Administration Guide | PUBLIC
SAP IQ 16.2
Document Version: 1.0.0 – 2026-01-08

SAP IQ Administration: Multiplex

Content

1 SAP IQ Administration: Multiplex

SAP IQ multiplex is a highly scalable, shared-disk grid technology that allows concurrent data loads and queries via independent data-processing nodes connected to a shared data source.

2 Multiplex Overview

IQ multiplex technology provides performance, scalability, and high availability benefits.

Performance	IQ multiplex uses the standard IQ server engine, providing proven query and load performance.
Low total cost of ownership	IQ multiplex uses standard, low cost hardware and operating systems.
Easy, flexible, setup and configuration	IQ multiplex can be configured easily using an SQL interface or through the SAP IQ Cockpit GUI.
Scalability	IQ multiplex can scale to large number of nodes to support tens of thousands of users, with little or no data load and query performance degradation as nodes are added to the multiplex.
High availability	Failure of any node leaves query jobs unaffected on other nodes.

IQ multiplex provides an ideal platform for enterprise-wide deployment of critical applications.

In this section:

2.1 SAP IQ Multiplex Architecture

SAP IQ multiplexes have a hybrid cluster architecture that involves both shared and local storage.

Shared storage is used for permanent IQ data and shared temporary data for distributed query processing. Local storage is used for catalog metadata, temporary data, and transaction logs.

Each server has its own temporary store and catalog store.

The shared IQ store and shared temporary store are common to all servers.

Data managed by SAP SQL Anywhere is not shared. Each node requires a separate copy of such data. *Introduction to SAP IQ* describes the distinction between data managed solely by and data that is managed by underlying SAP SQL Anywhere software.

Managed by SAP IQ	Managed by SAP SQL Anywhere
IQ_SYSTEM_MAIN	System catalog
IQ_SYSTEM_MSG	SA temporary dbspace
IQ_SYSTEM_TEMP	SA catalog dbspaces
IQ_SHARED_TEMP	
IQ user main dbspaces	

IQ_SYSTEM_MAIN, IQ_SHARED_TEMP, and IQ user main dbspaces are shared by all multiplex servers, and all servers need access to the same physical file.

Each server has its own set of files in IQ_SYSTEM_TEMP and IQ_SYSTEM_MSG.

> **Note**
>
> If you have a small test multiplex with servers on the same physical host, follow this guideline:
>
> - To address dbspaces managed by IQ, use absolute — not relative — paths.
> - To address catalog dbspaces managed by SA, use relative paths — not absolute — paths.

In this section:

2.1.1 Multiplex Architecture Diagram

SAP IQ multiplex nodes may have different roles with different capabilities, unlike other database cluster architectures, which usually follow either a “shared everything” or “shared nothing” architecture.

The configuration can be described as an "asymmetrical cluster."

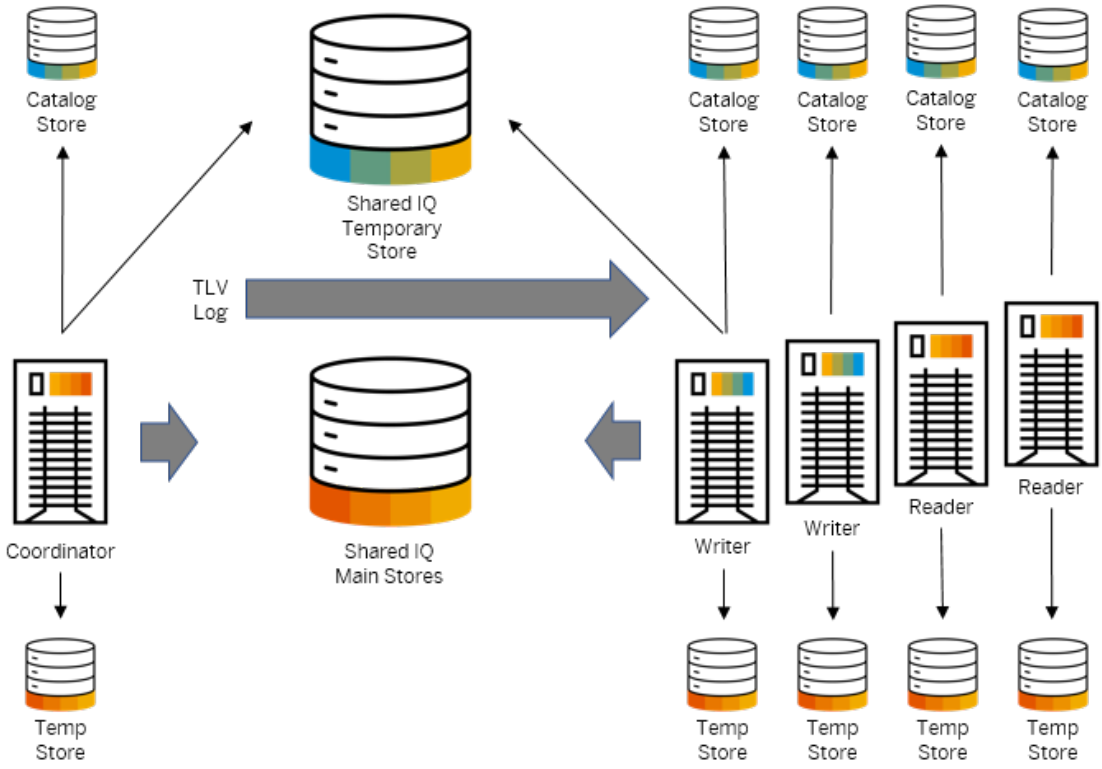

Figure 1: IQ Multiplex Architecture

Parent topic: SAP IQ Multiplex Architecture [page 6]

Related Information

Multiplex Configuration [page 8]
Shared Store Path Requirement [page 10]

2.1.2 Multiplex Configuration

SAP IQ Administration: Database multiplex configuration requires static and dynamic components.

- Static configuration – includes multiplex member nodes and their properties.
- Dynamic configuration – includes runtime process and connection status and table version usage.

SAP IQ Administration: Database stores IQ table metadata in the table's dbspace and the table version (TLV) log, which is a shared entity on the main store. The multiplex synchronizes stored table versions when an unavailable node comes back up.

