden Gate .....	3
3.	Instalación de Golden Gate.....	6
4.	Preparación de la base de datos .....	7
5.	Proceso de replicación .....	13
5.1.	CONFIGURAR EL MANAGER PROCESS .....	13
5.2.	CREACIÓN DEL GRUPO PARA EXTRAER DATOS.....	15
5.3.	CREACIÓN DEL GRUPO PARA REPLICAR DATOS .....	17
5.4.	COMIENZO DE LA EXTRACCIÓN Y REPLICACIÓN .....	18
6.	Sincronización de DDL .....	20

## 1. Introducción

En este paper se trata de dar un muy breve barniz para saber que es Oracle Golden Gate , su instalación y su forma básica de uso para replicación de 2 ambientes Oracle, aunque su espectro es muchísimo mayor, pero en este documento no se muestra su real alcance ni tampoco las arquitecturas que se pueden generar.

¿Qué és Oracle Golden Gate? Pues es una herramienta de replicación de datos en tiempo real, que puede unir muchos orígenes de datos con motores distintos y replicarlos en otros ambientes también distintos es un N:N (Ej : Oracle9i en Linux y replicando datos hacia un Sql\*Server en Windows 64bits)



Sin más comentarios, a entrar en materia


## 2. Descarga de utilitario Golden Gate

Para descarga Oracle Golden Gate , debemos ir al sitio

<http://edelivery.oracle.com>

En el debemos seleccionar Oracle Fusion Middleware y la plataforma que utilizaremos, para nuestro caso Linux x86.

Seleccionar un Paquete de Productos   

Plataforma  

Cuando presionamos Ir , veremos que nos aparecen un sinnúmero de archivos para descargar, buscamos los siguientes

	<a href="#">Oracle GoldenGate on Oracle Media Pack for Linux x86</a>	n/a	B56872-03	JUL-19-2010	6 / 205M
---	--	-----	-----------	-------------	----------

Una vez ingresado a ese pack, seleccionamos el que corresponda a nuestra versión de base de datos y sistema operativo

**Oracle GoldenGate on Oracle Media Pack v3 for Linux x86**

[Léame](#)

[Consultar Resumen](#)

Seleccionar	Nombre	Número de Parte	Tamaño (Bytes)
<a href="#">Descargar</a>	Oracle GoldenGate V10.4.0.x for Oracle 10g on RedHat 3.0	V18153-01	48M
<a href="#">Descargar</a>	Oracle GoldenGate V10.4.0.x for Oracle 10g on RedHat 4.0	V18156-01	48M
<a href="#">Descargar</a>	Oracle GoldenGate V10.4.0.x for Oracle 11g on RedHat 4.0	V18427-01	48M
<a href="#">Descargar</a>	Oracle GoldenGate V10.4.0.x for Oracle 10g on RedHat 5.0	V18428-01	26M
<a href="#">Descargar</a>	Oracle GoldenGate V10.4.0.x for Oracle 11g on RedHat 5.0	V18429-01	26M
<a href="#">Descargar</a>	Oracle GoldenGate Documentation on Oracle	V20521-01	7.0M
<b>Total: 6</b>			

Realizamos la descarga y procedemos a copiar a nuestro Sistema Operativo

Lo ideal es generar una carpeta de trabajo, imaginemos gg

```
[oracle@oracle10ga oracle]$ mkdir gg
[oracle@oracle10ga oracle]$ cd gg
[oracle@oracle10ga gg]$
[oracle@oracle10ga gg]$
[oracle@oracle10ga gg]$ pwd
/home/oracle/gg
[oracle@oracle10ga gg]$
```

Esto lo repetimos en ambas máquinas y copiamos el software en ambas máquinas

Aplicamos el comando unzip

```
[oracle@oracle10ga gg]$ unzip V18153-01.zip
Archive:  V18153-01.zip
  inflating: ggs_redhatAS30_x86_ora10g_32bit_v10.4.0.19_002.tar
```

Y posterior a eso, descomprimos el tar, mediante el siguiente comando

---

```
[oracle@oracle10ga gg]$ tar xvf ggs_redhatAS30_x86_oracle10g_32bit_v10.4.0.19_002.tar
```

Verificamos las variables Oracle disponibles en nuestro sistema

```
[oracle@oracle10gb gg]$ env | grep ORACLE
ORACLE_SID=orcl
ORACLE_BASE=/u01/app/oracle
ORACLE_HOME=/u01/app/oracle/product/10.2.0/db_1
```

