



Configure High Availability for Persistent Chat

- [High Availability for Persistent Chat Overview, on page 1](#)
- [High Availability for Persistent Chat Prerequisites, on page 3](#)
- [High Availability for Persistent Chat Task Flow, on page 4](#)
- [High Availability for Persistent Chat Use Cases, on page 8](#)

High Availability for Persistent Chat Overview

High Availability (HA) for Persistent Chat is an optional feature that you can deploy if you are using Persistent Chat rooms and you have system redundancy configured with Presence Redundancy Groups.

High Availability for Persistent Chat adds redundancy and failover capability to your persistent chat rooms. In the event of an IM and Presence Service node failure or Text Conferencing (TC) service failure, all persistent chat rooms hosted by that service are automatically hosted by the backup node or TC service. After failover, Cisco Jabber clients can seamlessly continue to use the persistent chat rooms.

External Database

The main difference between the Persistent Chat (non-HA) and Persistent Chat HA setup is around the external database requirements:

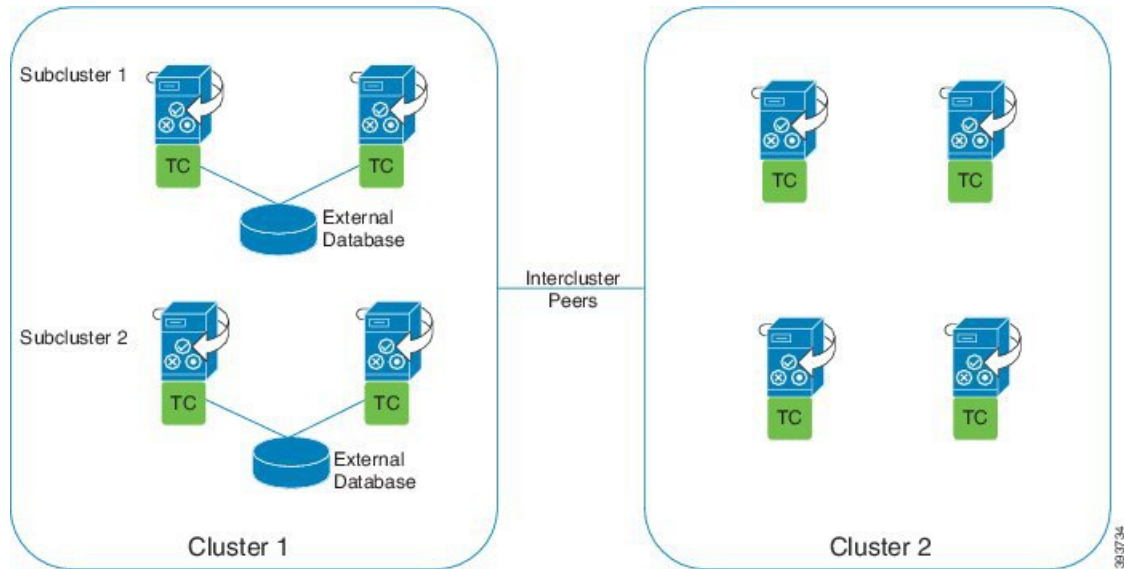
- If Persistent Chat is deployed without HA, the external database connects to an individual chat node only. Each node that hosts persistent chat rooms requires a separate external database instance. If a chat node fails, persistent chat rooms that were hosted on that node become unavailable until the chat node comes back up.
- If High Availability for Persistent Chat is deployed, the external database instance connects to both nodes in a subcluster (Presence Redundancy Group). If a persistent chat node fails, the backup node in the subcluster takes over, allowing chat to continue uninterrupted.

High Availability for Persistent Chat - Intercluster Example

The following illustration displays an intercluster network where Persistent Chat High Availability is deployed in Cluster 1 only. With Persistent Chat High Availability, each subcluster hosts an external database. Cluster 2 does not have Persistent Chat High Availability enabled, so there is no external database requirement. However, because the Cisco Text Conference Manager service is running on all nodes, users in Cluster 2 can join persistent chat rooms that are hosted in Cluster 1.



Note In this example, only the chat rooms in Cluster 1 are configured to host persistent chat rooms. You can also add persistent chat support on the Cluster 2 nodes, along with external database instances. In this case, all users in either cluster would be able to join persistent chat rooms that are hosted on any node in either cluster.



Comparison of Persistent Chat (non-HA) and Persistent Chat HA Requirements

If you are deploying Persistent Chat Rooms, Cisco recommends that you deploy High Availability for Persistent Chat as well as this adds failover capability to your persistent chat rooms. However, it is not mandatory.

The following table discusses the differences between Persistent Chat deployed with and without High Availability.

