Replicate data in the fast lane

DB2 Information Management Software

http://www-136.ibm.com/developerworks/db2

Table of contents

If you're viewing this document online, you can click any of the topics below to link directly to that section.

Section 1. Before you start

About this tutorial

Q replication is a new, high-speed technology for moving transactions between databases managed by IBM DB2 Universal Database. It's a highlight of the just-released DB2 Information Integrator Version 8.2. One of the reasons Q replication is so fast is that it uses the powerful business integration product WebSphere MQ to zoom data across networks in the form of compact messages. (*Fast* means the product replicates a high volume of database changes with low *latency*, or delay.)

In this tutorial, you'll learn how to set up a straightforward Q replication scenario: *one-way* or *unidirectional* replication between two remote databases. As a bonus, you'll learn how to create and use WebSphere MQ queue managers, message queues, and channels that provide the transport layer for replication, as Figure 1 shows.

Figure 1. Q replication components
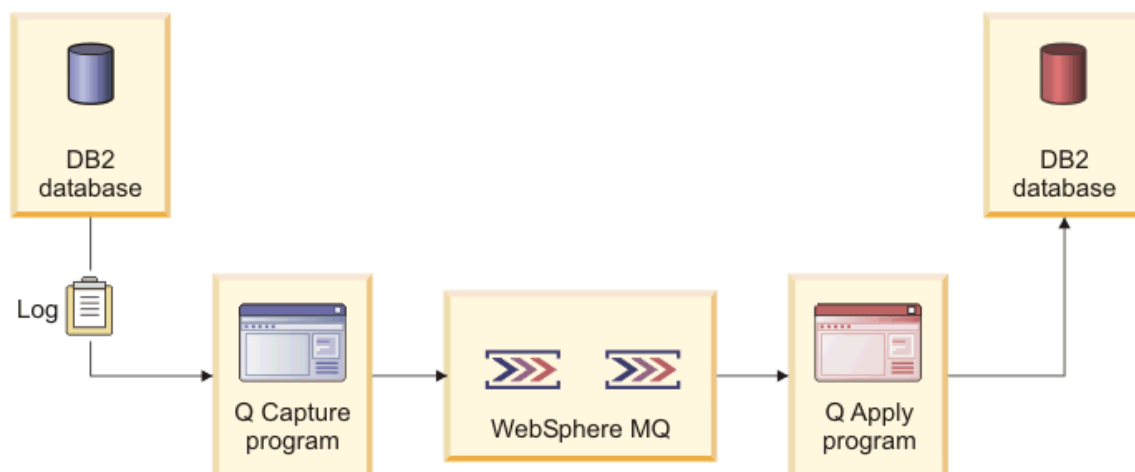
You will only need one computer to take the tutorial, but you will set up WebSphere MQ and Q replication to simulate data movement between two remote systems so that you can apply what you learn to a realistic production scenario. Along the way, you'll get a guided tour of the Replication Center and the tools and wizards that make it easy to configure and administer Q replication.

By the end of this tutorial, you will know:

° How to create and use WebSphere MQ queue managers, queues, and channels.
° How to enable a DB2 database for replication
° How to set up and operate the Q Capture and Q Apply programs

° How to create a replication queue map to link the source and target databases
° How to create a Q subscription to map a source table to a target table

If you have a basic knowledge of DB2 Universal Database (how to create a database, view the contents of a table using commands or the Control Center, and issue some simple SQL statements), then this tutorial will show you everything else you need to get WebSphere MQ and Q replication up and running.

Prerequisites

To complete the steps in the tutorial, you will need the following software:

° IBM DB2 Information Integrator Replication Edition for Linux, UNIX, and Windows Version 8.2
° IBM DB2 Universal Database for Linux, UNIX, and Windows Version 8.2
° IBM WebSphere MQ Version 5.3

Look for a trial version of the Replication Edition, and links to trial versions of DB2 UDB and WebSphere MQ, by following the "Trials and betas" link on the *Replication Edition* product Web site.

This tutorial will work on any of the supported platforms for DB2 Information Integrator Replication Edition V8.2.

After you install DB2 UDB, you need to do two things:

° Create the SAMPLE database by clicking **Create Sample Database** in the DB2 First Steps Launchpad or using the **db2sampl** command.
° Create another database named TARGET within the default DB2 instance.

About the author

Dell Burner is a technical writer on the DB2 Information Integrator replication team at the Silicon Valley Laboratory in San Jose, California. He is co-author of the *DB2 Information Integrator Replication and Event Publishing Guide and Reference* and author of *DB2 Information Integrator Tuning for Replication and Event Publishing Performance*.

For technical questions or comments about the content of this tutorial, contact Dell at *dellb@us.ibm.com*.
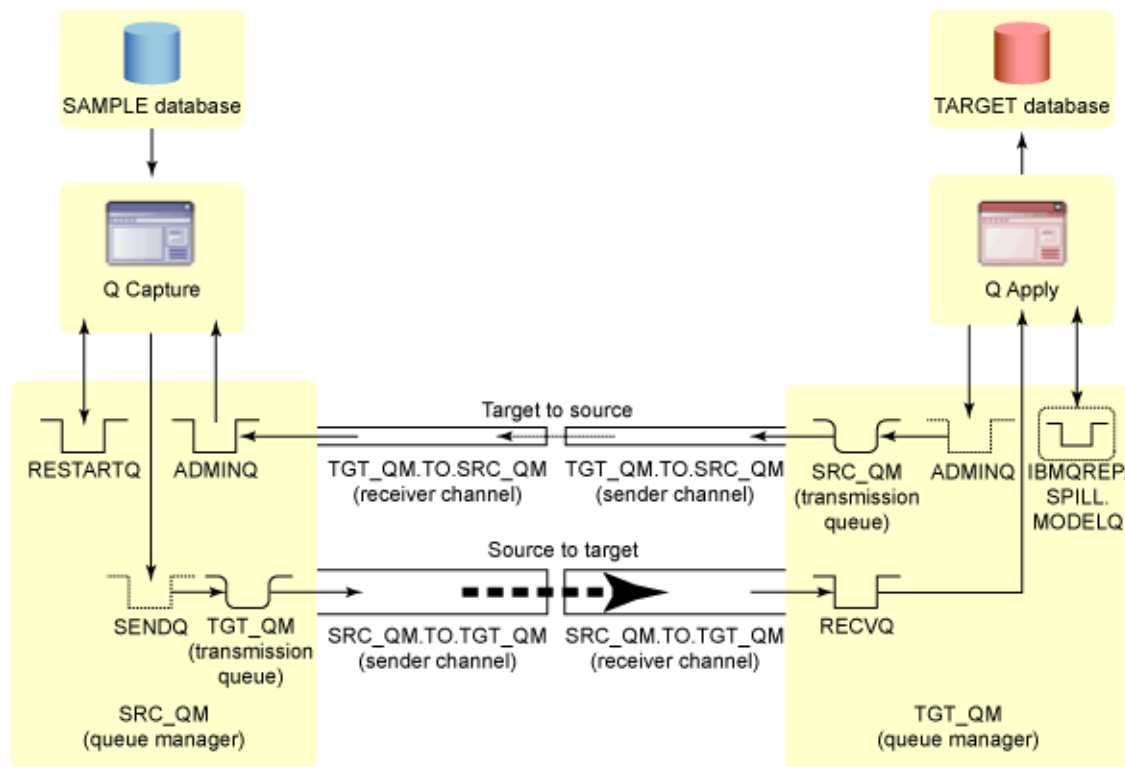
Section 2. Set up WebSphere MQ

Create the source and target queue managers

WebSphere MQ enables software applications to exchange data and communicate using messages and message queues. You will start by creating the "brain" of the messaging system, the *queue manager*. One queue manager will manage queues and messages for the source database, and the other will manage queues and messages for the target database.