Coordinator Node

Each multiplex configuration has one and only one coordinator node that is capable of read-only and read-write operations and is also responsible for bookkeeping to manage other nodes.

When you convert an existing SAP IQ Administration: Database server to multiplex, it becomes the coordinator node.

- Runs read-only and read-write operations against shared IQ objects.
- Manages IQ main dbspaces.
- Manipulates local data in SQL Anywhere system and user tables.
- Coordinates all read-write operations on shared IQ objects, including:
  - Shared IQ table locking
  - Shared IQ storage management
  - Providing global transaction IDs for read-write transactions involving shared IQ objects
  - Maintaining the global catalog
- Controls catalog synchronization for secondary servers
- Performs schema changes on shared IQ store objects
- Performs schema changes on SQL Anywhere store objects
- Maintains and cleans up object versions

Secondary Node

All nodes in a multiplex configuration are secondary nodes except the coordinator.

One secondary node acts as a designated failover node, the first choice node to assume the coordinator role if the current coordinator is unable to continue.

The number of secondary nodes supported depends on the license purchased, as follows:

- Demo/Trial Edition: Unlimited secondary nodes
- Enterprise Edition: Unlimited secondary nodes (license needed for each)

There are two types of secondary nodes:

- Writer nodes
  - Can run read-only and read-write operations against shared IQ objects.
  - Can manipulate local data in temporary and SA base tables.
- Reader nodes
  - Can run read-only operations against shared IQ objects.
  - Can manipulate local data in temporary and SA base tables.

Parent topic: SAP IQ Multiplex Architecture [page 6]

Related Information

Multiplex Architecture Diagram [page 7]
Shared Store Path Requirement [page 10]

2.1.3 Shared Store Path Requirement

Use the same paths from all nodes to access files in shared disk arrays.

All files in all dbspaces in the shared store, namely all files in IQ_SYSTEM_MAIN and shared dbspaces, must be accessible in the same way from all nodes. Files in the shared temporary store should be accessible from all nodes in the multiplex.

For IQ_SYSTEM_TEMP and IQ_SYSTEM_MSG, independent files are maintained for each node. The paths for these files must be accessible on the node that owns them.

UNIX

Use either method to access shared disks:

- Absolute paths to shared stores. For example:

```
/dev/rdsk/c4t600A0B80005A7F5D0000024B49757E55d0s0
/dev/rdsk/c4t600A0B80005A7F5D0000024B49757E55d0s1
/dev/rdsk/c4t600A0B80005A7F5D0000024B49757E55d0s2
```

- Soft links to shared store, where each soft link points to a raw device, as shown in the absolute path example. For example:

```
store/mainstore/userdb1store/userdb2
```

> **Note**
>
> Raw devices aren't available in Linux. See SAP Note 3483103 The RAW_DRIVER, and support for devices located in `/dev/raw/`, was deprecated in:
>
> - SuSE Linux Enterprise Server (SLES) 15.4
> - Red Hat Enterprise Linux 9

Windows

Use either method to access shared disks:

- The Disk Physical number. For example:

```
\\\\.\\PhysicalDrive1
```

```
\\\\.\\PhysicalDrive2
\\\\.\\PhysicalDrive3
```

- Absolute paths using drive letters. For example:

  ```
  x:\main
  y:\userdb1
  z:\userdb2
  ```

  Using the drive letters limits the number of disks to fewer than 26, so the first method is better.

Parent topic: SAP IQ Multiplex Architecture [page 6]

Related Information

Multiplex Architecture Diagram [page 7]
Multiplex Configuration [page 8]

2.2 Communication Infrastructure

The servers that participate in the multiplex share messages and data using two frameworks.

In this section:

2.2.1 Internode Communication (INC)

INC provides transactional communication between coordinator and secondary nodes.

INC links exist between the coordinator to the secondary nodes and from secondary nodes to the coordinator, but not between secondary nodes.

Secondary servers communicate with the coordinator via INC for executing DDL and read-write DML statements. The coordinator communicates with the secondary servers via internode communication when certain stored procedures are executed on the coordinator. Secondary servers never communicate with each other using INC.

The INC link consists of heartbeat and pooled connections.

Heartbeat Connections

Every secondary node has a heartbeat connection, a dedicated connection to the coordinator node. This connection is established during secondary node startup and remains active if the secondary node is active.

Both the coordinator and secondary node monitor the heartbeat connection. If this connection is broken, the node is declared offline. If the heartbeat is not reestablished within a given period, the coordinator can automatically exclude the secondary server depending on setting of the option `MPX_AUTOEXCLUDE_TIMEOUT`.

Pooled Connections

Each secondary node maintains an INC connection pool. The connection pool manages connections from the secondary node to the coordinator node used by INC for transactional communication.

The INC connection pool reduces communication overhead by re-using existing connections rather than opening new ones, and controls how many INC connections may be established by a secondary node to the coordinator at the same time.

Two database options govern the characteristics of the INC connections from the secondary servers:

- `MPX_MAX_CONNECTION_POOL_SIZE`: The maximum number of connections to be allowed to be open from a secondary node to the coordinator..
- `MPX_MAX_UNUSED_POOL_SIZE`: The maximum number of unused connections to be kept alive when the connection is returned to the unused pool.

2.2.2 Multiplex Interprocess Communication (MIPC)

MIPC connects all multiplex nodes to support distributed query processing and high availability.

As multiplex servers start, they establish MIPC connections.

MIPC is a fully meshed communication framework that runs on both public and private interconnection configurations. Public interconnection configuration is mandatory while private configuration is optional.

Private high-speed interconnection configurations are for distributed query processing. Currently, private interconnection configurations are restricted to physical networks supporting the TCP/IP protocol.

If no private interconnection configuration is provided, MIPC uses the legacy public interconnection configuration specified in the system table column SYSIQMPXSERVER.conn_info. This set of host/port addresses is shared between external user connections, INC connections, and internal MIPC connections.

Redundant networks provide a more robust cluster. If both private and public interconnection configurations exist, MIPC uses the private connections if available, and reroutes traffic to the public connections if the private network fails. The multiplex monitors the messages between nodes to detect failures.

Possible interconnection link failures include:

- A physical failure, such as a cable that is disconnected or broken
- A power supply failure, such as a piece of network infrastructure equipment

- A software failure within the networking stack

For recommendations on planning network configurations for distributed query processing, see the *SAP IQ Installation and Update Guide* for your platform.