Es obligatorio setear ORACLE\_SID , ORACLE\_HOME y LD\_LIBRARY\_PATH, además de colocarlas en el PATH

```
[oracle@oracle10gb gg]$ export LD_LIBRARY_PATH=$ORACLE_HOME/lib:/home/oracle/gg
```

Para ello se recomienda el .bash\_profile o .profile (de acuerdo al interprete de comandos) , posea mas o menos las siguientes definiciones

```
ORACLE_BASE=/u01/app/oracle
ORACLE_HOME=$ORACLE_BASE/product/10.2.0/db_1
ORACLE_SID=orcl

GOLDEN_GATE_HOME=/home/oracle/gg
PATH=/bin:/usr/bin:/usr/local/bin:/usr/X11R6/bin

LD_LIBRARY_PATH=/usr/lib:/usr/X11R6/lib:$ORACLE_HOME/lib:/home/oracle/gg

PATH=$ORACLE_HOME/bin:$PATH

PATH=$GOLDEN_GATE_HOME:$PATH

export PATH LD_LIBRARY_PATH
export ORACLE_BASE ORACLE_HOME ORACLE_SID
```

---

### 3. Instalación de Golden Gate

Una vez descargado el utilitario, copiado en disco y seteado las variables de medio ambiente, procedemos a utilizarlo .

Para ello ejecutamos el comando ggsci

```
[oracle@oracle10gb gg]$ ./ggsci

Oracle GoldenGate Command Interpreter for Oracle
Version 10.4.0.19 Build 002
Linux, x86, 32bit (optimized), Oracle 10 on Sep 23 2009 13:54:17

Copyright (C) 1995, 2009, Oracle and/or its affiliates. All rights reserved.
```

```
GGSCI (oracle10gb.inmotion.cl) 1>
```

Ya en el prompt, Oracle Golden Gate necesita estructuras base para poder trabajar, estas se generan con el comando **CREATE SUBDIRS**

La salida sería mas o menos lo siguiente

```
GGSCI (oracle10gb.inmotion.cl) 1> create subdirs

Creating subdirectories under current directory /home/oracle/gg

Parameter files           /home/oracle/gg/dirprm: created
Report files             /home/oracle/gg/dirrpt: created
Checkpoint files        /home/oracle/gg/dirchk: created
Process status files    /home/oracle/gg/dirpcs: created
SQL script files        /home/oracle/gg/dirsq: created
Database definitions files /home/oracle/gg/dirdef: created
Extract data files      /home/oracle/gg/dirdat: created
Temporary files         /home/oracle/gg/dirtmp: created
Veridata files          /home/oracle/gg/dirver: created
Veridata Lock files     /home/oracle/gg/dirver/lock: created
Veridata Out-Of-Sync files /home/oracle/gg/dirver/oos: created
Veridata Out-Of-Sync XML files /home/oracle/gg/dirver/oosxml: created
Veridata Parameter files /home/oracle/gg/dirver/params: created
Veridata Report files   /home/oracle/gg/dirver/report: created
Veridata Status files   /home/oracle/gg/dirver/status: created
Veridata Trace files    /home/oracle/gg/dirver/trace: created
Stdout files            /home/oracle/gg/dirout: created
```

Esto se debe hacer para ambos nodos, los cuales participarán de la replicación

## 4. Preparación de la base de datos

Una vez realizados los pasos anteriores, se deben verificar ciertos puntos en nuestra base de datos

- ❖ Debe estar activado el modo archiveolog
- ❖ Debe ser desactivada la papelera  

```
SQL> alter system set recyclebin=off scope=both;

System altered.

SQL>
```

Si la papelera no se desactiva arrojará errores al momento de ejecutar los scripts posteriores

```
Working, please wait ...
BEGIN
*
ERROR at line 1:
ORA-20783:
GoldenGate DDL Replication setup:
*** RECYCLEBIN must be turned off.
*** For 10gr2 and up, set RECYCLEBIN in parameter file to OFF. For 10gr1, set
_RECYCLEBIN in parameter file to FALSE. Then restart database and installation.
ORA-06512: at line 19
```

- ❖ Se debe activar el Supplemental logging  

```
SQL> alter database add supplemental log data;

Database altered.

SQL>
```