Figure 2 shows the SRC_QM queue manager at the source and TGT_QM queue manager at the target. The two queue managers interact with the Q Capture and Q Apply programs, queues, and channels to rapidly move data in the form of messages.

Notice there are two paths for messages: one for replicated data and communication from the source, and one that the Q Apply program uses to communicate with the Q Capture program. The source-to-target channel is depicted as larger because it carries replicated transactions, the bulk of the messaging "action."

Figure 2. WebSphere MQ queue managers, queues, and channels

As you go through the setup steps for WebSphere MQ, you will see repeated sections of this figure to help you focus on the queues and channels that you

are creating at each phase of the setup.

To create and start the queue managers, type (or copy and paste) the following commands in an operating system command window:

1. To create the source queue manager:

    ```
    crtmqm SRC_QM
    ```

2. To create the target queue manager:

    ```
    crtmqm TGT_QM
    ```

3. To start the source queue manager, and then the target queue manager:

    ```
    strmqm SRC_QM
    strmqm TGT_QM
    ```

Now that you have created and started the two queue managers, you can use them to create the queues and channels that you'll need.


Create the source queues

A *queue* is a data structure used to store messages. Each queue is owned by a queue manager. You will use the SRC_QM queue manager to define four queues:

°   A remote queue definition (known as a *send queue*) that the Q Capture program uses to direct messages containing replicated data to the *receive queue* at the target
°   A local *transmission queue* to hold messages before they cross the channel
°   A local *administration queue* that the Q Capture program uses to receive communications from the Q Apply program at the target
°   A local *restart queue* to store restart information for the Q Capture program

Figure 3. Message queues at the source
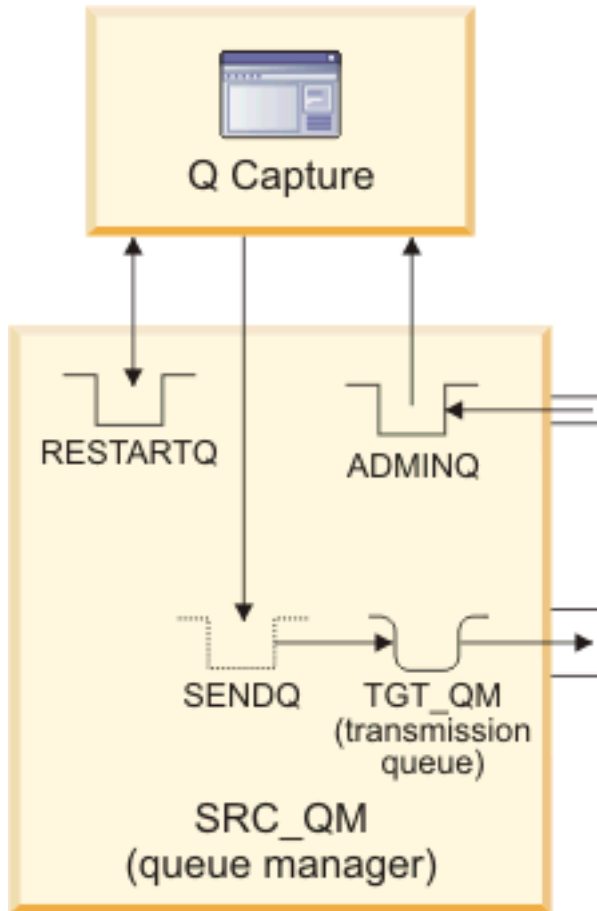
Use the following commands:

1. To start an interactive session with the SRC_QM queue manager so that you can define the queues: **runmqsc SRC_QM**. Your command window changes from an operating system prompt to a **runmqsc** session, as Figure 4 shows.
   Figure 4. runmqsc prompt at beginning of interactive session

2. To create the administration queue, type or paste the following command:

```
DEFINE QLOCAL ('ADMINQ') DEFPSIST(YES)
```

3. To create the restart queue:

```
DEFINE QLOCAL ('RESTARTQ') DEFPSIST(YES)
```

4. To create the transmission queue (`xmitq`):

    ```
    DEFINE QLOCAL('TGT_QM') USAGE(XMITQ) DEFPSIST(YES)
    ```

    Tip: Naming the transmission queue after the remote queue manager

    (TGT_QM) is not required, but this WebSphere MQ convention helps you
    remember the destination for messages. DEFPSIST(YES) means messages
    that use this queue are persistent; that is, they will be logged for recovery in
    case of a system failure or shutdown.

5. To create the remote queue definition that points to the receive queue at the
    target:

    ```
    DEFINE QREMOTE('SENDQ') RNAME('RECVQ') RQMNAME('TGT_QM')
    XMITQ('TGT_QM') DEFPSIST(YES)
    ```

    A remote queue definition names the destination queue (`RNAME`), the
    destination queue manager (`RQMNAME`) and the transmission queue that will
    be used (`XMITQ`).

Next, you will create the queues at the target.

Create the target queues

In this section, you will use the TGT_QM queue manager to define four queues:

°   A local queue to receive messages from the Q Capture program (a *receive
    queue*)
°   A remote queue definition that points to the source administration queue,
    and a local transmission queue to hold the administrative messages
°   A model queue definition for *spill queues*, which make it possible to load
    data at the same time data is being replicated. Spill queues are created
    dynamically to hold any transactions that arrive at the source while the target
    table is being loaded.

