# Database Administration

# External Database Cleanup Utility

The External Database Cleanup Utility makes it easy for administrators to manage external database growth, thereby ensuring that your system continues to perform at the optimum level. The utility lets you create jobs that monitor the external database on an ongoing basis, deleting old records automatically as they expire. This ensures that the external database has adequate space and that system performance is not impacted by unchecked database growth.

The External Database Cleanup Utility can be used to manage external database growth for the following IM and Presence Service features, each of which relies on the external database:

- Persistent Chat High Availability

- Managed File Transfer

- Message Archiver

### Interactions

The following interactions apply:

- Records that are deleted from the database are deleted without archiving.

- You can run the Database Cleanup utility in offline mode.

- A persistent chat room configuration option is provided to override the cluster-wide setting for retention durations. This allows chat room owners to customize the settings within a controlled range. This is dependent on Jabber client changes to enable this menu option.

### Stored Procedures

The External Database Cleanup Utility uses the following stored procedures to delete records

- `tc_msgarchive_auto_cleanup`

- `tc_timelog_auto_cleanup`

- `aft_log_auto_cleanup`

- `im_auto_cleanup`

# Run the External Database Cleanup Utility

Use this procedure to run the External Database Cleanup Utility to delete expired records from the external database. You can run a manual cleanup to complete a one-time deletion of records from the database or you can configure the system to monitor and delete records from the external database automatically.

**Procedure**

**Step 1**   Log into Cisco Unified CM IM and Presence Administration on the database publisher node.

**Step 2**   Choose **Messaging** > **External Server Setup** > **External DataBase Jobs**.

**Step 3**   Click **Clear External DB**.

**Step 4**   Do one of the following:

- For manual cleanup of an external database that connects to the publisher node, select **SameCup Node**.
- For manual cleanup of an external database that connects to a subscriber node, select **Other CupNode** and then select the external database details.
- If you are configuring the system to monitor and clean the external database automatically, check the **Automatic Clean-up** radio button.

**Note**        We recommend that you run a manual cleanup prior to setting up the automatic cleanup.

**Step 5**   Set the **Number of Days** that you want to go back for file deletion. For example, if you enter 90, the system deletes records that are older than 90 days.

**Step 6**   Click **Update Schema** to create the Indexes and stored procedures for the database.

**Note**        You need to update the schema only the first time that you run the job.

**Step 7**   Set the **Number of Days** that you want to go back for file deletion. For example, if you enter **90**, the system deletes records that are older than 90 days.

**Step 8**   In the **Feature Tables** section, select each feature for which you want to clean records:

- **Text Conference (TC)**—Select this option to clean database tables for the Persistent Chat feature.
- **Message Archiver (MA)**—Select this option to clean database tables for the Message Archiver feature.
- **Managed File Transfer (MFT)**—Select this option to clean database tables for the Managed File Transfer feature

**Step 9**   Click **Submit Clean-up Job**.

**Note**        If you have the **Automatic** option enabled, and you want to disable it, click the **Disable Automatic Clean-up Job** button.

**What to do next**

If you just ran a manual cleanup, repeat the procedure and select **Automatic Clean-up** to set the utility to monitor and delete records automatically.

# Troubleshooting the External Database Cleanup Utility

If you run into issues with the external database utility:

- Make sure that the database publisher node can connect to the external database and that the database has been provisioned. You can verify this on the database publisher node by selecting **Messaging** > **External Server Setup** > **External Databases** > **External DB Configuration**.

- For PostgreSQL databases, make sure that the IM and Presence database publisher node has full access permission to all of the other configured external databases.

Logs for the External Database Cleanup Tool are available under admin logs: `/var/log/active/tomcat/logs/cupadmin/log4j/`

# Merge External Databases

Use this procedure to merge external databases.

✎

**Note**    Microsoft SQL database is not supported for merging external databases.

Optional. If you have upgraded from a release prior to 11.5(1), and multiple external databases were used to manage redundancy, use the External Database Merge Tool to merge your external databases into a single database.

**Example**

If you have upgraded from a release prior to 11.5(1), and you had persistent chat configured with each persistent chat node connecting to a separate external database instance, use this procedure to merge the two databases in a subcluster into a single database that connects to both nodes.

**Before you begin**

- Ensure that the two source destination databases are assigned correctly to each IM and Presence Service node in the presence redundancy group. This verifies that both of their schemas are valid.

- Back up the tablespace of the destination database.

- Ensure that there is enough space in the destination database for the new merged databases.