Table 1: Comparison of Persistent Chat with and without High Availability

	Persistent Chat (without HA)	Persistent Chat HA
Database Requirements	<p>You require a separate external database instance for each cluster node that hosts persistent chat rooms. These external database instances can be created on the same external database server.</p> <p>Recommended: For optimum performance and scalability, deploy a unique logical external database instance for each node or redundancy group in the IM and Presence cluster. However, this is not mandatory.</p> <p>Minimum Requirement: You must have at least one external database instance for Persistent Chat across an IM and Presence intercluster network. However, this deployment may be inadequate for high-use networks.</p> <p>Supported Database Types</p> <ul style="list-style-type: none"> • PostgreSQL (version 9.1 and above) • Oracle • Microsoft SQL Server 	<p>You require a separate external database instance for each subcluster (Presence Redundancy Group) that hosts persistent chat rooms. These external database instances can be created on the same external database server.</p> <p>Recommended: For optimum performance and scalability, deploy a separate external database instance for each subcluster within an IM and Presence cluster. However, this is not mandatory.</p> <p>Minimum Requirement: You require at least one external database instance for Persistent Chat HA across an IM and Presence intercluster network. However, this deployment may be inadequate for high-use networks.</p> <p>Supported Database Types</p> <ul style="list-style-type: none"> • PostgreSQL (version 9.1 and above) • Oracle • Microsoft SQL Server (as of 11.5(1)SU2)
Behavior when persistent chat node fails	<ul style="list-style-type: none"> • Persistent chat rooms hosted on the failed node are inaccessible until the node comes back up. • Users homed on the failed node fail over to the backup node in the subcluster, provided cluster redundancy is configured. However, they cannot access persistent chat rooms from the failed node. 	<ul style="list-style-type: none"> • Persistent chat rooms failover to the backup node in the subcluster. Users can continue messaging with no interruption of services. • Any users homed on the failed node also fail over.

High Availability for Persistent Chat Prerequisites

Before you configure High Availability for Persistent Chat, make sure that:

- Persistent Chat rooms are enabled. For details, see [Configure Chat Room Settings](#).
- High availability is enabled in each Presence Redundancy Groups. For details, see [Presence Redundancy Group Task Flow](#).

- You have configured the external database. For database setup and support information, see the *Database Setup Guide for the IM and Presence Service*.

High Availability for Persistent Chat Task Flow

Procedure

	Command or Action	Purpose
Step 1	Set up External Database, on page 4	You require a separate external database instance for each subcluster where persistent chat rooms are hosted. These separate external database instances can be hosted on the same database server
Step 2	Add External Database Connection, on page 5	Configure a connection to the external database from the IM and Presence Service.
Step 3	Verify High Availability for Persistent Chat Settings, on page 5	Confirm your system settings for Persistent Chat High Availability.
Step 4	Start Cisco XCP Text Conference Manager Service, on page 6	If the Cisco XCP Text Conference Manager service was stopped on any nodes, use this procedure to start it.
Step 5	Merge External Databases, on page 6	Optional. If you are upgrading from an earlier release where you had Persistent Chat configured with multiple external databases, use this procedure to merge your external databases into a single database.

Set up External Database

To deploy High Availability for Persistent Chat, you require a separate external database instance for each subcluster where persistent chat rooms are hosted. These separate external database instances can be hosted on the same database server.

A subcluster is a redundant pair of IM and Presence nodes (Presence Redudancy Group). You can have a maximum of three subclusters in an IM and Presence cluster of 6 nodes. If HA for Persistent Chat is enabled in an IM and Presence cluster of 6 nodes, you will have three external database instances and three subcluster pairs.

You can use PostgreSQL, Oracle, or Microsoft SQL Server for the external database connection. For setup details, refer to the *Database Setup Guide for IM and Presence Service*.

What to do next

[Add External Database Connection, on page 5](#)

Add External Database Connection

Configure connections to the High Availability for Persistent Chat external database instances from the IM and Presence Service. Make sure that both nodes in the subcluster are assigned to the same unique logical external database instance.

Procedure

-
- Step 1** From Cisco Unified CM IM and Presence Administration, choose **Messaging > External Servers Setup > External Databases**.
 - Step 2** Click **Add New**.
 - Step 3** In the **Database Name** field, enter the name of external database instance.
 - Step 4** From the **Database Type** drop-down, select the type of external database that you are deploying.
 - Step 5** Enter the **User Name** and **Password information** for the database.
 - Step 6** In the **Hostname** field, enter the hostname or IP address of the database.
 - Step 7** Complete the remaining settings in the **External Database Settings** window. For help with the fields and their settings, refer to the online help.
 - Step 8** Click **Save**.
 - Step 9** Repeat this procedure to create connections to each external database instance.
-

What to do next

[Verify High Availability for Persistent Chat Settings, on page 5](#)

Verify High Availability for Persistent Chat Settings

Use this procedure to confirm that your system is set up for High Availability for Persistent Chat.