Figure 5. Message queues at the target
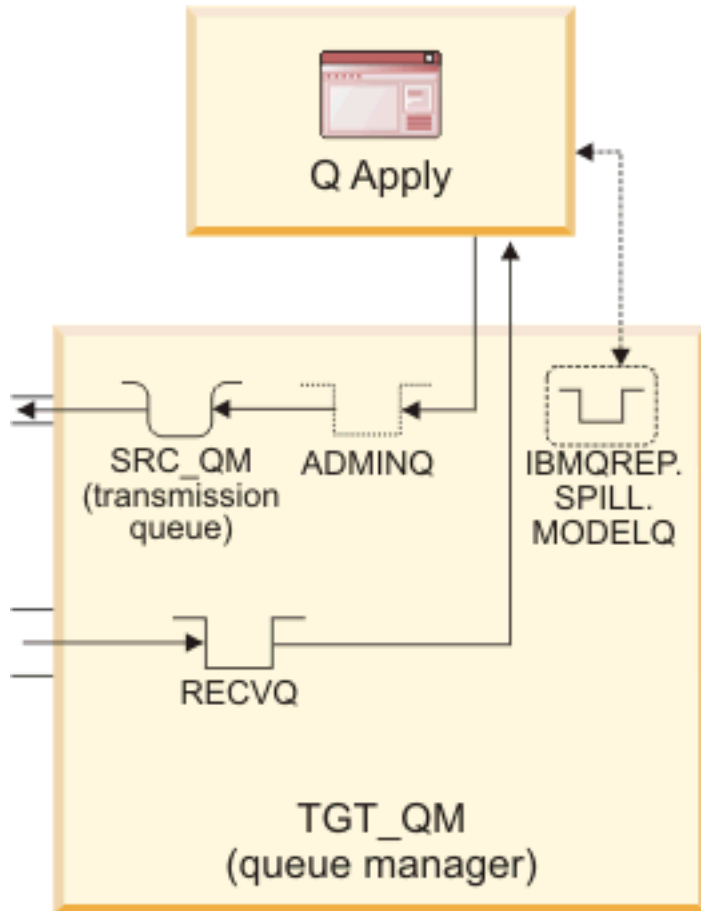
To create the queues, enter the following commands:

1. If your **runmqsc** session with the SRC_QM queue manager is still active,
   use the **end** command to stop the interactive session.
2. To start an interactive session with the TGT_QM queue manager:

   ```
   runmqsc TGT_QM
   ```

3. To create the receive queue:

   ```
   DEFINE QLOCAL('RECVQ') DEFPSIST(YES)
   ```

4. To create the target transmission queue:

   ```
   DEFINE QLOCAL('SRC_QM') USAGE(XMITQ) DEFPSIST(YES)
   ```

   Tip: Again, the transmission queue is named after the remote queue

   manager (SRC_QM).

5. To create the remote queue definition that that points to the source
   administration queue:

```
DEFINE QREMOTE('ADMINQ') RNAME('ADMINQ') RQMNAME('SRC_QM')
XMITQ('SRC_QM') DEFPSIST(YES)
```

6. To create the model queue definition for spill queues:

```
DEFINE QMODEL('IBMQREP.SPILL.MODELQ') DEFSOPT(SHARED)
MSGDLVSQ(FIFO) DEFTYPE(PERMDYN)
```

The source and target queues are ready. Next, you will create the channels to connect them.

Create WebSphere MQ channels

WebSphere MQ uses channels to transmit messages between queue managers. In this section, you will create the channels between the source and target queue managers in both directions.

Tip: Each channel has two ends: A sender channel defined within the originating queue manager, and a receiver channel defined within the destination queue manager.

Figure 6. Message channels between the source and target
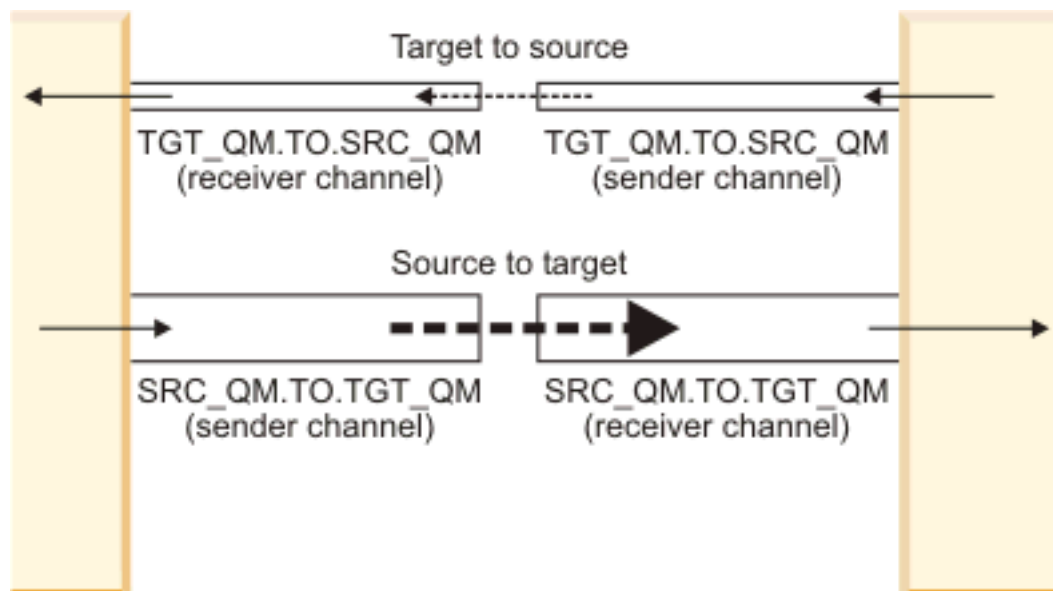
Before you begin: To differentiate the channels, you will need to identify two unused ports on your computer because both the source and target queue managers are on a single computer with one IP address. Typically, ports 1450 and 1451 are free and can be used for this tutorial.

° To check whether the ports are available on Windows, open the `C:\WINNT\system32\drivers\etc\services` file and use **Edit -> Find** (or **Ctrl+F**) to search for 1450 and 1451.
° On Linux or UNIX, use this command: **cat /etc/services | egrep "1450|1451"**

If these ports are already assigned, choose different ports and use the same procedure to verify that they are available.

Channel objects at the source

Within the source queue manager, you will define a sender channel for transactions going to the target, and a receiver channel for administrative messages from the Q Apply program.

To define the channels:

1. If your **runmqsc** session with the TGT_QM queue manager is still active, use the **end** command.
2. Determine your computer's IP address:
   ° **Windows:** Use the **ipconfig** command.
   ° **Linux, UNIX:** Use the the **ifconfig -a** command.
3. Start an interactive session with the SRC_QM queue manager by entering this command:

```
runmqsc SRC_QM
```

4. Define the sender channel from the SRC_QM queue manager to the TGT_QM queue manager (using the TCP/IP transmission protocol) by entering the following command:

> **Important:** Replace *IP address* with your computer's IP address, determined in Step 2, and replace *port* with 1450 or an alternate port that you determined earlier.

```
DEFINE CHL ('SRC_QM.TO.TGT_QM') CHLTYPE(SDR) TRPTYPE(TCP) CONNAME
('IP address(port)') XMITQ('TGT_QM') DISCINT(0)
```

5. Define a receiver channel at the SRC_QM queue manager from the TGT_QM queue manager by entering this command:

```
DEFINE CHL ('TGT_QM.TO.SRC_QM') CHLTYPE(RCVR) TRPTYPE(TCP)
```

Channel objects at the target

Within the target queue manager, you will define a sender channel for communicating with the Q Capture program, and a receiver channel for transaction messages from the Q Capture program.

