HP NonStop SQL/MX to HP NonStop SQL/MX via Windows ODBC

Objective

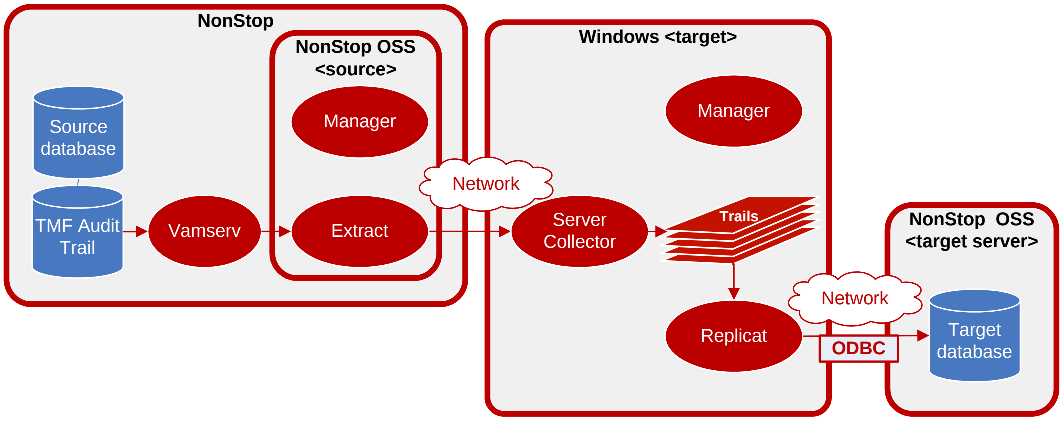
Upon completion of this lesson, you will be able to keep two NonStop SQL/MX databases synchronized.

During this lesson, you will learn how to:

- Prepare your environment to configure the GoldenGate processes
- Configure and execute the initial data synchronization
- Configure and start the change capture of database operations
- Configure and start the change delivery of database operations

NonStop SQL/MX configuration

The following diagram illustrates two HP NonStop SQL/MX databases with the source GoldenGate components installed on the NonStop OSS and the target GoldenGate components installed on a Windows server and communicating with the target database via ODBC.



Note the locations of the **`<source>`** and **`<target>`** GoldenGate systems (where you will be excuting commands) as opposed to the source and target databases that will be accessed by GoldenGate

Overview of Tasks

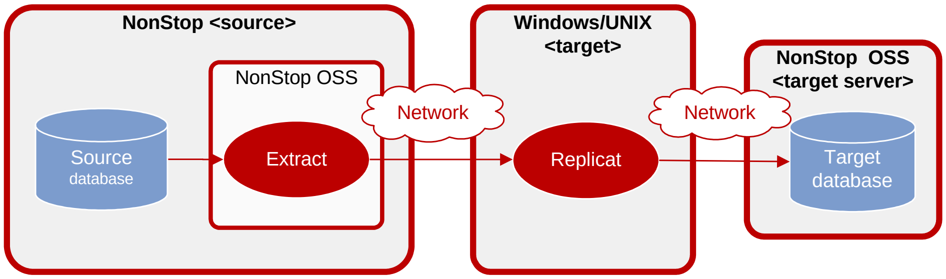
Prepare the Environment

In order to execute this lesson, the GoldenGate application must be installed on both the source NonStop system and the Windows ODBC server as illustrated in the diagram. The installation includes a sample database and scripts to generate initial data as well as subsequent update operations. The source and target tables are created and loaded with initial data. The GoldenGate Manager processes are also started so that other processes may be configured and started.

Configure Initial Data Load

To initially load data over TCP/IP, GoldenGate's Extract can send data to the remote system while the application remains active.

This lesson demonstrates using GoldenGate's direct load method to pull data from the source tables and load it directly to the target system without using the GoldenGate trail.



Configure Change Capture

For audited tables, the Extract process is configured to capture change data from the TMF audit trails via a VAM interface and store the changes in a data queue known as the GoldenGate remote trail.

Configure Change Delivery

Once the tables have been initially loaded with data and the Extract process is capturing all operations, the Replicat process is configured on the Windows ODBC server to deliver the captured operations to the target SQL/MX database.