- ❖ Creamos un esquema para soportar la replicación

```
SQL> create user golden identified by golden
2 default tablespace users
3 temporary tablespace temp;

User created.

SQL> grant connect,resource,unlimited tablespace, select any dictionary to golden;
Grant succeeded.

SQL> grant select any table,create any table, drop any table , insert any table,
update any table, delete any table, create any view , flashback any table to golden;
Grant succeeded.

SQL> grant execute on dbms_flashback to golden;
Grant succeeded.

SQL> grant execute on utl_file to golden;
Grant succeeded.
```

<b>OBS 1</b>	Este esquema se debe crear tanto en el ambiente de origen , como en el destino
--------------	--

<b>OBS 2</b>	El ideal es que este esquema posea Grant de DBA dado que será el encargado de generar todos los objetos en el esquema destino, lo otro es asignarle los privilegios que tenga el esquema de origen.
--------------	---

- ❖ Se ejecutan los scripts para soportar la replicación , sólo en el origen (marker\_setup.sql)

```
SQL> start marker_setup.sql

Marker setup script

You will be prompted for the name of a schema for the GoldenGate database objects.
NOTE: The schema must be created prior to running this script.
NOTE: Stop all DDL replication before starting this installation.

Enter GoldenGate schema name:golden

Marker setup table script complete, running verification script...
Please enter the name of a schema for the GoldenGate database objects:
Setting schema name to GOLDEN

MARKER TABLE
-----
OK

MARKER SEQUENCE
-----
OK

Script complete.
SQL>
```

Un punto a tener en cuenta es que el script solicitará el esquema que servirá para la replicación

- ❖ Se ejecutan el siguiente script (ddl\_setup.sql)

```
SQL> start ddl_setup.sql

GoldenGate DDL Replication setup script

Verifying that current user has privileges to install DDL Replication...

You will be prompted for the name of a schema for the GoldenGate database objects.
NOTE: The schema must be created prior to running this script.
NOTE: On Oracle 10g and up, system recycle bin must be disabled.
NOTE: Stop all DDL replication before starting this installation.

Enter GoldenGate schema name:golden

You will be prompted for the mode of installation.
To install or reinstall DDL replication, enter INITIALSETUP
To upgrade DDL replication, enter NORMAL
Enter mode of installation:INITIALSETUP

Working, please wait ...
Spooling to file ddl_setup_spool.txt

Using GOLDEN as a GoldenGate schema name, INITIALSETUP as a mode of installation.
```



---

Working, please wait ...

RECYCLEBIN must be empty.  
This installation will purge RECYCLEBIN for all users.  
To proceed, enter yes. To stop installation, enter no.

Enter yes or no:**yes**

DDL replication setup script complete, running verification script...  
Please enter the name of a schema for the GoldenGate database objects:  
Setting schema name to GOLDEN

DDLORA\_GETTABLESPACE SIZE STATUS:

Line/pos  
-----  
Error  
-----  
No errors  
No errors

CLEAR\_TRACE STATUS:

Line/pos  
-----  
Error  
-----  
No errors  
No errors

CREATE\_TRACE STATUS:

Line/pos  
-----  
Error  
-----  
No errors  
No errors

TRACE\_PUT\_LINE STATUS:

Line/pos  
-----  
Error  
-----  
No errors  
No errors

INITIAL\_SETUP STATUS:

Line/pos  
-----  
Error  
-----  
No errors  
No errors

DDLVERSIONSPECIFIC PACKAGE STATUS:

Line/pos  
-----  
Error  
-----  
No errors  
No errors

DDLREPLICATION PACKAGE STATUS:

---

```
Line/pos
-----
Error
-----
No errors
No errors

DDLREPLICATION PACKAGE BODY STATUS:

Line/pos
-----
Error
-----
No errors
No errors

DDL HISTORY TABLE
-----
OK

DDL HISTORY TABLE(1)
-----
OK

DDL DUMP TABLES
-----
OK

DDL DUMP COLUMNS
-----
OK

DDL DUMP LOG GROUPS
-----
OK

DDL DUMP PARTITIONS
-----
OK

DDL DUMP PRIMARY KEYS
-----
OK

DDL SEQUENCE
-----
OK

GGS_TEMP_COLS
-----
OK

GGS_TEMP_UK
-----
OK

DDL TRIGGER CODE STATUS:

Line/pos
-----
Error
-----
No errors
No errors

DDL TRIGGER INSTALL STATUS
-----
OK

DDL TRIGGER RUNNING STATUS
-----
ENABLED
```

```
STAYMETADATA IN TRIGGER
-----
OFF

DDL TRIGGER SQL TRACING
-----
0

DDL TRIGGER TRACE LEVEL
-----
0

LOCATION OF DDL TRACE FILE
-----
/u01/app/oracle/admin/orcl/udump/ggs_ddl_trace.log

Analyzing installation status...

STATUS OF DDL REPLICATION
-----
SUCCESSFUL installation of DDL Replication software components

Script complete.
SQL>
SQL>
```

Como observación , se va a pedir el esquema asociado a la replicación

```
Enter GoldenGate schema name:golden
```

Se va a preguntar por el modo de instalación, que para nuestro caso es setup inicial

```
Enter mode of installation:INITIALSETUP
```

Y si deseamos purgar todos los elementos desde la papelera

```
Enter yes or no:yes
```

❖ Ejecutamos el siguiente script (role\_setup.sql)

```
SQL> start role_setup.sql

GG5 Role setup script

This script will drop and recreate the role GGS_GGSUSER_ROLE
To use a different role name, quit this script and then edit the params.sql script
to change the gg_role parameter to the preferred name. (Do not run the script.)

You will be prompted for the name of a schema for the GoldenGate database objects.
NOTE: The schema must be created prior to running this script.
NOTE: Stop all DDL replication before starting this installation.

Enter GoldenGate schema name:golden
Wrote file role_setup_set.txt

PL/SQL procedure successfully completed.

Role setup script complete

Grant this role to each user assigned to the Extract, GGSCI, and Manager processes,
by using the following SQL command:

GRANT GGS_GGSUSER_ROLE TO <loggedUser>

where <loggedUser> is the user assigned to the GoldenGate processes.
```

Como observación , se va a pedir el esquema asociado a la replicación

```
Enter GoldenGate schema name:golden
```

Y solicita que el rol **GGG\_GGSUSER\_ROLE** se lo asignemos a nuestro usuario de replicación , o sea, al usuario golden.

Llevamos a cabo la ejecución

```
SQL> grant GGS_GGSUSER_ROLE to golden;
```

```
Grant succeeded.
```

```
SQL>
```

❖ Por último ejecutamos el scrip ddl\_enable.sql

```
SQL> start ddl_enable.sql
```

```
Trigger altered.
```

```
SQL>
```

---

## 5. Proceso de replicación

Ya estando la base de origen seteada, lo único que nos queda es definir el proceso de replicación , lo anterior se debe expresar en las tablas de origen que deseamos replicar en el destino, para no hacer un ejemplo demasiado complejo , lo que haremos será generar 2 esquemas , uno en el origen y otro en el destino.

Les daremos nombres diferentes con lo cual sabemos que la replicación no para por el nombre del esquema.

Utilizaremos el esquema DATA en el origen y el esquema DATA\_REPLICADO en el destino y les otorgaremos los privilegios correspondientes

### ❖ En la base de datos origen

```
SQL> Create user data identified by data
temporary tablespace temp;
2
```

User created.

```
SQL> grant connect , resource , unlimited tablespace to data;
```

Grant succeeded.

SQL>

### ❖ En la base de datos destino

```
SQL> Create user data_replicado identified by data_replicado
temporary tablespace temp;
2
```

User created.

SQL>

```
SQL> grant connect , resource , unlimited tablespace to data_replicado;
```

Grant succeeded.

SQL>

Para comenzar la replicación , debemos seguir los siguientes pasos

### 5.1. Configurar el Manager Process

Para que el proceso de replicación comience , debemos configurar el Manager Process

#### ❖ Verificar el proceso manager

Para ello ejecutamos el comando **INFO ALL, ALL PROCESSES** (este proceso se debe hacer en ambos nodos)

---

```
GGSCI (oracle10ga.inmotion.cl) 1> info all, all processes
```

```
Program      Status      Group      Lag      Time Since Chkpt
MANAGER      STOPPED
```

```
GGSCI (oracle10ga.inmotion.cl) 2>
```

<b>OBS</b>	Se debe siempre trabajar en el directorio del Home del Golden Gate, sino, no va a aparecer información
------------	--