Enter the following commands:

1. If your **runmqsc** session with the SRC_QM queue manager is still active, use the **end** command.
2. To start an interactive session with the TGT_QM queue manager:

    ```
    runmqsc TGT_QM
    ```

3. To define a sender channel from the TGT_QM queue manager to the SRC_QM queue manager:

    ```
    DEFINE CHL ('TGT_QM.TO.SRC_QM') CHLTYPE(SDR) TRPTYPE(TCP)
    CONNAME('IP address(port)') XMITQ('SRC_QM') DISCINT(0)
    ```

    Important: Replace *IP address* with your computer's IP address, and

    replace *port* with 1451 or an alternate port that you determined earlier.

4. To define a receiver channel at the TGT_QM queue manager from the SRC_QM queue manager:

    ```
    DEFINE CHL ('SRC_QM.TO.TGT_QM') CHLTYPE(RCVR) TRPTYPE(TCP)
    ```

5. Use the **end** command to end your **runmqsc** session with the TGT_QM queue manager.

At this point you have successfully created the WebSphere MQ objects that you will need to bridge the source and target databases. Next, you will begin working with the Q replication programs.

Section 3. Set up Q replication

Create the source and target control tables

The Q Capture program at the source database and Q Apply program at the target database use *control tables* to store important information such as which tables to capture data from, which queues to use, and which tables to apply replicated data to.

In this section, you will create control tables for the Q Capture program and Q Apply program.

Make sure that DB2 Universal Database is running before you proceed. You can use the **db2start** command or the Control Center.

Q Capture control tables

1. Launch the Replication Center: You can use the **db2rc** command on all platforms, or from the Windows **Start** menu, click **Programs -> IBM DB2 -> General Administration Tools -> Replication Center**.
2. On the Replication Center Launchpad, click the Q replication icon.
3. Click **Create Q Capture Control Tables**.
4. On the Start page of the Create Q Capture Control Tables wizard, accept the default choice of **Typical**, and click **Next**.
5.
   On the Server page, click the [...] button next to the **Q Capture server** field, and select the SAMPLE database. Type a valid user ID and password for connecting to the database, accept the default **Q Capture schema** of ASN, and click **Next**.
   Figure 7. Entering server information in Create Q Capture Control Tables wizard

Q Capture server	SAMPLE	...	(SAMPLE)
User ID	db2admin		
Password	********		
Q Capture schema	ASN		

    Tip: The term *server* in the Replication Center refers to a DB2 UDB

database on Linux, UNIX, and Windows, and to a DB2 UDB subsystem on z/OS. *Schema* refers to an instance of the Q Capture or Q Apply program,

including its set of control tables. You can have multiple schemas on a server.

6. On the Enter Queues page, use the **Queue manager** field to type the name of the SRC_QM queue manager and the ADMINQ and RESTARTQ queues that you created in the remaining fields. Click **Next**.
Figure 8. Entering names for the queue manager and queues

7. Click **Finish** to generate the SQL script for creating the control tables, and then close the Messages and SQL Scripts window.
8. On the Run Now or Save SQL window, click **OK**. You should see a confirmation message that the scripts ran successfully.

Q Apply control tables

With the Replication Center Launchpad visible on your screen, follow these steps to create the Q Apply control tables. (if the Launchpad is no longer on your screen, from the Replication Center menu click **Replication Center -> Launchpad**).

1. Click **Create Q Apply Control Tables**.
2. On the Start page of the Create Q Apply Control Tables wizard, accept the default choice of **Typical**, and click **Next**.
3.
   On the Server page, click the [...] button next to the **Q Apply server** field, and select the **TARGET** database. Type a valid user ID and password for connecting to the database, accept the default **Q Apply schema** of ASN, and click **Next**.
   Figure 9. Entering server information in the Create Q Apply Control Tables wizard

Q Apply server	TARGET	•••	(TARGET)
User ID	db2admin		
Password	********		
Q Apply schema	ASN		

4. On the Enter Queues page, type the name of the TGT_QM queue manager that you created in the **Queue manager** field, then click **Next**.
5. Click **Finish** to generate the SQL script for creating the control tables, and then close the Messages and SQL Scripts window.
6. On the Run Now or Save SQL window, click **OK**. You should see a confirmation message that the scripts ran successfully.

Enable the source database for replication

To replicate transactions, you need to turn on archive logging for the SAMPLE database. Archive logging ensures that the DB2 recovery log will not be overwritten with new log records, and that you will not lose data needed to keep the source and target tables synchronized.

To turn on archive logging at the SAMPLE database, follow these steps:

1. Close the Replication Center Launchpad, and in the Replication Center

   object tree, use the ⊞ signs to open folders in the following order: **Q replication -> Definitions -> Q Capture Servers -> SAMPLE**.
2. Right-click the SAMPLE icon and click **Enable database for replication**.
3. In the Turn On Archive Logging window, click **OK**.
   The Replication Center does the following:

   ° Forces all applications off the SAMPLE database
   ° Turns on archive logging (which puts the database in a backup pending state)
   ° Backs up the database to the default DB2 installation directory
   ° Deletes the backup.

   If you want to save the backup, you select **Use Configure Logging Wizard** to launch the DB2 wizard that lets you specify backup options.

   Tip: The backup process can take several minutes. After it completes,

   notice how the SAMPLE icon under **Q Capture Servers** in the Replication

Center object tree turns color to signal that the database is enabled for replication.

Figure 10. SAMPLE server icon indicating database is enabled for replication

Create a replication queue map

A *replication queue map* ties together the send queue at the source and the receive queue at the target to define a pathway for replicated data. You can replicate one or more tables using the same replication queue map.

In this section, you will create a replication queue map to link the SAMPLE database with the TARGET database.

Follow these steps:

1.

In the Replication Center object tree, use the ⊞ signs to open folders in the following order: **Q replication -> Definitions -> Q Capture Servers -> SAMPLE -> Q Capture Schemas -> ASN**.
Figure 11. Replication Center object tree opened to display Replication Queue Maps folder
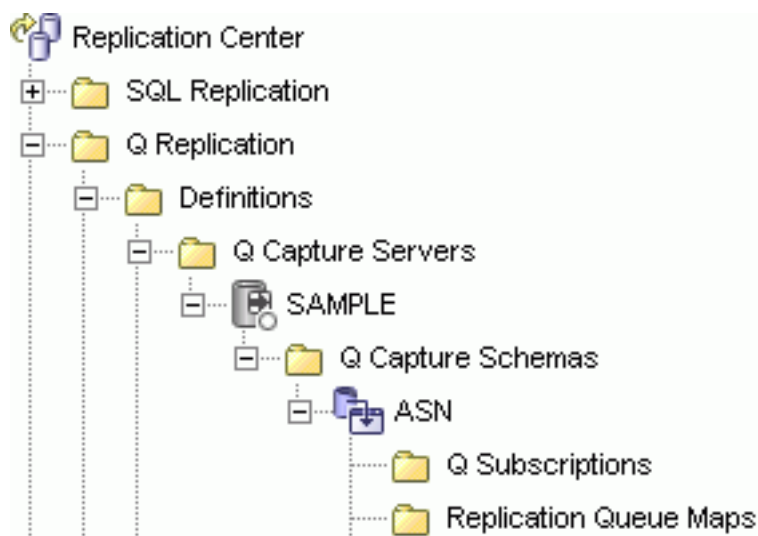
2. Right-click the **Replication Queue Maps** folder, and click **Create**. The Create Replication Queue Map window opens.