Exercise 1.
Prepare the Environment

Objective

The goals of this exercise are to:

- Configure and start the Manager processes.
- Prepare the source and target sample databases.

Preparing your source HP NonStop environment

1. Configure and start the Manager process on the source system

The Manager provides a number of important functions, including monitoring critical system components and starting GoldenGate processes.

Before running any other GoldenGate programs on the OSS, you must start Manager. Before starting Manager, you must edit Manger's parameter file and add the PORT.

○ Execute the following commands on the **<source>** to use GGSCI to start the **vi** program for editing the parameters:

```
OSS> cd
OSS> ./ggsci
GGSCI> EDIT PARAMS MGR
```

○ Add the following parameters, then save the file and quit.

```
-- GoldenGate Manager Parameter file
PORT <port>
```

○ Start the Manager on the **<source>** using the following commands.

```
GGSCI > START MANAGER
```

Verify the results:

```
GGSCI> INFO MANAGER
GGSCI> EXIT
```

2. Create and load source files

Execute the following commands on the **<source>** system to create and load the source tables.

- Create the catalog and schema

```
OSS> cd <install location>
OSS> mxci
>> create catalog <catalog>;
>> set catalog <catalog>;
>> create schema <schema>;
>> set schema <schema>;
```

- Create and populate the tables

```
>> obey demo_sqlmx_create.sql;
>> obey demo_sqlmx_insert.sql;
```

- Verify the results:

```
>> select * from tcustmer;
>> select * from tcustord;
>> exit;
```

3. Create source definitions

Note: Source definitions are required even though the source and target databases have the same structure because ASSUMETARGETDEFS is not supported for ODBC.

Execute the following commands on the **<source>** system to create the source definitions file.

- Create the DEFGEN parameter file and add its parameters.

```
OSS> cd <install location>
OSS> ./ggsci
GGSCI> edit param defgen
```

```
DEFSFILE dirdef/source.def, PURGE
SOURCEDB <catalog>
TABLE <schema>.TCUSTMER;
TABLE <schema>.TCUSTORD;
```

Save the file and exit from GGSCI.

- Execute DEFGEN from the command line with the following command.

```
OSS> defgen paramfile dirprm/defgen.prm
```

- Transfer the source definitions to the target system

```
Shell> ftp <target>
Name (<source>:): <login>
Password: password
ftp> ascii
ftp> cd <install location>/dirdef
ftp> lcd <install location>/dirdef
ftp> put source.def
ftp> bye
```

Note: To avoid overlaying existing definition files, always **put** the file as **<db type>.def** instead of source.def (e.g. SMX.def).

Prepare your target Windows SQL/MX environment

1. Configure and start the target Manager

Execute the following commands on the **<target>** system.

Note! The port numbers must be unique if you are installing GoldenGate more than once on the same server.

```
Shell> cd <install location>
Shell> ggsci
GGSCI> EDIT PARAMS MGR
```

- In the parameter file, enter the following parameter, then save and close the file.

```
-- GoldenGate Manager Parameter file
PORT <port>
```

- Start Manager.