Note If you've already enabled High Availability for your Presence Redundancy Groups (subclusters) and your chat room configuration includes Persistent Chat then your High Availability for Persistent Chat may be completed.

Procedure

-
- Step 1** Confirm that High Availability is enabled in each subcluster:
 - a) From Cisco Unified CM Administration, choose **System > Presence Redundancy Groups**.
 - b) Click **Find** and choose the Presence Redundancy Group that you want to check.
 - c) Verify that the **Enable High Availability** check box is checked. If the check box is unchecked, then check it.
 - d) Click **Save**.
 - e) Repeat these steps for each presence redundancy group in the cluster.

- Step 2** Confirm that persistent chat is enabled:
- From Cisco Unified CM Administration, choose **Messaging > Group Chat and Persistent Chat**.
 - Confirm that the **Enable Persistent Chat** check box is checked. If the check box is unchecked, then check it.
 - Click **Save**.
- Step 3** From Cisco Unified CM Administration, confirm that the **Cisco XCP Text Conference Manager Service** is running on all cluster nodes.
- Choose **System > Presence Topology**.
 - For each cluster node, click **view** to view the node details
 - Under **Node Status**, verify that the **Cisco XCP Text Conference Manager** service is **STARTED**.
 - In the left navigation bar, click **Presence Topology** to return to the cluster topology and and repeat the above steps until you've confirmed the status for all cluster nodes.

What to do next

If the **Cisco XCP Text Conference Manager Service** service needs to be enabled, [Start Cisco XCP Text Conference Manager Service, on page 6](#).

Start Cisco XCP Text Conference Manager Service

Use this procedure to start the Cisco XCP Text Conference Manager service. This service must be running on all cluster nodes for users on those nodes to be able to join persistent chat rooms.

Procedure

- Step 1** In **Cisco Unified IM and Presence Serviceability**, choose **Tools > Control Center - Feature Services**.
- Step 2** From the **Server** drop-down list, choose the IM and Presence cluster node and click **Go**.
- Step 3** Under **IM and Presence Services**, select **Cisco XCP Text Conference Manager** and click **Start**.
- Step 4** Click **OK**.
- Step 5** (Optional) Click **Refresh** if you want to verify that the service has fully restarted.

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 EXTENSION—`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

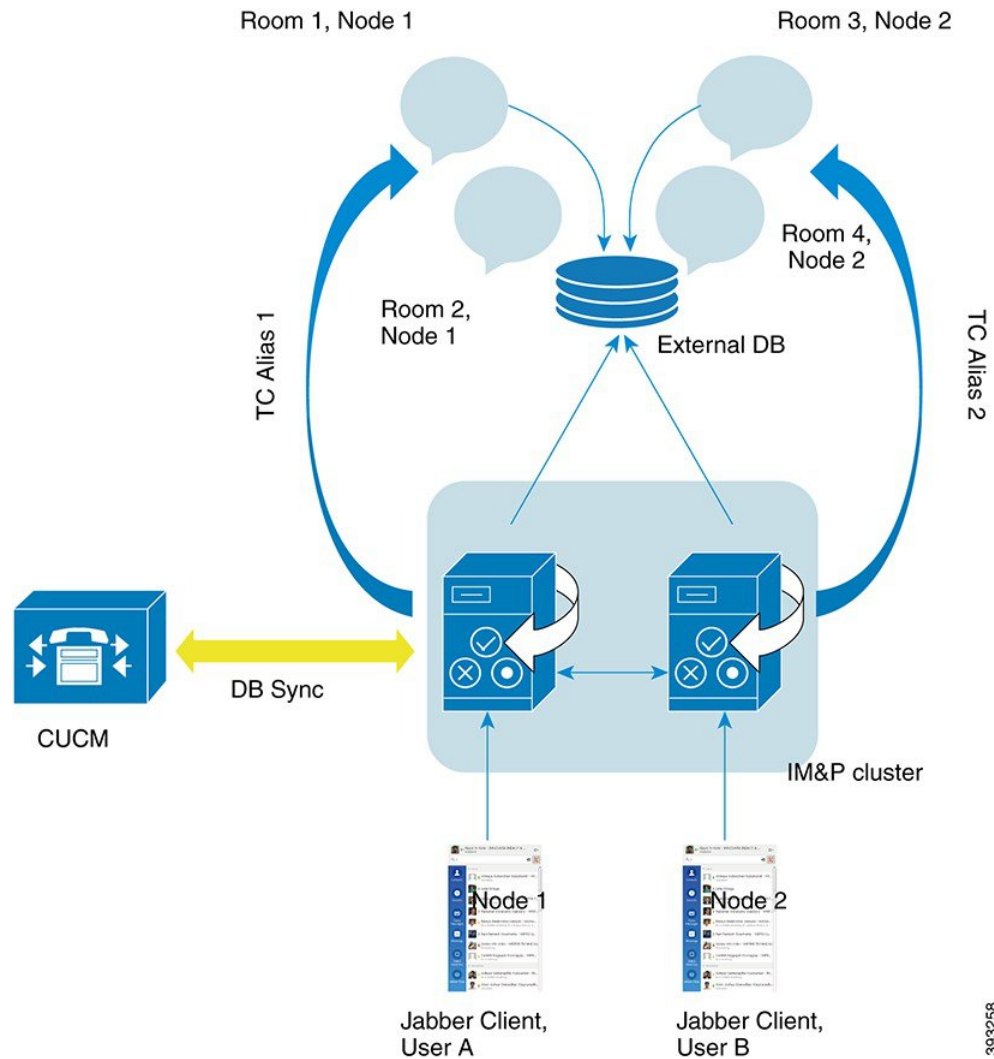
High Availability for Persistent Chat Use Cases