3. On the General page, type **EMPLOYEE_QMAP** as the name for the replication queue map.

4.
   On the Properties page, click the [...] button, and in the Select a Q Apply Server window click the TARGET database and then click **OK**.

5. Enter the names of the following three queues that you created earlier in the appropriate fields:
   ° SENDQ in the **Send queue** field
   ° RECVQ in the **Receive queue** field
   ° ADMINQ in the **Administration queue** field

   Figure 12. Entering target and queue information on Create Replication Queue Map window
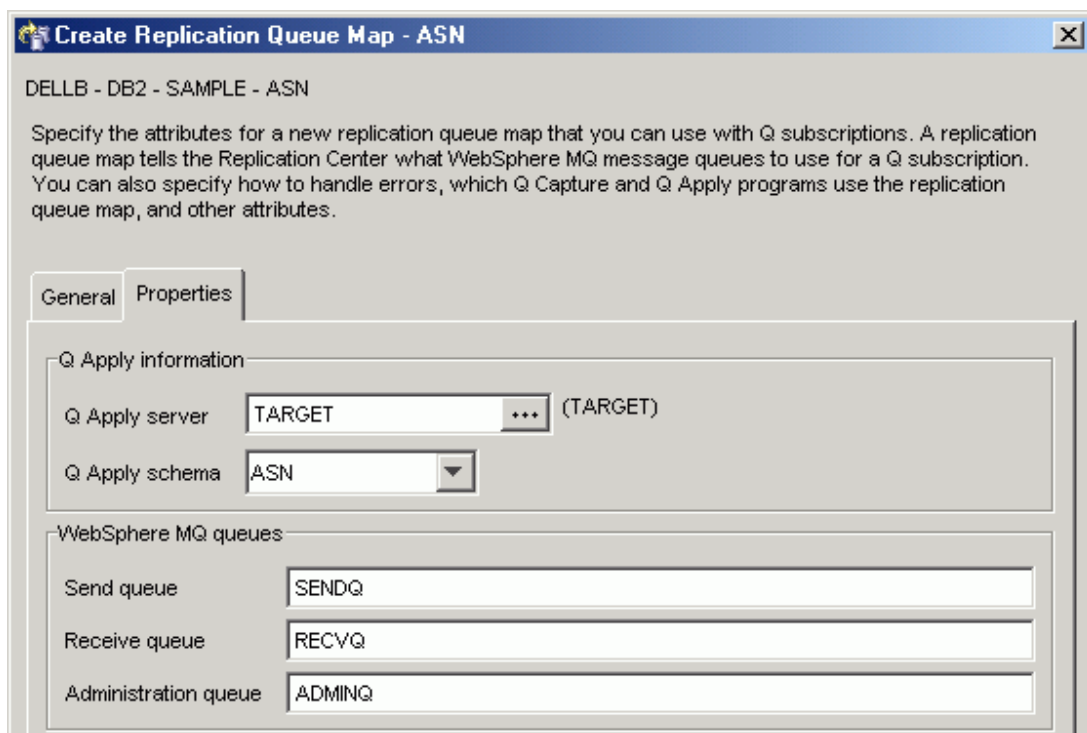
6. Accept the other defaults on the Properties page, and click **OK**.

7. Close the Messages and SQL Scripts window, and on the Run Now or Save SQL window, click **OK**.

Create a Q subscription

A *Q subscription* maps the table that contains your source data to a replicated copy of that table at the target database (or in advanced scenarios, to stored

procedures). You specify the queues that will be used to transmit data, the rows and columns that you want to replicate, and other preferences. You create a Q subscription for each table that you want to replicate.

In this section, you will create a Q subscription to map the EMPLOYEE table in the SAMPLE database with a copy called TGT_EMPLOYEE at the TARGET database.
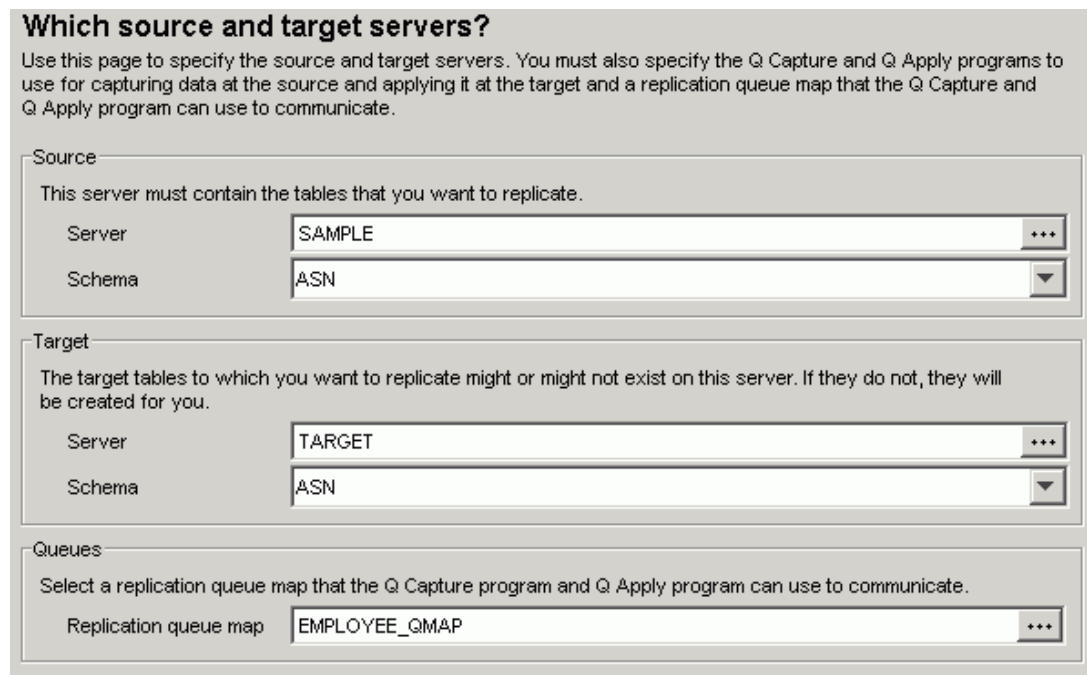
Follow these steps:

1. On the Replication Center Launchpad, click **Create a Q Subscription**.
2. On the Create Q Subscription wizard, click **Next** to open the Replication page. Accept the default replication type of **Unidirectional**, and click **Next**.
3. On the Servers page, verify that the SAMPLE and TARGET databases

   appear in the **Servers** fields for source and target, and then click the [...]

   button next to the **Replication queue map** field.
4. On the Select Replication Queue Map window, select EMPLOYEE_QMAP and click **Next**.
   Figure 13. Entering source, target, and queue information on Create Q Subscription wizard

5. On the Source Tables page, click **Add**.
6. On the Select Source Table window, click **Retrieve All**.
7. Scroll until you find the EMPLOYEE table, select it, and click **OK**. Click **Next**.
8. On the Target page, accept the default setting of **Create a new table**, change the **Table name** to **TGT_EMPLOYEE**, and click **Next**.