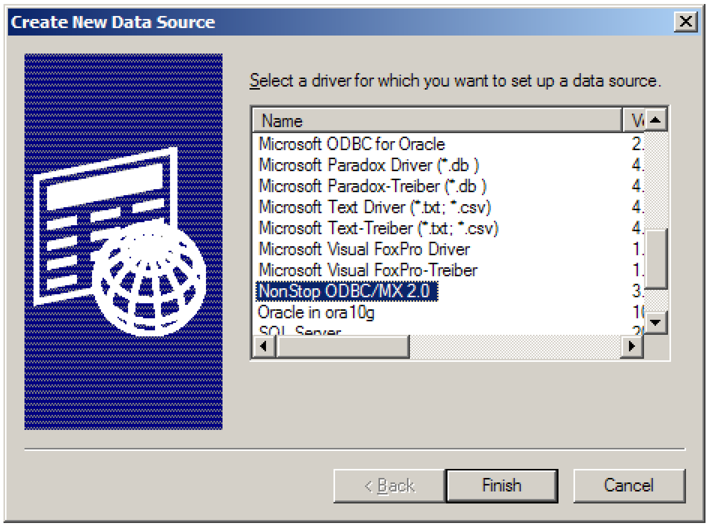
```
GGSCI> START MANAGER
```

- Verify the results:

```
GGSCI> INFO MANAGER
```

2. Configure the ODBC

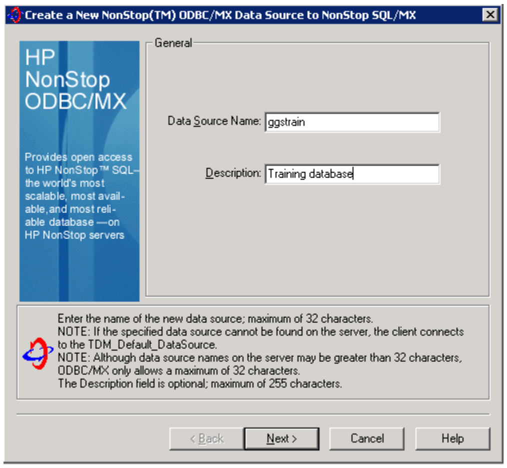
- Go to the ODBC Adminstrator by selecting **Start/Programs/NonStop ODBC-MX 2.0/MS ODBC Administrator**. Click on the **System DSN tab.**



Select a driver for which you want to set up a data source.

Name	V
Microsoft ODBC for Oracle	2.
Microsoft Paradox Driver (*.db)	4.
Microsoft Paradox-Treiber (*.db)	4.
Microsoft Text Driver (*.txt; *.csv)	4.
Microsoft Text-Treiber (*.txt; *.csv)	4.
Microsoft Visual FoxPro Driver	1.
Microsoft Visual FoxPro-Treiber	1.
NonStop ODBC/MX 2.0	3.
Oracle in ora10g	1(
SQL Server	2

- ❍ Press the **Add** button to add a **Data Source Name** (**dsn**). This will display a new window with a list of ODBC drivers.
- ❍ Double-click on the **NonStop ODBC/MX 2.0** driver to select it. This will display the first of a series of windows for entering information about the ODBC.



Create a New NonStop(TM) ODBC/MX Data Source to NonStop SQL/MX

General

Data Source Name: ggstrain

Description: Training database

HP
NonStop
ODBC/MX
Provides open access to HP NonStop™ SQL–the world's most scalable, most available, and most reliable database —on HP NonStop servers

Enter the name of the new data source; maximum of 32 characters.
NOTE: If the specified data source cannot be found on the server, the client connects to the TDM_Default_DataSource.
NOTE: Although data source names on the server may be greater than 32 characters, ODBC/MX only allows a maximum of 32 characters.
The Description field is optional; maximum of 255 characters.

Move through the windows entering the following information and pressing **Next**. The other values can be left to default.

Data Source Name: enter a name that will be your **<dsn>**

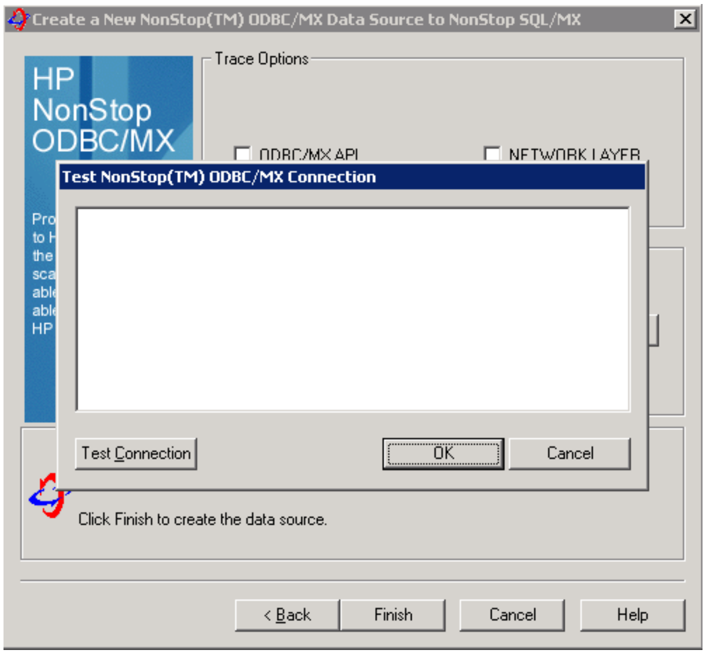
Description: enter text or leave blank

MXCS IP Address: enter the IP address of the **<target server>**; the NonStop SQL/MX system that holds the target database