The following flows demonstrate the high availability for persistent chat flows for failover and failback. This example covers an IM and Presence cluster with two nodes. An IM and Presence cluster can have a maximum of 6 nodes, which allows for three subclusters. If persistent chat rooms are hosted on all nodes, you require three separate external database instances.



Note For this enhancement the Text Conferencing (TC) service has been made a critical service. As a result, the TC high availability failover flow remains the same even if the failover has been caused by the failure of another critical service on the node, such as the Cisco XCP Router service.

Figure 1: High Availability for Persistent Chat Structure



High Availability for Persistent Chat Failover Use Case

For this example, there are four users on four IM and Presence Service nodes with two High Availability (HA) pairs or subclusters. The users are assigned as follows:

Subcluster 1	Subcluster 2
<ul style="list-style-type: none"> • Andy is on Node 1A—Node 1A hosts the chat room • Bob is on Node 1B 	<ul style="list-style-type: none"> • Catherine is on Node 2A • Deborah is on Node 2B

1. All four users are chatting in the same chat room, which is hosted on Node 1A.
2. The Text Conferencing (TC) service fails on Node 1A.

3. After 90 seconds, the Server Recovery Manager (SRM) determines the failure of the TC critical service and starts an automatic failover.
4. Node 1B takes over the users from 1A and transitions to the **Failed Over with Critical Services not Running** state, before transitioning to the HA state **Running in Backup Mode**.
5. In line with the HA Failover Model, Andy is signed out from node 1A automatically and is signed in to the backup Node 1B.
6. The other users are not affected, but continue to post messages to the chat room, which is now hosted on Node 1B.
7. Andy enters the persistent chat room, and continues to read or post messages to the room.

High Availability Persistent Chat Fallback Use Case

For this example there are four users on four IM and Presence Service nodes with two High Availability (HA) pairs or subclusters. The users are assigned as follows:

Subcluster 1	Subcluster 2
<ul style="list-style-type: none"> • Andy is on Node 1A—Node 1A hosts the chat room • Bob is on Node 1B 	<ul style="list-style-type: none"> • Catherine is on Node 2A • Deborah is on Node 2B

1. All four users are chatting in the same chat room, which is hosted on Node 1A.
2. The Text Conferencing (TC) service fails on Node 1A.
3. Node 1B takes over the users from 1A and transitions to the **Failed Over with Critical Services not Running**, before transitioning to the HA state **Running in Backup Mode**.
4. In line with the HA Failover model, Andy is signed out automatically and is signed in to the backup Node 1B.
5. Bob, Catherine and Deborah are unaffected, but continue to post messages to the chat room, which is now hosted on Node 1B.
6. The IM and Presence Service administrator starts a manual fallback.
7. Node 1A transitions to **Taking Back** and Node 1B transitions to **Falling Back**.
8. Andy is signed out of Node 1B. Bob, Catherine, and Deborah continue to use the persistent chat room, and once **Fallback** has occurred, the room is moved back to Node 1A.
9. Node 1B moves from the HA state **Falling Back** to **Normal** and unloads its peer node rooms.
10. Node 1A moves from the HA state **Taking Back** to **Normal** and it reloads the chat room.
11. Andy enters the persistent chat room, and continues to read or post messages to the room.