Tip: The Replication Center will automatically create the target table for you and handle all of the mappings between columns, data types, and attributes.

9.
   On the Rows and Columns page, click the [...] button next to the **Index or primary key** field.

10. On the Select Target Table Index or Primary Key window, click **Create a unique index on the target table**. Then use the [>] button to move the EMPNO column into the **Selected** area, and click **OK**.
   Figure 14. Selecting the EMPNO column as the unique index for the new TGT_EMPLOYEE table
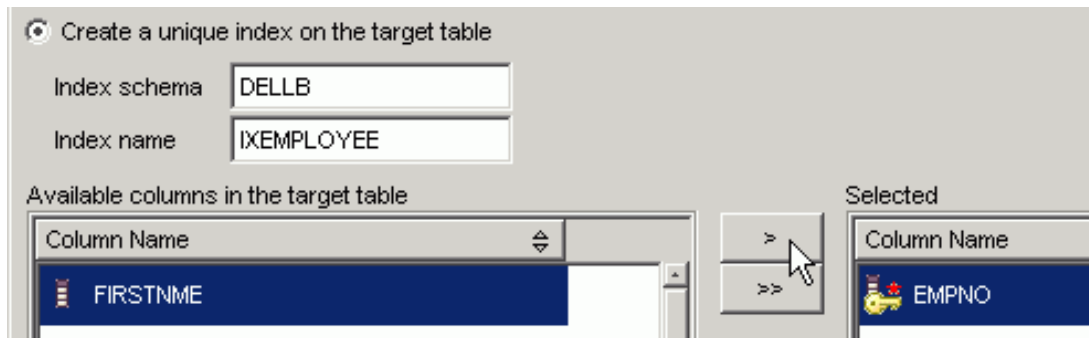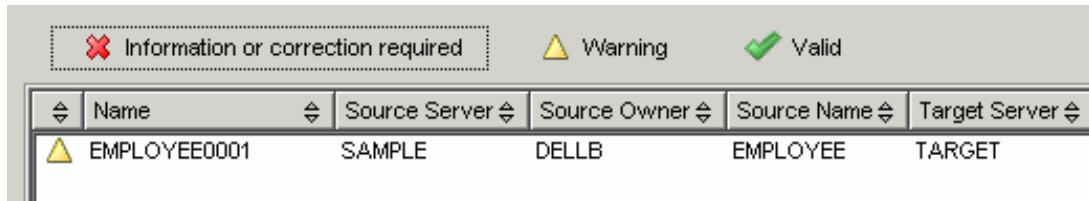

11. Click **Next**, and on the Unexpected Conditions page of the wizard, accept the defaults and click **Next**.

12. On the Loading target tables page, accept the defaults and click **Next**.

   Tip: The default setting for the load process tells the Q Apply program to automatically copy the data from the EMPLOYEE table to the new TGT_EMPLOYEE table using the best available utility for your computer. For new Q subscriptions with the default setting, this target table loading process happens when the replication programs start.

   On the Review Q Subscriptions page, a yellow warning symbol appears next to the EMPLOYEE0001 Q subscription. This occurs because the EMPLOYEE table in the SAMPLE database does not have a primary key. Either the source or target table must have a unique constraint on the key that is used for replication. When the Replication Center creates the TGT_EMPLOYEE target table, it will create a primary key on the EMPNO column.
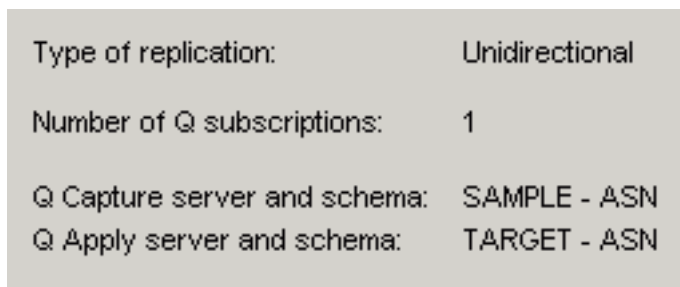
   Figure 15. Warning icon on Review Q Subscriptions page

	Name		Source Server	Source Owner	Source Name	Target Server
⚠	EMPLOYEE0001		SAMPLE	DELLB	EMPLOYEE	TARGET

Legend above table: ✖ Information or correction required   ⚠ Warning   ✔ Valid

13. Click **Next** to get to the Summary page, review your choices and click **Finish**.
    Figure 16. Summary page

Type of replication:	Unidirectional
Number of Q subscriptions:	1
Q Capture server and schema:	SAMPLE - ASN
Q Apply server and schema:	TARGET - ASN

14. Close the Messages and SQL Scripts window, and on the Run Now or Save SQL window, click **OK**.

You have now defined the mapping between the EMPLOYEE table and the newly created TGT_EMPLOYEE table. You have also defined the queue pathway that will be used to replicate data between the two tables. You are just about ready to replicate!

Section 4. Start Q replication

Start WebSphere MQ channels and listeners

At this point, you will start your two WebSphere MQ channels at the sender ends of each channel, and start a *listener* at the receiving ends of each channel to monitor for incoming messages.

Follow these steps:

1. Make sure that the SRC_QM queue manager at the source and TGT_QM queue manager at the target are running by issuing these commands:

    ```
    strmqm SRC_QM
    strmqm TGT_QM
    ```

2. Start the listener at the source for the receiver end of the channel from the target, using TCP/IP protocol for queue manager SRC_QM, port 1451, by issuing this command (if you used a port other than 1451, substitute the port number in the command):

    ```
    runmqlsr -t tcp -m SRC_QM -p 1451
    ```

    Tip: On Windows, you can also use **start runmqlsr** to open a new command window for the listener so that you can continue using your current command window. You can do the same thing on Linux and UNIX by adding a space and ampersand (&) to the end of the command.

3. Start an interactive session with the SRC_QM queue manager by issuing this command:

    ```
    runmqsc SRC_QM
    ```

4. Start the sender channel at the source by issuing this command:

    ```
    start channel (SRC_QM.TO.TGT_QM)
    ```

5. Use the **end** command to stop the interactive session with the source queue manager.
6. Start the listener at the target for the receiver end of the channel from the source, using TCP/IP protocol for queue manager TGT_QM, port 1450, by issuing this command (if you used a port other than 1450, substitute the port number in the command):

    ```
    runmqlsr -t tcp -m TGT_QM -p 1450
    ```

7. Start an interactive session with the TGT_QM queue manager by issuing this command:

```
runmqsc TGT_QM
```

8. Start the sender channel at the target by issuing this command:

```
start channel (TGT_QM.TO.SRC_QM)
```

9. Use the **end** command to stop the interactive session with the target queue manager.

Start replication between source and target

You are in the homestretch:

° Your channels are running. You can check their status by issuing the following commands from a **runmqsc** session with the SRC_QM queue manager or TGT_QM queue manager:

```
DISPLAY CHSTATUS (SRC_QM.TO.TGT_QM)
DISPLAY CHSTATUS (TGT_QM.TO.SRC_QM)
```

The status of the channels should be RUNNING or STARTING.