Catalog: enter the target SQL/MX **<target catalog>**

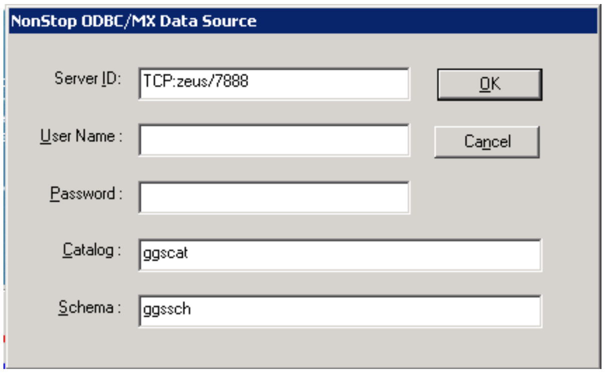
Schema: enter the target SQL/MX **<target schema>**

- ❍ On the last window, press the **Finish** button. A window that allows you to test the connection is displayed.



Click Finish to create the data source.

- Press **Test Connection**. You will need an HP NonStop OSS userid and password to connect. Enter that and press **OK** to perform the test.



Server ID: TCP:zeus/7888

User Name :

Password :

Catalog : ggscat

Schema : ggssch

- This will take you back to the previous window displaying a message on the results of the test.

3. FTP the demo scripts to the NonStop OSS

Execute the following commands to move the demo scripts from the Windows install location to the NonStop OSS.

```
Shell> ftp <target>
Name (<target>:): <login>
Password: password
ftp> ascii
ftp> cd <install location>
ftp> lcd <install location
ftp> put demo_sqlmx_create.sql
ftp> bye
```

Prepare your target HP NonStop SQL/MX database

4. Create the target files

Execute the following command in the **<target>** NonStop OSS space.

- Create the catalog and schema

  ```
  OSS> cd <install location>
  OSS> mxci
  >> create catalog <target catalog>;
  >> set catalog <target catalog>;
  >> create schema <target schema>;
  >> set schema <target schema>;
  ```

- Create the tables

  ```
  >> obey demo_sqlmx_create.sql;
  ```

- Verify the results:

  ```
  >> showddl tcustmer;
  >> showddl tcustord;
  >> exit;
  ```

Exercise 2.
Initial Data Load

Objective

The goal of this exercise is to load the initial data using a pair of Extract and Replicat batch tasks.

Configure initial load capture

1. Add the initial load capture batch task group

Execute the following commands on the **<source>** system to add an Extract group named EINI<unique id>[1].

○ Start the command interface.

```
OSS> ./GGSCI
```

○ Add the Extract batch task.

```
GGSCI> ADD EXTRACT EINI<unique id>, SOURCEISTABLE
```

○ Verify the results:

```
GGSCI> INFO EXTRACT *, TASKS
```

2. Configure the initial load Extract parameter file

Execute the following commands on the **<source>** system.

```
GGSCI> EDIT PARAMS EINI<unique id>
```

```
--
-- GoldenGate Initial Data Capture
-- for tables as listed below
--
EXTRACT EINI<unique id>
SOURCEDB <catalog>
RMTHOST <target>, MGRPORT <port>
RMTTASK REPLICAT, GROUP RINI<unique id>
```

[1] The process names used in lab exercises, for example EINIBD1, are made up of 1) one character to identify the type of process (E for Extract, R for Replicat); 2) three characters to describe the type of process (INI for initial data load, ORA for capture from or delivery to an Oracle database, etc.) and 3) three characters usually made up of the student's initials plus a sequential number to uniquely identify multiple occurrences of that type of process.

```
TABLE <schema>.TCUSTMER;
TABLE <schema>.TCUSTORD;
```

Configure initial load delivery

3. Add the initial load Replicat batch task group

Execute the following commands on the **<target>** system.

```
GGSCI> ADD REPLICAT RINI<unique id>, SPECIALRUN
```