3 Create a Multiplex

Convert a single running SAP IQ server into a multiplex.

Procedure

1. Plan and configure the main store and hardware to meet requirements.
2. Convert the server to a coordinator.
3. Configure the secondary node.
4. Add additional secondary nodes.

In this section:

3.1 Planning the Configuration

Before you create a multiplex, consider hardware and software requirements.

In this section:

Multiplex Planning Worksheet [page 18]
Before creating a multiplex, check that the paths of proposed objects meet requirements.

3.1.1 Multiplex Storage Requirements

Create multiplex stores on the appropriate device.

Stores	Supported Devices
IQ shared stores (IQ_SYSTEM_MAIN, IQ_SHARED_TEMP, user dbspaces)	Raw device supported **Note** Raw devices aren't available in Linux. See SAP Note 3483103 The RAW_DRIVER, and support for devices located in `/dev/raw/`, was deprecated in: • SuSE Linux Enterprise Server (SLES) 15.4 • Red Hat Enterprise Linux 9 Any storage type can be used for SAP IQ so long as it's configured in a way that SAP IQ supports (raw devices on non-Linux platforms, local file systems for databases on any platform, NFS on Linux, and GPFS on Linux) and follows proper sizing guidelines.
IQ non-shared store (IQ_SYSTEM_TEMP)	Both raw devices and operating system files are supported **Note** Raw devices aren't available in Linux. See SAP Note 3483103 The RAW_DRIVER, and support for devices located in `/dev/raw/`, was deprecated in: • SuSE Linux Enterprise Server (SLES) 15.4 • Red Hat Enterprise Linux 9
Catalog and transaction log	Operating system files required

Parent topic: Planning the Configuration [page 14]

Related Information

Setting Up Windows Access to Raw Devices [page 16]
Main Store Requirements [page 17]
Hardware Requirements [page 17]
Multiplex Planning Worksheet [page 18]

3.1.2 Setting Up Windows Access to Raw Devices

Windows systems restrict raw device access to user accounts with Administrator privilege.

Context

To run the SAP IQ servers using an account that lacks Administrator privilege, you must enable new device access permissions for that account after each system reboot.

The `rawaccedit` utility sets permissions for devices for the current session.

Set up read-write access for the write servers and read access for query servers.

Procedure

1. Type the following at a command prompt:

   ```
   rawaccedit
   ```

2. In the IQ Raw Device Access window, type the name of the user and the device to which you want to grant access.

   You can use `Alt` + `n` to tab to the User's Name box and `Alt` + `d` to tab to the Raw Device Name box.

Option	Description
To specify an unpartitioned raw device	Type unpartitioned drives are named with the format `\\?\Volume{GUID}`, where `<GUID>` is the device identifier. To find the device identifier, you may use the Windows utility `mountvol.exe` to print the GUID of all available devices.
To specify a partitioned raw device	Type the letter assigned to that partition.

3. Click **Add**.
4. Correct any errors in the user name and device name that display in the top panel and click **Update ACL and Exit**.

Results

Device access permissions remain until you restart Windows.

Task overview: Planning the Configuration [page 14]

Related Information

Multiplex Storage Requirements [page 15]
Main Store Requirements [page 17]
Hardware Requirements [page 17]
Multiplex Planning Worksheet [page 18]

3.1.3 Main Store Requirements

Make sure that your main store meets prerequisites for multiplex operation.

Main store file paths must be absolute or relative in such a way that they are accessible to all servers in the multiplex.

Migrate your database from an earlier release of SAP IQ to the current release.

Parent topic: Planning the Configuration [page 14]

Related Information

Multiplex Storage Requirements [page 15]
Setting Up Windows Access to Raw Devices [page 16]
Hardware Requirements [page 17]
Multiplex Planning Worksheet [page 18]

3.1.4 Hardware Requirements

An SAP IQ multiplex requires shared storage for data on IQ_SYSTEM_MAIN, IQ_SHARED_TEMP, and user dbspaces.

All machines running servers participating in the multiplex must have the current version of SAP IQ installed. For upgrade instructions, see the *SAP IQ Installation and Update Guide* for your platform.

When you convert an existing SAP IQ server to multiplex, it becomes the coordinator. For secondary servers, set up a computer with access to the shared IQ store using paths identical to those used by the coordinator.

SAP IQ does not support:

- Multiplexes of SAP IQ servers at different release levels.
- Heterogeneous multiplexes (UNIX and Windows servers in a mixed multiplex). Coordinator and secondary servers must be on the same operating system and hardware platform.

Parent topic: Planning the Configuration [page 14]

Related Information

Multiplex Storage Requirements [page 15]
Setting Up Windows Access to Raw Devices [page 16]
Main Store Requirements [page 17]
Multiplex Planning Worksheet [page 18]

3.1.5 Multiplex Planning Worksheet

Before creating a multiplex, check that the paths of proposed objects meet requirements.

Database paths on all platforms, whether raw device or operating system files, are limited to 128 bytes. SAP IQ supports:

- Raw devices for IQ shared stores (IQ_SYSTEM_MAIN, IQ_SHARED_TEMP, and user dbspaces).

  > **Note**
  >
  > Raw devices aren't available in Linux. See SAP Note 3483103 The RAW_DRIVER, and support for devices located in `/dev/raw/`, was deprecated in:
  >
  > - SuSE Linux Enterprise Server (SLES) 15.4
  > - Red Hat Enterprise Linux 9

- Both raw devices and operating system files for non-shared IQ stores (IQ_SYSTEM_TEMP)
- Operating system files only for the catalog and transaction log

Dialog item	Type/Length	Notes	Value
Server name	CHAR 128	The name of the coordinator. The name must be unique across the local area network.	

Dialog item	Type/Length	Notes	Value
Database path	CHAR 128	The absolute path to the database files. Store the database files on the local disk of the coordinator or secondary node, not on a remote location. The path must exists before executing the CREATE MULTIPLEX SERVER statement.	
IQ store paths (temp and main)	CHAR 128	All files in IQ_SYSTEM_MAIN and shared user dbspaces must be accessible in the same way using the same file paths from all nodes. IQ main dbspace paths are shared and temporary and `.iqmsg` dbspace paths are only valid on the owning node.	
User name and password	Minimum 6 characters long, containing upper and lowercase letters and numbers.	The only user initially created and is granted all IQ system and object-level privileges in the database.	
Host names	CHAR 128	The IP address or machine name where the database engine will run.	
Roles	reader or writer	The designated role of each secondary node. A reader can only run read-only operations against shared IQ objects. A writer can run read-only and read-write operations against shared IQ objects. Both a reader and writer can manipulate local data in temporary and SA base tables. The default, if not specified, is READER.	