° Your Q subscription EMPLOYEE0001 maps the EMPLOYEE table to the newly created TGT_EMPLOYEE table.
° Your EMPLOYEE_QMAP defines a pair of queues that will be used to transport the data.

Now you can start the Q Capture and Q Apply programs to begin replicating data.

Start the Q Capture program

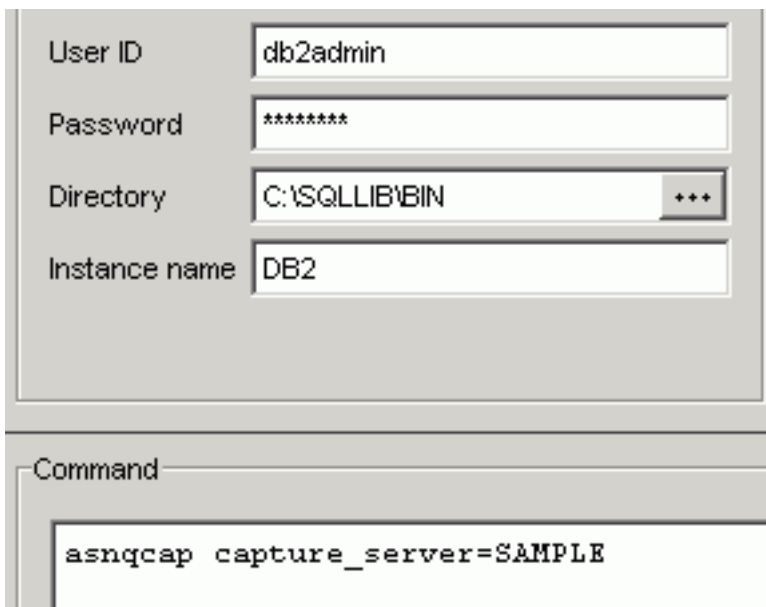Follow these steps to start the Q Capture program:

1. On the Replication Center Launchpad, click **Start a Q Capture program**.
2. Verify that the SAMPLE database is listed in the **Q Capture server** field, and ASN is listed in the **Q Capture schema** field, then click **OK**.
   Figure 17. Start Q Capture window
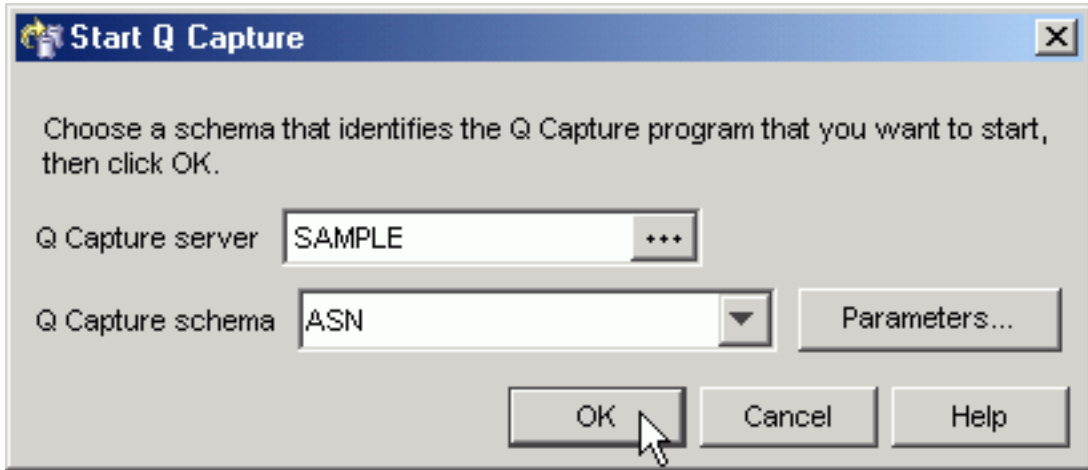
3. On the Run Now or Save Command window, enter a valid user ID and password for the SAMPLE database.

4.
   Click the [...] button next to the **Directory** field and browse to a directory where you will run the command to start Q Capture (this directory is also where the Q Capture log files will be sent by default). Click **OK**.
   Figure 18. Entering information on the Run Now or Save Command window

A DB2 Message window opens to confirm that the command was submitted.

Start the Q Apply program

Follow these steps to start the Q Apply program:

1.

    In the Replication Center object tree, use the ⊞ signs to open folders in the
    following order: **Q replication -> Operations**. Click the **Q Apply Servers**
    folder.

2. In the contents pane, right click the TARGET server, and click **Start Q Apply
   Program**.

3. Verify that TARGET is listed in the **Q Apply server** field, and ASN is listed in
   the **Q Apply schema** field. Click **OK**.

4. On the Run Now or Save Command window, enter a valid user ID and
   password for the TARGET database. Browse to a directory where you will
   run the command (and where the Q Apply log files will be sent by default).
   Click **OK**. A DB2 Message window opens to confirm that the command was
   submitted.

In the next section, you will check whether the two programs are successfully
replicating data.

See the results of replication

When you started the Q Capture and Q Apply programs, the following things
happened behind the scenes:

° Your EMPLOYEE0001 Q subscription was activated automatically, which is
  the default action.
° Because the Q subscription was active, the Q Capture program read the
  DB2 recovery log looking for changes to the EMPLOYEE table (there are
  none ... yet).
° The Q Apply program invoked a DB2 utility to copy the data in the
  EMPLOYEE table and another utility to load it into the new
  TGT_EMPLOYEE table at the TARGET database.

To verify that the loading process worked, you can issue the following SQL
statements from a DB2 command window:

```
db2 connect to target
select * from schema.tgt_employee
```

Where *schema* is the default name assigned to DB2 objects that are created by
your user ID.

You can also use the DB2 Control Center to view the newly loaded table:

**Figure 19. Control Center view of TGT_EMPLOYEE table loaded with
source data**

If the TARGET table has 32 rows with data similar to the figure above, you have successfully used Q replication to make a copy of the source table.

But the process of loading a target table takes place outside of WebSphere MQ. So to truly test replication over queues, you need to change the EMPLOYEE table and see the change replicate to the TGT_EMPLOYEE table.