Verify the results:

```
GGSCI> INFO REPLICAT *, TASKS
```

4. Configure the initial load delivery parameter file

Execute the following commands on the **<target>** system.

```
GGSCI> EDIT PARAMS RINI<unique id>
```

Note: ASSUMETARGETDEFS is not supported for ODBC, so source definitions are required even when the source and target databases have the same structure.

```
--
-- GoldenGate Initial Load Delivery
--
REPLICAT RINI<unique id>
DISCARDFILE ./dirrpt/RINI<unique id>.dsc, PURGE
TARGETDB <dsn>, USERID <userid>, PASSWORD <password>
SOURCEDEFS <install location>/dirdef/source.def
MAP <schema>.TCUSTMER, TARGET <schema>.TCUSTMER;
MAP <schema>.TCUSTORD, TARGET <schema>.TCUSTORD;
```

Note: Remember to use <db type>.def if you renamed the source.def when you transferred it to the <target>.

5. Execute the initial load process

Execute the following commands on the **<source>** system.

```
GGSCI> START EXTRACT EINI<unique id>
```

Verify the results:

Execute the following commands on the **<source>** system.

```
GGSCI> VIEW REPORT EINI<unique id>
```

Execute the following commands on the **`<target>`** system.

```
GGSCI> VIEW REPORT RINI<unique id>
```

Exercise 3.
Configure Change Capture

Objective

The goals of this exercise are to:

- Configure and add the Extract process that will capture changes for SQL/MX tables.
- Add the trail that will store the changes.
- Start the Extract process.

Configure change capture

1. Create the Extract parameter file

Execute the following commands on the **<source>** system to define an Extract named ESMX<unique id> that will work with VAMSERV to capture changes to GoldenGate trails.

Note: SQL/MX transactions are logged in the NonStop Guardian TMF audit trails. The VAMSERV program communicates between the OSS Extract and the TMF audit trails.

```
GGSCI> EDIT PARAMS ESMX<unique id>
```

```
--
-- Change Capture parameter file to capture
-- TCUSTMER and TCUSTORD Changes
--
EXTRACT ESMX<unique id>
SOURCEDB <catalog>
RMTHOST <target>, MGRPORT <port>
RMTTRAIL dirdat/<trail id>
TABLE <schema>.TCUSTMER;
TABLE <schema>.TCUSTORD;
```

Note: Record the two characters selected for your <trail id>: _______. You will need this for step 3 and later when you set up the Replicat.

2. Add the Extract group

Execute the following commands on the **<source>** system.

```
GGSCI> ADD EXTRACT ESMX<unique id>, TRANLOG, BEGIN NOW
```

Verify results:

```
GGSCI> INFO EXTRACT ESMX<unique id>
```

3. Define the GoldenGate trail

Execute the following commands on the **<source>** system to add the trail declared in the ESMX<unique id> Extract parameters.

```
GGSCI> ADD RMTTRAIL dirdat/<trail id>, EXTRACT ESMX<unique id>,
MEGABYTES 5
```

4. Start the Extract process

Execute the following commands on the **<source>** system.

```
GGSCI> START EXTRACT ESMX<unique id>
```

Verify the results:

```
GGSCI> INFO ESMX<unique id>
```

Discussion points

1. Identifying a remote system

What parameters are used to identify the remote target system?

____

____

2. Sizing the GoldenGate trail

Where do you set how large a GoldenGate trail file may get before it rolls to the next file? What option do you use?

____

____

Exercise 4.
Configure Change Delivery

Objective

The goal of this exercise is to configure and start the delivery process.

Set up the checkpoint table

1. Create a GLOBALS file on the target system

Execute the following commands on the **<target>** system.

- Create and edit the **GLOBALS** parameter file to add the checkpoint table.

```
Shell> cd <install location>
Shell> ggsci
GGSCI> EDIT PARAMS ./GLOBALS
```

In the text editor, type:

```
CHECKPOINTTABLE <catalog>.<schema>.ggschkpt
```

- Record the checkpoint table owner and name, then save and close the file.

Table owner ____________________ name ____________________

Note: You could name the table anything you want, but for training purposes we are using **ggschkpt**.