- Ensure that the database users, created for the source and destination databases, have the permissions to run these commands:

    - CREATE TABLE

    - CREATE PUBLIC DATABASE LINK

- If your database users do not have these permissions, you can use these commands to grant them:

- PostgreSQL:

  `CREATE EXTENTION`—This creates the dblink and requires super user or dbowner privileges. After this, you EXECUTE privilege for dblink by running following:

  `GRANT EXECUTE ON FUNCTION DBLINK_CONNECT(text) to <user>`

  `GRANT EXECUTE ON FUNCTION DBLINK_CONNECT(text,text) to <user>`

- Oracle:

  `GRANT CREATE TABLE TO <user_name>;`

  `GRANT CREATE PUBLIC DATABASE LINK TO <user_name>;`

- If you are using a PostgreSQL external database, make sure that the following access is configured in the pg_hba.conf file:

  - The IM and Presence publisher node must have full access to each external database.

  - The external PostgreSQL database must have full access to each database instance. For example, if the external database is configured on 192.168.10.1 then each database instance must be configured in the pg_hba.conf file as `host dbName username 192.168.10.0/24 password`.

**Procedure**

---

**Step 1**      Sign in to **Cisco Unified CM IM and Presence Administration** on the IM and Presence Service publisher node.

**Step 2**      Stop the Cisco XCP Text Conference Service on the **System** > **Services** window for each IM and Presence Service node in the presence redundancy group.

**Step 3**      Click **Messaging** > **External Server Setup** > **External Database Jobs**.

**Step 4**      Click **Find** if you want to see the list of merge jobs. Choose **Add Merge Job** to add a new job.

**Step 5**      On the **Merging External Databases** window, enter the following details:

- Choose **Oracle** or **Postgres** from the **Database Type** drop-down list.
- Choose the IP address and hostname of the two source databases and the destination database that will contain the merged data.

          If you chose Oracle as the **Database Type** enter the tablespace name and database name. If you chose Postgres as the **Database Type** you provide the database name.

**Step 6**      In the **Feature Tables** pane, the Text Conference(TC) check-box is checked by default. For the current release, the other options are not available.

**Step 7**      Click **Validate Selected Tables**.

          **Note**      If the Cisco XCP Text Conference service has not been stopped you receive an error message. Once the service has been stopped, validation will complete.

**Step 8**      If there are no errors in the **Validation Details** pane, click **Merge Selected Tables**.

**Step 9**      When merging has completed successfully, the **Find And List External Database Jobs** window is loaded. Click Find to refresh the window and view the new job.

          Click **Find** to refresh the window and view the new job.

Click the **ID** of the job if you want to view its details.

**Step 10** Restart the Cisco XCP Router service.

**Step 11** Start the Cisco XCP Text Conference Service on both IM and Presence Service nodes.

**Step 12** You must reassign the newly merged external database (destination database) to the presence redundancy group

# Migrate Persistent Chat Rooms from One External Database to Another

You can move all persistent chat rooms from an existing external database to a new database of the same type or different types without changing the IM and Presence nodes. This enables you to migrate all persistent chat rooms from one database to another, for example, from Oracle to Oracle, Oracle to MSSQL, MSSQL to PostgreSQL, and so on.

In this context, when a new database in connected to any of the IM and Presence services such as Persistent Chat, Message Archiver or Managed File Transfer, the service first checks whether the IM and Presence schema is present in the database. If the schema exists, it reuses the same. However, it creates a new schema only if the database does not have the required IM and Presence schema.

After the data migration, you can verify either through admin login into your application or through the backend of your newly configured external database by running the respective select statements for the tables.

**Note** There is no specific level of access required for database modification.

In the following procedures, migrating persistent chat rooms from Oracle to PostgreSQL has been considered and the tools used for import/export of data mentioned in this procedure are examples only. You can choose any tools for this purpose.

**Before you begin**

You need to set up and configure the new external database on the IM and Presence node. Refer to Set Up External Database Entry on IM and Presence Service for more details.

**Procedure**

**Step 1** Export the persistent chat rooms from the existing external database on the IM and Presence node.

**Step 2** Import the data to the new database.

**Step 3** Assign the configured external database entry on the respective IM and Presence Node.

**Step 4** Restart the following presence services, such as XCP Router, Text Conference Manager and Message Archiver.

**Step 5** To verify After restarting the services, log in to the Cisco Jabber and verify whether the chat rooms are available.

**Example Scenarios**

The following procedures are included to provide more clarity on how to migrate persistent chat rooms between various external databases. We have considered migrating persistent chat rooms in Cisco Jabber between various database types such as Oracle, PostgreSQL and MSSQL. The tools used for import/export of data mentioned in these procedures are examples only. However, you can choose any tools for this purpose.