Follow these steps:

1. From a DB2 command window, issue the following SQL statements:

```
db2 connect reset
db2 connect to SAMPLE
db2 insert into schema.employee values
('000350','EMILY','L','GIBSON','E21','4860','12/11/2001','DESIGNER',19,'F',
'02/02/1957',31500.00,750.00,2160.00)
```
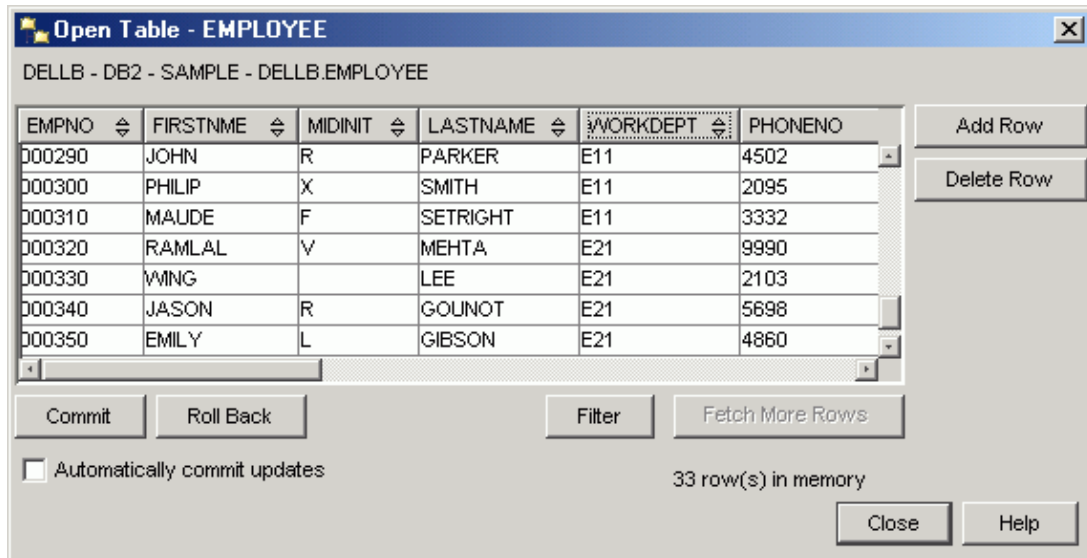
2. Issue the following SQL statement to view the new row:

```
db2 select * from schema.employee
```

You can also use the Control Center to view the new row.

**Figure 20. Control Center view of new row inserted into EMPLOYEE table (at bottom)**

3. By now, the new row has been replicated to the TGT_EMPLOYEE table.
    Issue the following SQL statements to view it:

```
db2 connect reset
db2 connect to target
db2 select * from schema.tgt_employee
```

You can also use the Control Center to view the new row.

Figure 21. Control Center view of new row inserted into EMPLOYEE table (at top)
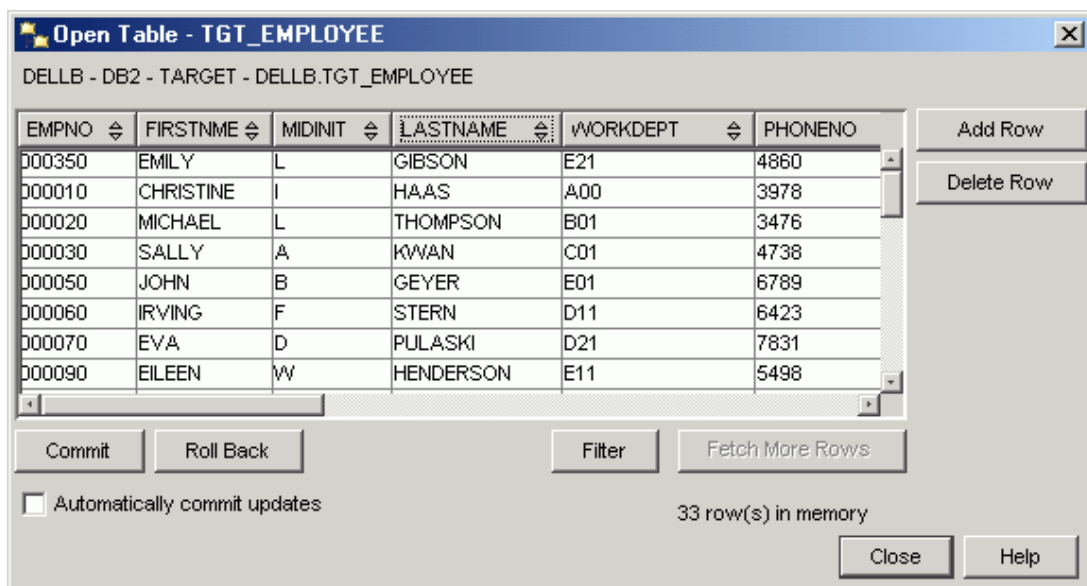
If you're seeing the row, congratulations! You have successfully used Q replication and WebSphere MQ to replicate data between two DB2 databases.

If not, try the following:

° Make sure the Q Capture and Q Apply programs are running. In the Replication Center object tree, open the **Q replication** and **Operations** folders and click the **Q Capture Servers** or **Q Apply Servers** folder. In the contents pane, right click on the SAMPLE server or TARGET server and

   click **Check Status**. Click the button and the program threads should

   appear, most likely in the "Resting" state. If the window says "Presumed down," start the Q Capture or Q Apply program.
° Make sure your queue managers are running by using the **strmqm SRC_QM** or **strmqm TGT_QM** commands. If so, use the **DISPLAY CHSTATUS (SRC_QM.TO.TGT_QM)** or **DISPLAY CHSTATUS (TGT_QM.TO.SRC_QM)** commands to check the status of your sender channels.

If you'd like to try a few updates or deletes, use the following syntax:

```
db2 connect reset
db2 connect to sample
db2 update schema.employee set column_name=value where column_name=value
```

Or:

```
db2 delete from schema.employee where column_name=value
```

Section 5. Summary and resources

Summary

In this tutorial, you learned how to use one of several types of Q replication: unidirectional between remote databases. This is a powerful tool for such business needs as placing data closer to geographically distributed applications or maintaining read-only copies of a production database.

With this tutorial complete, you're only a few steps away from setting up the more advanced configurations:

°   Bidirectional replication, in which changes are replicated in two directions between two databases (useful for managing hot standby or backup systems).
°   Peer-to-peer replication, which maintains convergence between two or more databases that are being updated in parallel by using version information that is added to tables.
°   Event publishing, which captures database changes, or *events*, and publishes them to applications in the form of XML messages.

You have a solid introduction to the Replication Center and are ready to learn more advanced functions that let you automatically monitor replication or publishing, filter data at the source, gather statistics for high performance tuning, or replicate stored procedures.

Good luck, happy replicating, and may your databases always stay in synch!

Resources

To learn more about Q replication and WebSphere MQ:

°   Get the full story about the *DB2 Information Integrator replication* solution.

°   Check out the *DB2 Information Integrator Introduction to Replication and Event Publishing* and *DB2 Information Integrator Replication and Event Publishing Guide and Reference* along with other technical publications on the *DB2 Information Integrator support page*.

°   Learn more about the *WebSphere MQ* family of business integration products.

°   Get the latest *WebSphere MQ books and manuals*.

Your feedback

Colophon

This tutorial was written entirely in XML, using the developerWorks Toot-O-Matic tutorial generator. The open source Toot-O-Matic tool is an XSLT stylesheet and several XSLT extension functions that convert an XML file into a number of HTML pages, a zip file, JPEG heading graphics, and two PDF files. Our ability to generate multiple text and binary formats from a single source file illustrates the power and flexibility of XML. (It also saves our production team a great deal of time and effort.)

For more information about the Toot-O-Matic, visit www-106.ibm.com/developerworks/xml/library/x-toot/ .