Parent topic: Planning the Configuration [page 14]

Related Information

Multiplex Storage Requirements [page 15]
Setting Up Windows Access to Raw Devices [page 16]
Main Store Requirements [page 17]
Hardware Requirements [page 17]

3.2 Converting a Database to a Multiplex

To change an SAP IQ server to act as a coordinator in a multiplex, connect to the SAP IQ server and use `CREATE MULTIPLEX SERVER`.

Prerequisites

- Requires the MANAGE MULTIPLEX system privilege.
- All IQ main store file paths must be absolute or relative in a way accessible to all servers in the multiplex.
- The IQ server being converted is running.
- Your system meets hardware prerequisites. The main store dbfiles must be visible to secondary nodes.

Procedure

1. In a command prompt window, start Interactive SQL and connect to the server being converted.

   ```
   dbisql
   ```

2. Run the `CREATE MULTIPLEX SERVER` command.

   {·} Sample Code

   On Windows:

   ```
   CREATE MULTIPLEX SERVER mpxnode_w1
       DATABASE 'y:\mpxnode_w1\mpxtest.db' HOST 'host_w1' PORT 2957 ROLE
   WRITER
   ```

   Here, `y:\mpxnode_w1\mpxtest.db` specifies where the writer catalog will reside when `dbbackup` is run on the writer node.

   On UNIX:

   ```
   CREATE MULTIPLEX SERVER mpxnode_w1
       DATABASE '/mpxnode_w1/mpxtest.db' HOST 'host_w1' PORT 2957 ROLE WRITER
   ```

   As a final step in the conversion process, the original server shuts down and must be restarted to re-initialize it as a multiplex coordinator. As a result of the shutdown, Interactive SQL may return a disconnected (-308) error that requires no corrective action.)

> **Note**
>
> While converting simplex to multiplex on machines with multiple interfaces (for example public, private), use the `-x` start-up option with the `MyIP` parameter to correctly configure the public IP of the coordinator as follows:
>
> ```
> start_iq @default.cfg -n <server_name> -x
> tcpip{MyIP=<public_IP>;port=<port>}
> ```
>
> You can later change/configure a private IP address using the command `ALTER multiplex server`.

3. Restart the converted server.

> **Sample Code**
>
> On Windows:
>
> ```
> start_iq z:\mpxnode_c1\mytest.db -n mpxnode_c -x tcpip{port=2763}
> ```
>
> On UNIX
>
> ```
> start_iq /mpxnode_c1/mytest.db -n mpxnode_c -x "tcpip{port=2763}"
> ```

Next Steps

Configure the secondary node.

In this section:

Multiplex Database Files [page 21]
Each server in the multiplex uses several files to store data, metadata, and configuration information.

3.2.1 Multiplex Database Files

Each server in the multiplex uses several files to store data, metadata, and configuration information.

> **Note**
>
> Raw devices aren't available in Linux. See SAP Note 3483103 The RAW_DRIVER, and support for devices located in `/dev/raw/`, was deprecated in:
>
> - SuSE Linux Enterprise Server (SLES) 15.4
> - Red Hat Enterprise Linux 9

Each multiplex server includes some (or all) of the files listed here:

Table 1: Contents of multiplex database directories

Filename	Purpose
`dbname`.db	File that contains the catalog store. This cannot be a raw device.
`dbname.iq`	This is the IQ_SYSTEM_MAIN file which is shared between all MPX nodes.
`dbname`.iqmsg	File that contains messages from SAP IQ
`dbname`.iqtmp	This is the IQ temporary store. This file exists only if the temporary file is added as an OS file. The IQ_SYSTEM_TEMP store can be on either a raw device or a system file.
`dbname`.log	File that contains the database transaction log. This cannot be a raw device.
`dbname`.lmp	License management property file. Creating a database generates this file automatically. Dropping a database deletes all database files except this one.

3.3 Configuring the Secondary Node

After creating the secondary node, synchronize the data between the coordinator and secondary node.

Prerequisites

- Requires the SERVER OPERATOR, BACKUP DATABASE, and MANAGE ANY DBSPACE system privileges.
- The server has been converted to a coordinator and is running.

Procedure

1. On the secondary node, switch to the secondary directory where the copy of the catalog store from the coordinator should be copied.
2. Copy the catalog store to the directory.

   {·} Sample Code

   In this example, the credentials, port, engine are of the coordinator. The folder is where copy of the catalog store will reside.

   (Windows)

   ```
   dbbackup -y -x -c
   "uid=<user_name>;pwd=<password>;links=tcpip{host=<coordinator_host>:<coordi
   nator_port>};Server=mpxnode_c" -d -o bkup.ot y:\mpxnode_w1
   ```

(UNIX)

```
dbbackup -y -d -c
"uid=<user_name>;pwd=<password>;links=tcpip{host=<coordinator_host>:port=27
63};Server=mpxnode_c" -o bkup.ot '/mpxnode_w1'
```

3. (UNIX Only) Remove the transaction log files (if they exist) from the directory where the copy of the catalog store resides.

   Sample Code

   ```
   rm -rf mpxtest.log
   ```

4. Set the transaction log file path.

   Sample Code

   ```
   dblog -r -t mpxtest.log mpxtest.db
   ```

5. Start the secondary node.

   Sample Code

   (Windows)

   ```
   start_iq  y:\mpxnode_w1\mpxtest.db -n mpxnode_w1 -x tcpip{port=2957}
   ```

   (UNIX)

   ```
   start_iq  /mpxnode_w1/mpxtest.db -n mpxnode_w1 -x 'tcpip{port=2957}'
   ```

   Starting the secondary node creates a message log for you, but you must create the temporary store (dbfile) yourself. See Adding a Temporary Dbfile [page 37].