# Migrate Persistent Chat Rooms from Oracle to PostgreSQL

The following procedure shows how to migrate persistent chat rooms created in Jabber which is currently connected to Oracle database to a newly created PostgreSQL database configured as an external database for the same IM and Presence node.

### Before you begin

- Persistent chat rooms are available in Jabber.

- You have configured the new external database, PostgreSQL in this case, in your environment.

- Ensure that you have installed a database migration tools in your machine such as, **Oracle SQL Developer** to export data from Oracle and **Table Plus** to import data into PostgreSQL in this case.

### Procedure

**Step 1**　Export the persistent chat rooms from your existing Oracle database. To do this:

- Open the Oracle SQL Developer tool and connect to your existing Oracle database by providing details such as database name, username, password, hostname, and port.

- After establishing the connection, run the following query to view the persistent chat rooms, which you have created in Jabber:

  ```
  select * from tc_rooms;
  ```

- From the object tree view, right click on the table that you want to export.

- Select CSV as the export data format.

- Browse the destination folder.

- Click **Next** and **Finish**.

This exports the selected table data in an Excel file.

**Step 2**   Import the persistent chat rooms to the new PostgreSQL database. To do this:

- Open the Table Plus tool and connect to the new PostgreSQL database by providing details such as database name, username, password, hostname, and port.

- Import the Excel file that was previously exported from the Oracle database.

You can view the imported table name in the tree structure.

**Step 3**   Assign the configured external database entry, that is PostgreSQL to the IM and Presence node. For more information on how to assign the external database on the IM and Presence node, see Set Up External Database Entry on IM and Presence Service.

**Step 4**   Restart the following services:

- XCP Router

- Text Conference Manager

- Message Archiver

Message Archiver configuration is not mandatory until and unless the persistent chat room users communicate their chat messages in the room.

**Step 5**   You can verify the data migration in either of the following ways:

- Log in to Jabber as an admin and verify if the chat rooms are present.

- Verify the migration in the target database by running select statements for the following tables:

  - Persistent chat tables - {tc_users, tc_rooms, tc_messages, tc_msgarchive and tc_timelog}

  - Message Archiver - {JM}

  - Managed File Transfer - {aft_log}

# Migrate Persistent Chat Rooms from Oracle to MSSQL

The following procedure shows how to migrate persistent chat rooms created in Jabber which is currently connected to Oracle database to a newly created MSSQL database configured as an external database for the same IM and Presence node.

**Before you begin**

- Persistent chat rooms are available in Jabber.

- You have configured the new external database, MSSQL in this case, in your environment.

- Ensure that you have installed a database migration tools in your machine such as, **Oracle SQL Developer** to export data from Oracle and **Microsoft SQL Server Management Studio** to import data into MSSQL in this case.

**Procedure**

**Step 1** Export the persistent chat rooms from your existing Oracle database. To do this:

- Open the Oracle SQL Developer tool and connect to your existing Oracle database by providing details such as database name, username, password, hostname, and port.

- After establishing the connection, run the following query to view the persistent chat rooms, which you have created in Jabber:

  ```
  select * from tc_rooms;
  ```

- From the object tree view, right click on the table that you want to export.

- Select CSV as the export data format.

- Browse the destination folder.

- Click **Next** and **Finish**.

This exports the selected table data in an Excel file.

**Step 2** Import the persistent chat rooms to the new PostgreSQL database. To do this:

- Open the Microsoft SQL server Management Studio tool and connect to the new MSSQL database by providing details such as database name, username, password, hostname, and port.

- Import the Excel file that was previously exported from the Oracle database.

You can view the imported table name in the tree structure.

**Step 3** Assign the configured external database entry, that is MSSQL to the IM and Presence node. For more information on how to assign the external database on the IM and Presence node, see Set Up External Database Entry on IM and Presence Service.

**Step 4** Restart the following services:

- XCP Router

- Text Conference Manager

- Message Archiver

Message Archiver configuration is not mandatory until and unless the persistent chat room users communicate their chat messages in the room.

**Step 5** You can verify the data migration in either of the following ways:

- Log in to Jabber as an admin and verify if the chat rooms are present.

- Verify the migration in the target database by running select statements for the following tables:

  - Persistent chat tables - {tc_users, tc_rooms, tc_messages, tc_msgarchive and tc_timelog}

  - Message Archiver - {JM}

• Managed File Transfer - {aft_log}

# Migrate Persistent Chat Rooms between Two Oracle Databases

The following procedure shows how to migrate persistent chat rooms created in Jabber which is currently connected to Oracle database to a newly created instance of Oracle database configured as an external database for the same IM and Presence node.