O podemos utilizar el comando **INFO MANAGER**

```
GGSCI (oracle10ga.inmotion.cl) 16> info manager
Manager is running (IP port oracle10ga.inmotion.cl.7809).

GGSCI (oracle10ga.inmotion.cl) 17>
```

❖ Verificamos los parámetros del Manager Process

Para ello ejecutamos el comando **EDIT PARAMS MGR**

<b>OBS</b>	<p><b>Se puede dar el siguiente error</b></p> <pre>GGSCI (oracle10ga.inmotion.cl) 2&gt; edit params mgr Cannot load ICU resource bundle 'ggMessage', error code 2 - No such file or directory Cannot load ICU resource bundle 'ggMessage', error code 2 - No such file or directory Aborted</pre> <p><b>Pero esto es a raíz de que no estamos posicionados en el directorio del Home del Golden Gate</b></p>
------------	--

Cuando ejecutamos el comando **EDIT PARAMS MGR**, se abre un archivo en formato edición , allí procedemos a colocar la siguiente instrucción

```
PORT 7809
```

El anterior puerto es el defecto, pero el ideal es que sea un puerto que no se este utilizando

La anterior información queda inscrita en un archivo llamado **mgr.prm** en la ruta **\$GOLDEN\_GATE\_HOME/dirprm**

```
[oracle@oracle10ga dirprm]$ more mgr.prm
PORT 7809
[oracle@oracle10ga dirprm]$ pwd
/home/oracle/gg/dirprm
[oracle@oracle10ga dirprm]$
```

- ❖ Damos inicio al Manager Process (en ambos nodos)

Para dar inicio al Manager process , ejecutamos el comando START MANAGER y verificamos su funcionamiento

```
GGSCI (oracle10ga.inmotion.cl) 5> start manager
```

```
Manager started.
```

```
GGSCI (oracle10ga.inmotion.cl) 6>
```

```
GGSCI (oracle10ga.inmotion.cl) 6> info all, all tasks o se puede utilizar info all, all processes
```

```
Program      Status      Group      Lag      Time Since Chkpt
MANAGER      RUNNING
```

```
GGSCI (oracle10ga.inmotion.cl) 7>
```

<b>OBS</b>	Esto se hace tanto en el origen como en el destino, debe arrancarse el Manager Process en ambas instalaciones de Golden Gate
------------	--

## 5.2. Creación del grupo para extraer datos

Para diferenciar múltiples procesos de extracción o de replicación de datos, es que Oracle Golden Gate los agrupa y a los cuales define como GROUPS

Un ejemplo de lo anterior es que podemos generar extracción de data y replicación mediante procesos en paralelo, por lo anterior, podríamos generar 2 grupos de extracción de data y de replicación de data, en la base de datos origen y en la de destino respectivamente.

- ❖ Creamos el grupo de extracción en el origen

Para crear el grupo de extracción ejecutamos los siguientes comandos (sólo en el nodo origen)  
**add extract <nombre grupo de extracción>, tranlog, begin now**

```
GGSCI (oracle10ga.inmotion.cl) 1> add extract extraer1, tranlog, begin now
EXTRACT added.
```

```
GGSCI (oracle10ga.inmotion.cl) 2>
```

Lo anterior implica lo siguiente

**add extract <nombre grupo de extracción> :** Comando para generar un grupo de extracción

---

**tranlog** : Indica que el log de transacciones será el origen de los datos.

**begin now** : Es la posición del primer registro que cuadre con la hora de comienzo de la extracción desde el log de transacciones

❖ Generamos el archivo de registro (trail) del grupo de extracción

Para crear el archivo donde quedará registro de la actividad del grupo de extracción ejecutamos el comando `add extrail <nombre archivo>, extract <grupo extraccion>, megabytes xxx`

```
GGSCI (oracle10ga.inmotion.cl) 5> add extrail /home/oracle/gg/dirdat/hu, extract extraer1  
EXTTRAIL added.
```

```
GGSCI (oracle10ga.inmotion.cl) 6>
```

El archivo generado debe contener solamente 2 caracteres, dado que Golden Gate después lo incrementa con un correlativo

Lo anterior implica lo siguiente

**add extrail <Archivo de registro>** : En este archivo quedará registro de las transacciones

**extract <grupo de extracción>**: Grupo de extracción que va a escribir en el archivo de trail, solamente un grupo de extracción puede escribir en un extract trail