6. Start Interactive SQL and connect to the secondary node.
7. Add the temporary dbfile (by specifying the absolute path of the secondary writer node directory) to the `IQ_SYSTEM_TEMP` dbspace using the `ALTER DBSPACE` statement.

   Secondary servers do not allow `CREATE DBSPACE`. Specify the logical file name (`mpxnode_w1_temp`), the physical file name (`mpxnode_w1_temp1.iqtmp`), and the size of the dbfile (`500`). You cannot run IQ queries until the dbfile is added.

   Sample Code

   ```
   ALTER DBSPACE IQ_SYSTEM_TEMP ADD FILE mpxnode_w1_temp '/mpxnode_w1/
   mpxnode_w1_temp1.iqtmp' SIZE 500
   ```

   In this statement, `'/mpxnode_w1/mpxnode_w1_temp1.iqtmp'` indicates the absolute path of the secondary writer node directory.

8. Run `sp_iqmpxvalidate` on the secondary node to make sure that no errors are reported on this server.

Next Steps

Add any additional secondary nodes to the multiplex.

3.4 Adding Additional Secondary Nodes

Repeat the steps for converting a database to a multiplex and configuring a secondary node to add each additional secondary node.

> **Note**
>
> When adding a secondary node to an existing multiplex, the coordinator does not automatically shut down at the end of the create multiplex process. There is also no need to restart the coordinator before configuring the new secondary node.

Related Information

Converting a Database to a Multiplex [page 20]
Configuring the Secondary Node [page 22]

4 Manage a Multiplex

You can use Interactive SQL and command line utilities to manage multiplex servers.

Enter Interactive SQL statements individually at the command line or run them in scripts to query, modify, or load SAP IQ servers.

In this section:

4.1 Multiplex System Privileges

The system privileges required to perform authorized tasks in a multiplex environment vary by task.

Grant system privileges using the WITH ADMIN OPTION, WITH NO ADMIN OPTION, or WITH ADMIN ONLY OPTION clause. If you do not specify a clause, the default is WITH NO ADMIN OPTION.

System Privilege	Description	User/Roles Can Inherit From
ACCESS SERVER LS	Allows logical server connection using the SERVER logical server context.	• SYS_AUTH_DBA_ROLE • SYS_AUTH_SA_ROLE • SYS_AUTH_MULTIPLEX_ADMIN_ROLE • SYS_AUTH_OPERATOR_ROLE
MANAGE MULTIPLEX	Allows a user to: • Create and manage a multiplex. • Issue multiplex-related CREATE, ALTER, DROP, or COMMENT statements on logical server policies and servers. • Perform exclusive assignment of a dbspace to logical servers. • Release a populated dbspace from the exclusive use of a logical server. • Manage failover configurations, and is required for a manual failover.	• SYS_AUTH_DBA_ROLE • SYS_AUTH_SA_ROLE • SYS_AUTH_MULTIPLEX_ADMIN_ROLE
SERVER OPERATOR and BACKUP DATABASE	Allows a user to synchronize copies of the coordinator's version of the database catalog to secondary servers.	• SYS_AUTH_DBA_ROLE • SYS_AUTH_SA_ROLE
MANAGE ANY DBSPACE	Allows a user to: Create, alter, drop, and comment on dbspaces. Grant and revoke CREATE object-level privileges on dbspaces. Move data to any dbspace. Issue a read-only selective restore statement on any dbspace. Run the database delete file function.	• SYS_AUTH_DBA_ROLE • SYS_AUTH_SA_ROLE • AYA_AUTH_SPACE_ADMIN_ROLE

In this section:

DBO System Role in a Multiplex Environment [page 27]

By default, the DBO system role is granted the SYS_AUTH_DBA_ROLE compatibility role. This ensures that the DBO system role is granted all privileges necessary to execute multiplex management stored procedures.

4.1.1 DBO System Role in a Multiplex Environment

By default, the DBO system role is granted the SYS_AUTH_DBA_ROLE compatibility role. This ensures that the DBO system role is granted all privileges necessary to execute multiplex management stored procedures.

You cannot revoke the underlying system privileges of a compatibility role; you must first migrate it to a user-defined role. Only then can you revoke individual underlying system privileges from the new role and grant them to other user-defined roles per the organization's security requirements. This enforces separation of duties.

You can migrate compatibility roles automatically or manually. The method of migration can impact the ability of a system role or the DBO system role to continue performing authorized tasks.

Automatic Migration

The `ALTER ROLE` statement creates a new user-defined role, automatically grants all underlying system privileges of the compatibility role to the new user-defined role, makes each member of the compatibility role a member of the new user-defined role, then drops the compatibility role.

Automatic migration assumes that the destination user-defined role does not already exist and that all system privileges are migrated to the same new user-defined role.

Manual Migration

Use the `CREATE ROLE` statement to create a new user-defined role. Use the `GRANT` statement to grant each underlying system privilege to one or more users or roles. Use the `DROP` statement to drop the compatibility role once all underlying system privileges are granted to at least one other user or role.

Members of the migrated compatibility role are not automatically granted membership in the new user-defined role. As a result, members of some system roles may no longer be able to perform the expected privileged tasks once the compatibility role is dropped. You must grant membership in the new user-defined role to the affected system roles or directly grant the required system privileges to affected members.

Regardless of the migration method used, once a compatibility role or the SYS_AUTH_DBA_ROLE role is dropped, if you revoke a system privilege from the new user-defined role and grant it to another user-defined role, you must do one of the following to ensure that system roles especially the dbo system role, retains all the system privileges required to execute the applicable privileged tasks or multiplex management stored procedures:

- Grant each system privilege revoked from the migrated user-defined role directly to the applicable system roles or dbo role; or

- Grant membership in the user-defined role to which the system privileges are granted to the applicable system roles or dbo system role.

4.2 Starting a Multiplex Server

The `start_iq` utility starts multiplex servers interactively or in scripts.

Context

Specify command line options with the `start_iq` command. The `-n <engine_name>` switch is required, where `<engine_name>` matches the server name used when creating the multiplex server. The `-x` connection string value must match TCP/IP connection parameters specified when creating the multiplex server. The database file path must match the database file path as specified when creating the multiplex server. For applications, the `-gn` value must be set higher than the total number of expected simultaneous requests, plus 5 (for internal events and connections).

Procedure

Run the start server utility, `start_iq`, specifying command line parameters for your server.