**Before you begin**

- Persistent chat rooms are available in Jabber.

- You have configured the new external database, Oracle in this case, in your environment.

- Ensure that you have installed a database migration tools in your machine such as, **Oracle SQL Developer** to export and import data from Oracle.

**Procedure**

**Step 1** Create a new external database.

**Step 2** Stop Text Conference Manager service on all the nodes.

**Step 3** Assign the configured external database entry, that is Oracle to the IM and Presence node. For more information on how to assign the external database on the IM and Presence node, see Set Up External Database Entry on IM and Presence Service.

**Step 4** Check the IM and Presence troubleshooting page and ensure the database schema verification is successful.

**Step 5** Check the databases, ensure the tables, indexes, and stored procedures have been created.

If something is not created, run the `postDBTool_oracle.sql` script directly on the database.

The script can be found on IM and Presence server under `/usr/local/xcp/schemas/sql/postDBTool_oracle.sql`.

**Step 6** Export the persistent chat rooms from your existing Oracle database. To do this:

- Open the Oracle SQL Developer tool and connect to your existing Oracle database by providing details such as database name, username, password, hostname, and port.

- After establishing the connection, run the following query to view the persistent chat rooms, which you have created in Jabber:

  `select * from tc_rooms;`

- From the object tree view, right click on the table that you want to export.

- Select CSV as the export data format.

- Browse the destination folder.

- Click **Next** and **Finish**.

This exports the selected table data in an Excel file.

**Step 7** Import the persistent chat rooms to the new Oracle database. To do this:

- Open the Oracle SQL Developer tool and connect to the new Oracle database by providing details such as database name, username, password, hostname, and port.

- Import the Excel file that was previously exported from the Oracle database.

You can view the imported table name in the tree structure.

**Step 8** Restart the following services:

- XCP Router

- Text Conference Manager

- Message Archiver

Message Archiver configuration is not mandatory until and unless the persistent chat room users communicate their chat messages in the room.

**Step 9** You can verify the data migration in either of the following ways:

- Log in to Jabber as an admin and verify if the chat rooms are present.

- Verify the migration in the target database by running select statements for the following tables:

  - Persistent chat tables - {tc_users, tc_rooms, tc_messages, tc_msgarchive and tc_timelog}

  - Message Archiver - {JM}

  - Managed File Transfer - {aft_log}

# Migrate Persistent Chat Rooms from MSSQL to PostgreSQL

The following procedure shows how to migrate persistent chat rooms created in Jabber which is currently connected to MSSQL database to a newly created PostgreSQL database configured as an external database for the same IM and Presence node.

**Before you begin**

- Persistent chat rooms are available in Jabber.

- You have configured the new external database, PostgreSQL in this case, in your environment.

- Ensure that you have installed a database migration tool in your machine such as, **Oracle SQL Developer** to export data from MSSQL and **Microsoft SQL Server Management Studio** to import data into PostgreSQL in this case.

**Procedure**

**Step 1**   Export the persistent chat rooms from your existing MSSQL database. To do this:

- Open the Oracle SQL Developer tool and connect to your existing MSSQL database by providing details such as database name, username, password, hostname, and port.

- After establishing the connection, run the following query to view the persistent chat rooms, which you have created in Jabber:

  ```
  select * from tc_rooms;
  ```

- From the object tree view, right click on the table that you want to export.

- Select CSV as the export data format.

- Browse the destination folder.

- Click **Next** and **Finish**.

This exports the selected table data in an Excel file.

**Step 2**   Import the persistent chat rooms to the new PostgreSQL database. To do this:

- Open the Microsoft SQL server Management Studio tool and connect to the new PostgreSQL database by providing details such as database name, username, password, hostname, and port.

- Import the Excel file that was previously exported from the MSSQL database.

You can view the imported table name in the tree structure.

**Step 3**   Assign the configured external database entry, that is PostgreSQL to the IM and Presence node. For more information on how to assign the external database on the IM and Presence node, see Set Up External Database Entry on IM and Presence Service.

**Step 4**   Restart the following services:

- XCP Router

- Text Conference Manager

- Message Archiver

Message Archiver configuration is not mandatory until and unless the persistent chat room users communicate their chat messages in the room.

**Step 5**   You can verify the data migration in either of the following ways:

- Log in to Jabber as an admin and verify if the chat rooms are present.