- Verify that the **GLOBALS** file was created in the root GoldenGate directory, and remove any file extension that was added.

2. Activate the GLOBALS parameters

For the **GLOBALS** configuration to take effect, you must exit the session in which the changes were made. Execute the following command to exit **GGSCI**.

```
GGSCI> EXIT
```

3. Add a Replicat checkpoint table

On the **<target>** system, execute the following commands in **GGSCI**:

```
Shell> cd <install location>
Shell> ggsci
GGSCI> DBLOGIN SOURCEDB <dsn> USERID <login>, PASSWORD <password>
GGSCI> ADD CHECKPOINTTABLE
```

Configure change delivery

2. Create delivery parameter file

Execute the following commands on the **<target>** system to define a Replicat named RSMX<unique id> to pull data from the GoldenGate trails and apply it to the target files.

Note: ASSUMETARGETDEFS is not supported for ODBC, so source definitions are required even when the source and target databases have the same structure.

```
GGSCI> EDIT PARAM RSMX<unique id>
```

```
--
-- REPLICAT parameter file to synchronize
-- TCUSTMER and TCUSTORD changes
--
REPLICAT RSMX<unique id>
HANDLECOLLISIONS
DISCARDFILE dirrpt/RSMX<unique id>, PURGE
TARGETDB <dsn>, USERID <userid>, PASSWORD <password>
SOURCEDEFS dirdef/source.def
MAP <schema>.TCUSTMER, TARGET <schema>.TCUSTMER.*;
MAP <schema>.TCUSTORD, TARGET <schema>.TCUSTORD.*;
```

Note: Remember to use <db type>.def if you renamed the source.def when you transferred it to the <target>.

3. Add the delivery group

Execute the following commands on the **<target>** system to add a delivery group named RSMX<unique id>.

```
GGSCI> ADD REPLICAT RSMX<unique id>, EXTTRAIL dirdat/<trail id>
```

Note: Refer to your Extract set up for the correct two-character <trail id>.

4. Start the delivery process

Execute the following commands on the **<target install>** system.

```
GGSCI> START REPLICAT RSMX<unique id>
```

Verify results:

```
GGSCI> INFO REPLICAT RSMX<unique id>
```

Discussion points

Search in the *Reference Guide* for the following parameters that were included in the delivery parameter file.

1. When to use HANDLECOLLISIONS

For which stage of GoldenGate processing – change capture, initial data load, or change delivery – would you use HANDLECOLLISIONS?

____

____

2. .What information is supplied by SOURCEDEFS

____

____

3. What is the purpose of the DISCARDFILE?

____

____

Exercise 5.
Generate Activity and Verify Results

Objective

The goals of this exercise are to:

- Execute miscellaneous update, insert, and delete operations on the source system.
- Verify the delivery of the changes to the target.
- Turn off the error handling used for the initial load.

Generate database operations

1. Execute miscellaneous update, insert, and delete operations

Execute the following commands on the **<source>** system to run a script that will generate activity in the source SQL/MX database.

```
OSS> mxci
>> SET CATALOG <catalog>;
>> obey demo_sqlmx_misc.sql;
```

Verify change capture and delivery

2. Verify your results on the source system

Execute the following commands on the **<source>** system.

```
>> SELECT * FROM <schema>.TCUSTMER;
>> SELECT * FROM <schema>.TCUSTORD;
>> EXIT;
```

3. Verify your results on the target system

Execute the following commands on the **<target>** system to verify the target data.

```
OSS> ./mxci
>> SET CATALOG <catalog>;
>> SELECT * FROM <schema>.TCUSTMER;
>> SELECT * FROM <schema>.TCUSTORD;
>> EXIT;

OSS> ./GGSCI
```

```
GGSCI> SEND REPLICAT RSMX<unique id>, REPORT
GGSCI> VIEW REPORT RSMX<unique id>
```

Turn off error handling

4. Turn off initial load error handling for the running delivery process

```
GGSCI> SEND REPLICAT RSMX<unique id>, NOHANDLECOLLISIONS
```

5. Remove initial load error handling from the parameter file

```
GGSCI> EDIT PARAMS RSMX<unique id>
```

Remove the HANDLECOLLISIONS parameter.