The following command starts a server named `host1_test1_writer`. It uses the database named `mpxtest.db` in the folder `mpxnode_c1`:

Option	Description
Windows	`start_iq -n host1_test1_writer -x "tcpip{port=2763}" y:\mpxnode_w1\mpxtest1.db`
UNIX	`start_iq -n host1_test1_writer -x "tcpip{port=2763}" /mpxnode_w1/mpxtest1.db`

4.3 Setting Up TLS Connections for the Multiplex

TLS connections enable encrypted communication between the multiplex servers and client applications. You will need to run TLS on the coordinator and secondary nodes. You'll need certificates, which you may buy from a certificate authority, or create using the Certificate Creation utility (createcert).

Context

The certificate that you specify with the -ec startup parameter needs to contain the private key and public certificate of the SAP IQ server. The certificate file should only contain the root certificate of the signer and not the signer's private key.

Procedure

1. Generate a CA (Certificate Authority) certificate. This example uses the Certificate Creation utility (createcert) to generate a certificate.

```
$ mkdir certificates
$ cd certificates
mysystem# createcert
SQL Anywhere X.509 Certificate Generator Version 16.0.0.807
Warning: The certificate will not be compatible with older versions
of the software including version 12.0.1 prior to build 3994 and version 16.0
prior to build 1691. Use the -3des switch if you require compatibility.
Enter RSA key length (512-16384): 2048
Generating key pair...
Country Code: CA
State/Province: Ontario
Locality: Waterloo
Organization: SAP Canada
Organizational Unit: Engg
Common Name: CA certificate
Enter file path of signer's certificate:
Certificate will be a self-signed root
Serial number [generate GUID]:
Generated serial number: 7e051580ec5b11e32349ae8781ad720
Certificate valid for how many years (1-100): 1
Certificate Authority (Y/N) [N]: Y
1.  Digital Signature
2.  Nonrepudiation
3.  Key Encipherment
4.  Data Encipherment
5.  Key Agreement
6.  Certificate Signing
7.  CRL Signing
8.  Encipher Only
9.  Decipher Only
Key Usage [6,7]: 6
Enter file path to save certificate: CA.crt
Enter file path to save private key: CA.key
Enter password to protect private key: capwd
Enter file path to save identity: CA.id
$ ls
```

```
CA.crt  CA.id   CA.key
```

2. Create an IQ Server certificate.

```
$ createcert
SQL Anywhere X.509 Certificate Generator Version 16.0.0.807
Warning: The certificate will not be compatible with older versions
of the software including version 12.0.1 prior to build 3994 and version 16.0
prior to build 1691. Use the -3des switch if you require compatibility.
Enter RSA key length (512-16384): 2048
Generating key pair...
Country Code: CA
State/Province: Ontario
Locality: Waterloo
Organization: SAP Canada
Organizational Unit: Engg
Common Name: IQ Server
Enter file path of signer's certificate: CA.crt
Enter file path of signer's private key: CA.key
Enter password for signer's private key: capwd
Serial number [generate GUID]:
Generated serial number: 11acb374ec5c11e38000af2328c8ed20
Certificate valid for how many years (1-100): 1
Certificate Authority (Y/N) [N]: N
1.  Digital Signature
2.  Nonrepudiation
3.  Key Encipherment
4.  Data Encipherment
5.  Key Agreement
6.  Certificate Signing
7.  CRL Signing
8.  Encipher Only
9.  Decipher Only
Key Usage [1,3,4,5]: 1
Enter file path to save certificate: iqserver.crt
Enter file path to save private key: iqserver.key
Enter password to protect private key: test
Enter file path to save identity: iqserver.id
$ ls
CA.crt    CA.id     CA.key    iqserver.crt    iqserver.id     iqserver.key
```

In this example:

- `iqserver.id` – should be used as Server Identity file (identity TLS parameter)
- `CA.crt` – trusted_certificate TLS parameter when connecting using dbisql client
- `CA.crt` – database public option (public.trusted_certificates_file)

3. Start the coordinator with this configuration file:

```
# iqdemo.cfg
# ------------------------------------------------------------
# Default startup parameters for the SAP IQ demo database
# ------------------------------------------------------------
-n  desi_iqdemo_154
-x  tcpip{port=26000}
# The following parameters are also found in the configuration file
# /system1/desi/iq162/IQ-16_2/scripts/default.cfg.  Any parameters not
specified below
# and not in the start up parameter list, will be added by start_iq
# using default.cfg as a guide.
#-cl 512m
#-ch 1024m
-c 1024m
-gc 20
-gd all
-gl all
```

```
-gm 10
-gp 4096
-iqtss 1024
-iqmc 5500
-iqtc 15000
-iqlm 5500
-iqmsgsz 5
-iqmsgnum 10
-iqmt 100
-ec "tls(tls_type=rsa;identity=/system1/desi/iq162/devices/
iqserver.id;identity_password=Mypassword)"
-s none
-zr all
-o console.out
```

4. Create this configuration file for the write server:

```
-c 128m
-gc 30
-gd DBA
-gl all
-gm 100
-iqmc 1000
-iqtc 1000
-ec "tls(tls_type=rsa;identity=//system1/desi/iq162/devices/
iqserver.id;identity_password=Mypassword)"
-iqlm 1000
```

5. Start the write server with this script:

```
bhandari@system:/system1/desi/iq162/devices/w1:171 >  more start_server.sh
#!/bin/sh
start_iq -STARTDIR "/system1/desi/iq162/devices/w1" @params.cfg
-n desi_iqdemo_162_w1 -x "tcpip(port=26001)"
"/system1/desi/iq162/devices/w1/iqdemo.db"
```

6. On the coordinator, configure the location of the text file that lists trusted certificate authorities that sign server certificates. You need the SET ANY SECURITY OPTION system privilege to set this option. You must set this option to use TLS for internal communication between multiplex nodes.