**megabytes xxxx** : Se le puede indicar el tamaño máximo para el archivo que se genera, con lo cual Oracle Golden Gate sólo lo generará hasta ese tamaño y después incrementará el secuencial del archivo, ejemplo hu0001, hu0002, etc

❖ Se editan los parámetros del grupo de extracción que hemos generado (en el nodo origen)

```
GGSCI (oracle10ga.inmotion.cl) 6> edit params extraer1
```

Y dentro de esa edición insertamos los siguientes datos

```
-- Grupo de extraccion  
extract extraer1  
-- Conexion a la base de datos--  
userid golden, password golden  
-- Nombre o IP del host remoto, en donde se replicara la informacion --  
-- Además del puerto en donde se esta ejecutando el Manager Process en el destino  
rmthost 192.168.239.11, mgrport 7809  
--Se especifica el archivo de trail en el destino--  
rmtrail /home/oracle/gg/dirdat/hu  
-- Se puede indicar que se incluyan todas las DDL que se hagan  
-- sobre los objetos del esquema origenmente la  
ddl include mapped objname DATA.*  
-- Se especifican las tablas a ser capturadas, si aparece * es porque se capturarán todas  
-- Se debe indicar el nombre del esquema del origen  
table DATA.*;
```



Después de eso , podemos proceder a ver nuevamente toda la información de nuestro replicador

```
GGSCI (oracle10ga.inmotion.cl) 6> info all, all processes o se puede utilizar info all, all tasks
```

Program	Status	Group	Lag	Time Since Chkpt
MANAGER	RUNNING			
EXTRACT	STOPPED	EXTRAER1	00:00:00	08:54:46

```
GGSCI (oracle10ga.inmotion.cl) 7>
```

### 5.3. Creación del grupo para replicar datos

Una vez generados los grupos (uno o más) para extraer la data, se procede a generar el grupo para replicar la data (esto último sólo en el nodo destino)

- ❖ Para lo anterior debemos agregar un punto de checkpoint para la base de datos destino

```
GGSCI (oracle10gb.inmotion.cl) 1> edit params ./GLOBALS
```

Y se agrega la siguiente información

```
GGSCHEMA golden  
CHECKPOINTTABLE golden.checkpoint
```

El archivo GLOBALS es un archivo de parámetros que se lee cada vez que se invoca al utilitario GGSCI o cuando se ejecuta el MANAGER PROCESS, es como un archivo de parámetros GLOBAL que le pasa los parámetros declarados al MANAGER PROCESS.

El Checkpointtable es un comando para generar una tabla de checkpoint en la base de datos de destino , el proceso de replicación usa esta tabla para mantener un registro de la posición que ha leído en el “trail” , esto es para propósitos de recuperación.

El uso de esta tabla es opcional, ya que el proceso de replicación también deja el punto de checkpoint en disco (en un archivo)

- ❖ Nos conectamos al esquema generado en puntos anteriores en la base de datos destino

```
GGSCI (oracle10gb.inmotion.cl) 1> dblogin userid golden password golden  
Successfully logged into database.
```

```
GGSCI (oracle10gb.inmotion.cl) 2>
```

- ❖ Y procedemos a generar la tabla de checkpoint

```
GGSCI (oracle10gb.inmotion.cl) 2> add checkpointtable golden.checkpoint
```

---

Successfully created checkpoint table GOLDEN.CHECKPOINT.

GGSCI (oracle10gb.inmotion.cl) 3>

- ❖ Generamos a continuación el grupo de replicación

```
GGSCI (oracle10gb.inmotion.cl) 4> add replicat replic1, exttrail /home/oracle/gg/dirdat/hu,  
checkpointtable golden.checkpoint  
REPLICAT added.
```

GGSCI (oracle10gb.inmotion.cl) 5>

Lo anterior implica lo siguiente

**add replicat <nombre grupo de replicación>** : Comando para generar un grupo de replicación

**exttrail** : Indica que el log de transacciones que será escrito

**checkpointtable** : La tabla de checkpoint que será utilizada para llevar registro de las transacciones aplicadas

- ❖ Generamos un archivo de parámetros para nuestro proceso replicador (en el destino)