- Verify the migration in the target database by running select statements for the following tables:

  - Persistent chat tables - {tc_users, tc_rooms, tc_messages, tc_msgarchive and tc_timelog}

  - Message Archiver - {JM}

• Managed File Transfer - {aft_log}

# Migrate Persistent Chat Rooms from MSSQL to Oracle

The following procedure shows how to migrate persistent chat rooms created in Jabber which is currently connected to MSSQL database to a newly created Oracle database configured as an external database for the same IM and Presence node.

### Before you begin

- Persistent chat rooms are available in Jabber.

- You have configured the new external database, Oracle in this case, in your environment.

- Ensure that you have installed a database migration tools in your machine such as, **Microsoft SQL Server Management Studio** to export data from Oracle and **Oracle SQL Developer** to import data into PostgreSQL in this case.

### Procedure

---

**Step 1** Export the persistent chat rooms from your existing MSSQL database. To do this:

- Open the Microsoft SQL Server Management Studio tool and connect to your existing MSSQL database by providing details such as database name, username, password, hostname, and port.

- After establishing the connection, run the following query to view the persistent chat rooms, which you have created in Jabber:

  ```
  select * from tc_rooms;
  ```

- From the object tree view, right click on the table that you want to export.

- Select CSV as the export data format.

- Browse the destination folder.

- Click **Next** and **Finish**.

This exports the selected table data in an Excel file.

**Step 2** Import the persistent chat rooms to the new Oracle database. To do this:

- Open the Oracle SQL Developer tool and connect to the new Oracle database by providing details such as database name, username, password, hostname, and port.

- Import the Excel file that was previously exported from the MSSQL database.

You can view the imported table name in the tree structure.

**Step 3**   Assign the configured external database entry, that is Oracle to the IM and Presence node. For more information on how to assign the external database on the IM and Presence node, see Set Up External Database Entry on IM and Presence Service.

**Step 4**   Restart the following services:

- XCP Router

- Text Conference Manager

- Message Archiver

Message Archiver configuration is not mandatory until and unless the persistent chat room users communicate their chat messages in the room.

**Step 5**   You can verify the data migration in either of the following ways:

- Log in to Jabber as an admin and verify if the chat rooms are present.

- Verify the migration in the target database by running select statements for the following tables:

  - Persistent chat tables - {tc_users, tc_rooms, tc_messages, tc_msgarchive and tc_timelog}

  - Message Archiver - {JM}

  - Managed File Transfer - {aft_log}

# Migrate Persistent Chat Rooms between Two MSSQL Databases

The following procedure shows how to migrate persistent chat rooms created in Jabber which is currently connected to MSSQL database to a newly created instance of MSSQL database configured as an external database for the same IM and Presence node.

**Before you begin**

- Persistent chat rooms are available in Jabber.

- You have configured the new external database, MSSQL in this case, in your environment.

- Ensure that you have installed a database migration tools in your machine such as, **Microsoft SQL Server Management Studio** to export and import data from MSSQL.

**Procedure**

**Step 1**   Export the persistent chat rooms from your existing MSSQL database. To do this:

- Open the Microsoft SQL Server Management Studio tool and connect to your existing MSSQL database by providing details such as database name, username, password, hostname, and port.

- After establishing the connection, run the following query to view the persistent chat rooms, which you have created in Jabber:

```
select * from tc_rooms;
```

- From the object tree view, right click on the table that you want to export.

- Select CSV as the export data format.

- Browse the destination folder.

- Click **Next** and **Finish**.

This exports the selected table data in an Excel file.

**Step 2**  Import the persistent chat rooms to the new MSSQL database. To do this:

- Open the Microsoft SQL Server Management Studio tool and connect to the new MSSQL database by providing details such as database name, username, password, hostname, and port.

- Import the Excel file that was previously exported from the MSSQL database.

You can view the imported table name in the tree structure.

**Step 3**  Assign the configured external database entry, that is MSSQL to the IM and Presence node. For more information on how to assign the external database on the IM and Presence node, see Set Up External Database Entry on IM and Presence Service.

**Step 4**  Restart the following services:

- XCP Router

- Text Conference Manager

- Message Archiver

Message Archiver configuration is not mandatory until and unless the persistent chat room users communicate their chat messages in the room.

**Step 5**  You can verify the data migration in either of the following ways:

- Log in to Jabber as an admin and verify if the chat rooms are present.

- Verify the migration in the target database by running select statements for the following tables:

    - Persistent chat tables - {tc_users, tc_rooms, tc_messages, tc_msgarchive and tc_timelog}

    - Message Archiver - {JM}

    - Managed File Transfer - {aft_log}