> **Note**
>
> Specify a file path at a shared location accessible from all the multiplex nodes using the same path.

```
SET OPTION PUBLIC.TRUSTED_CERTIFICATES_FILE = '/system1/desi/iq161/devices/
CA.crt'
```

7. Synchronize the secondary servers with this script:

```
::::::::::::::
sync_server_encr.sh
::::::::::::::
#!/bin/sh
# Usage: sync_server.bat login password
if [ $# -lt 2 ]
then
  echo "Usage: sync_server.sh <login> <password>"
  exit 1
fi
uid=$1
pwd=$2
CONN_STR="eng=desi_iqdemo_162;LS=SERVER;ENC=tls(fips=NO;
tls_type=rsa;trusted_certificate=/system1/desi/iq161/devices/
CA.crt);links=tcpip(host=system2.svr;
```

```
port=26000);uid=$uid;pwd=$pwd"
dbbackup -y -d -c $CONN_STR /system1/desi/iq161/devices/w1
rm -rf "iqdemo.log"
dblog -r -t "iqdemo.log" "/system1/desi/iq161/devices/w1/iqdemo.db"
```

See *-ec database server option* in SAP IQ Utility Reference.

4.4 Checking the Multiplex Status

Check the multiplex status to determine if any servers are not responding or not included in the multiplex, and to see which server will be the new coordinator in the event of a failover.

Prerequisites

- Requires MANAGE MULTIPLEX system privilege
- Requires `EXECUTE` object-level privilege on the `sp_iqmpxinfo` procedure.

Procedure

1. Start Interactive SQL and connect to the coordinator.

   Sample Code

   ```
   dbisql
   ```

2. Run `sp_iqmpxinfo`.

   See *sp_iqmpxinfo Procedure* in SAP IQ SQL Reference.

Related Information

sp_iqmpxinfo Procedure

4.5 Adding a Secondary Server to a Multiplex

Follow these steps to add secondary servers to a multiplex using Interactive SQL.

Prerequisites

- Requires the MANAGE MULTIPLEX, SERVER OPERATOR, BACKUP DATABAE and MANAGE ANY DBSPACE system privileges.
- The data folder to contain a synchronized copy of the `.db` file exists on the secondary node.
- All IQ main store file paths must be absolute or relative in a way accessible to all servers in the multiplex.
- The coordinator is running.

Procedure

1. Start Interactive SQL and connect to the coordinator.

   ```
   dbisql
   ```

2. Add a secondary node to the multiplex.

   Sample Code

   On Windows:

   ```
   CREATE MULTIPLEX SERVER mpxnode_w2
       DATABASE 'X:\mpxnode_w2\mpxtest.db' HOST 'host_w2' PORT 2958 ROLE
   WRITER
   ```

   On UNIX:

   ```
   CREATE MULTIPLEX SERVER mpxnode_w2
       DATABASE '/mpxnode_w2/mpxtest.db' HOST 'host_w2' PORT 2958 ROLE WRITER
   ```

   See *CREATE MULTIPLEX SERVER Statement* in SAP IQ SQL Reference.

3. On the secondary node, switch to the directory where the copy of the catalog store from the coordinator will reside.
4. Copy the catalog store to the directory.

   Sample Code

   In this example, the credentials, port, engine are of the coordinator. The folder is where copy of the catalog store will reside.

On Windows:

```
dbbackup -y -d -c
"uid=<user_name>;pwd=<password>;links=tcpip{port=2763};eng=mpxnode_c1" -o
bkup.ot X:\mpxnode_w2
```

On UNIX:

```
dbbackup -y -d -c
"uid=<user_name>;pwd=<password>;links=tcpip{port=2763};eng=mpxnode_c1" -o
bkup.ot '/mpxnode_w2'
```

5. (UNIX Only) If it exists, remove the transaction log files in the directory where the copy of the catalog store resides.

   {·} Sample Code

   ```
   rm -rf mpxtest.log
   ```

6. Run `dblog` to reset the transaction log.

   {·} Sample Code

   ```
   dblog -r -t mpxtest.log mpxtest.db
   ```

7. Start the secondary server.

   {·} Sample Code

   For Windows:

   ```
   start_iq  X:\mpxnode_w2\mpxtest.db -n mpxnode_w2 -x tcpip{port=2958}
   ```

   For UNIX:

   ```
   start_iq  /mpxnode_w2/mpxtest.db -n mpxnode_w2 -x 'tcpip{port=2958}'
   ```

   Starting the secondary node creates a message log for you, but you must create the temporary store (dbfile) yourself. See Adding a Temporary Dbfile [page 37].

8. Start Interactive SQL and connect to the secondary node.
9. Add the temporary dbfile to the IQ_SYSTEM_TEMP dbspace. You cannot run IQ queries until the dbfile is added.

   Secondary servers do not allow `CREATE DBSPACE`.

   {·} Sample Code

   Specify the logical file name (`mpxnode_w2_temp`), the physical file name (`mpxnode_w2_temp1.iqtmp`), and the size of the dbfile.

   ```
   ALTER DBSPACE IQ_SYSTEM_TEMP ADD FILE mpxnode_w1_temp
   'mpxnode_w1_temp1.iqtmp' SIZE 500
   ```

10. Run `sp_iqmpxvalidate` on the secondary node to make sure that no errors are reported on this server.

Related Information

ALTER DBSPACE Statement
CREATE MULTIPLEX SERVER Statement
sp_iqmpxvalidate Procedure

4.6 Dropping a Secondary Server from a Multiplex

Dropping a secondary server removes it from the multiplex configuration.

Prerequisites

- Requires the MANAGE MULTIPLEX system privilege.
- The coordinator is running.

Context

If the target server is running, it is recommended you stop it before dropping it, but it is not mandatory. It will stop by itself once dropped. You cannot drop the designated failover node unless it is the last secondary node. Once you drop the last secondary node, the multiplex is converted back to an SAP IQ server and the coordinator shuts down.

The free list is an internal structure that manages space allocation for a dbspace. A write server that is holding free list cannot be dropped.

Procedure

1. Start Interactive SQL and connect to the coordinator.

   ```
   dbisql
   ```

2. Drop the secondary node.

   Sample Code

   ```
   DROP MULTIPLEX SERVER <server_name>
   ```

See *DROP MULTIPLEX SERVER Statement* in SAP IQ SQL Reference.

In this section:

Related Information

ALTER LS POLICY Statement
Coordinator Failure [page 45]
DROP MULTIPLEX SERVER Statement
Global Transaction Resiliency [page 71]
MPX_LIVENESS_TIMEOUT Option

4.6.1 Reclaiming Free List

A normal restart of the write server gives up its free list, so that you can shut it down and drop it.