Le insertamos los datos, para ello ejecutamos el comando `edit params replic1`

```
-- Grupo de replicacion --  
replicat replic1  
--Definiciones en origen y destino, con este comando le indicamos que son iguales  
ASSUMETARGETDEFS  
--Conexion a la base de datos destino --  
userid golden, password golden  
--Aca definimos un archivo en donde se dejan los registros que no se han insertado, por  
ejemplo por  
--errores a nivel de base de datos, es opcional, pero se recomienda su uso  
discardfile /home/oracle/gg/discard/repl_discard.txt, append, megabytes 10  
--Aca especificamos el mapeo en cuanto al esquema origen y el destino  
map DATA.*, target DATA_REPLICADO.*;
```

## 5.4. Comienzo de la extracción y replicación

Para comenzar con la extracción se debe ejecutar el comando `START EXTRACT <nombre extractor>`

```
GGSCI (oracle10ga.inmotion.cl) 3> info all, all processes o se puede utilizar info all, all  
tasks
```

Program	Status	Group	Lag	Time Since Chkpt
MANAGER	RUNNING			
EXTRACT	STOPPED	EXTRAER1	00:00:00	1665:57:00

---

```
GGSCI (oracle10ga.inmotion.cl) 4> start extract extraer1
```

```
Sending START request to MANAGER ...  
EXTRACT EXTRAER1 starting
```

```
GGSCI (oracle10ga.inmotion.cl) 5>
```

<b>OBS</b>	Si el proceso de extracción no comienza al dar un START EXTRACT implica que los parámetros del proceso de EXTRACT generado tienen alguna anomalía
------------	---

Y comenzamos con la replicación con el comando START REPLICAT <nombre replicador>

```
GGSCI (oracle10gb.inmotion.cl) 3> info all, all processes o se puede usar info all, all  
tasks
```

Program	Status	Group	Lag	Time Since Chkpt
MANAGER	RUNNING			
REPLICAT	STOPPED	REPLIC1	00:00:00	1655:32:22

```
GGSCI (oracle10gb.inmotion.cl) 4> start replicat replic1
```

```
Sending START request to MANAGER ...  
REPLICAT REPLIC1 starting
```

<b>OBS</b>	Si el proceso de replicación no comienza al dar un START REPLICAT implica que los parámetros del proceso de REPLICAT generado tienen alguna anomalía
------------	--

## 6. Sincronización de DDL

Oracle Golden Gate es capaz de realizar la sincronización de las sentencias DML y DDL desde el origen hacia el destino no viceversa.

Ya sea, sólo las DDL, sólo las DML o ambas, la sincronización de las sentencias DDL sólo está soportada desde Oracle9i en adelante

Cuando se lleva a cabo la sincronización de las sentencias DDL , Golden Gate deja fuera a cierta cantidad de esquemas que son internos de Oracle, por ejemplo , podemos mencionar :

- ▶ AURORA
- ▶ CTXSYS
- ▶ ODM
- ▶ ORDSYS
- ▶ SYS
- ▶ SYSMAN
- ▶ SYSTEM
- ▶ OUTLN
- ▶ PERFSTAT
- ▶ Etc....

La papelera de reciclaje debe estar inactiva mientras se lleve a cabo la replicación de las sentencias DDL, dado que hay DDL implícitas cuando un objeto es renombrado y convertido a un formato RECYCLE BIN.

Oracle Golden Gate puede dejar sentencias DDL fuera, pero existe una forma de almacenar esas sentencias para posterior consulta, esto se hace ejecutando el script `ddl_ddl2file.sql` con el usuario SYS , tal cual se hace con otros scripts indicados en el punto 4, la información de las sentencias DDL ignoradas quedan en el `USER_DUMP_DEST` de la máquina.

Los objetos a los cuales se les va a aplicar la sentencia DDL deben ser iguales en el origen como en el destino, sino, el replicador quedará en un estado llamado `BENDING`, para soportar la replicación de DDL se debe utilizar el parámetro `ASSUMETARGETDEFS` en los parámetros de replicación .

Cada vez que en el origen se lleve a cabo un movimiento de una tabla mediante el comando `ALTER TABLE MOVE TABLESPACE`, se debe deshabilitar la extracción y replicación, de esta forma el movimiento de la tabla es transparente para Golden Gate.

Todos los comandos `ALTER DATABASE` y `ALTER SYSTEM` no son considerados por Golden Gate como DDL a replicar.