Prerequisites

- Requires the MANAGE MULTIPLEX system privilege.

Procedure

1. Verify that the write server process is shut down.
2. Restart the coordinator with the `-iqmpx_reclaimwriterfreelist` switch. This forces the coordinator to reclaim the writer's free list so that you can then drop it.

> ⚠ Caution
>
> If the write server process is still writing to the database when the coordinator reclaims its free list, this may corrupt the database.

4.7 Altering a Multiplex Server

You can use Interactive SQL to alter a multiplex server.

The named server automatically shuts down once the change is committed, except when altering role from reader to writer.

In this section:

4.7.1 Adding a Temporary Dbfile

Use the ALTER DBSPACE command to add additional dbfiles to the IQ_SYSTEM_TEMP dbspace on the secondary node.

Prerequisites

- Requires the MANAGE ANY DBSPACE system privilege.
- The secondary node is running.
- All IQ main store file paths must be absolute or relative in a way accessible to all servers in the multiplex.

Procedure

1. Start Interactive SQL and connect to the secondary node.

```
dbisql
```

2. Add the temporary dbfile to the IQ_SYSTEM_TEMP dbspace.

   Secondary servers do not allow `CREATE DBSPACE`. Specify the logical file name (`mpxnode_w1_temp`), the physical file name (`mpxnode_w1_temp1.iqtmp`), and the size of the dbfile. You cannot run IQ queries until a temporary store is added.

   Sample Code

   ```
   ALTER DBSPACE IQ_SYSTEM_TEMP ADD FILE mpxnode_w1_temp
   'mpxnode_w1_temp1.iqtmp' SIZE 500
   ```

3. Run `sp_iqmpxvalidate` to make sure that no errors are reported on this server.

4.7.2 Changing Host and Port

You can use Interactive SQL to change the host and port information for a multiplex server.

Prerequisites

- Requires the MANAGE MULTIPLEX system privilege.
- The coordinator is running.

Context

Run `sp_iqmpxinfo` to review change. See Checking the Multiplex Status [page 32].

Procedure

1. Start Interactive SQL and connect to the coordinator.

   ```
   dbisql
   ```

2. Change the host and port.

   ```
   ALTER MULTIPLEX SERVER <server name> HOST '<host_name>' PORT <port_number>
   ```

   This command also shuts down the named server.
3. Restart the server.
4. Verify the change using `sp_iqmpxinfo`.

4.7.3 Changing Server Roles

Use Interactive SQL to change a reader to a write server.

Prerequisites

- Requires the MANAGE MULTIPLEX system privilege.
- The coordinator is running.

Context

You cannot change the role of coordinator. Changing a write server to a reader automatically shuts down the server. Run `sp_iqmpxinfo` to review change. See Checking the Multiplex Status [page 32].

Procedure

1. Start Interactive SQL and connect to the coordinator.

   ```
   dbisql
   ```

2. Change the role of the node.

   Sample Code

   ```
   ALTER MULTIPLEX SERVER <server_name> ROLE {WRITER|READER}
   ```

3. Restart the changed server, if necessary.

4.7.4 Moving the Database to a New Location

Change the file path with Interactive SQL. This moves your database, such as to a disk with more space.

Prerequisites

- Requires the MANAGE MULTIPLEX system privilege.
- All IQ main store file paths must be absolute or relative in a way accessible to all servers in the multiplex.
- The coordinator is running.

Context

Enter the path as an absolute value. The folder structure doesn't need to exist before running the ALTER MULTIPLEX SERVER command. Run `sp_iqmpxinfo` to review change. See Checking the Multiplex Status [page 32].

Procedure

1. Start Interactive SQL and connect to the coordinator

   ```
   dbisql
   ```

2. Define the new database file path:

   Sample Code

   ```
   ALTER MULTIPLEX SERVER <server_name> DATABASE
   '<new_db_file_path>\<database_name>.db'
   ```

   This command also shuts down the named server.
3. Create the new folder structure, if it doesn't already exist.
4. Move all files except the `.iq` file to the new location.
5. Restart the server specifying the new path to the database.

Next Steps

To move the `.iq` file, see Changing an IQ_SYSTEM_MAIN File Path [page 62]

.

4.7.5 Including or Excluding a Multiplex Server

Add or remove a secondary node from the multiplex.

Prerequisites

- Requires the MANAGE MULTIPLEX system privilege.
- The coordinator is running.
- All IQ main store file paths must be absolute or relative in a way accessible to all servers in the multiplex.

Context

If a secondary server will be shut down for an extended period, exclude that server from the multiplex.

Procedure

1. Start Interactive SQL and connect to the coordinator.

   ```
   dbisql
   ```

2. If excluding a server, you should shut it down first, even though it will eventually shut itself down.
3. Add or remove a secondary node from the multiplex.

   {·} Sample Code

   ```
   ALTER MULTIPLEX SERVER <server name> STATUS {INCLUDED|EXCLUDED}
   ```

4. If including a server, synchronize the server and then start it. See Synchronizing Multiplex Servers [page 42]

4.7.6 Renaming a Multiplex Server

You can use Interactive SQL to change the name of a multiplex server.

Prerequisites

- Requires the MANAGE MULTIPLEX system privilege.
- The coordinator is running.

Context

Run `sp_iqmpxinfo` to review change. For reference information, see *sp_iqmpxinfo Procedure* in SAP IQ SQL Reference.

Procedure

1. Start Interactive SQL and connect to the coordinator.

   ```
   dbisql
   ```

2. Rename the multiplex server.

   Sample Code

   ```
   ALTER MULTIPLEX SERVER <oldname> RENAME <newname>
   ```

   This command renames and shuts down the named server. See *ALTER MULTIPLEX SERVER Statement* in SAP IQ SQL Reference.

3. Restart the server.

Related Information

ALTER MULTIPLEX SERVER Statement
Checking the Multiplex Status [page 32]
sp_iqmpxinfo Procedure

4.8 Synchronizing Multiplex Servers

Synchronizing copies the coordinator's version of the database catalog to secondary servers.

Prerequisites

- Requires the BACKUP DATABASE and SERVER OPERATOR system privileges.

Context

Perform these steps on each secondary server in the multiplex.

Procedure

1. On the secondary server, remove the `.LOG` transaction log file from the folder where the copy of the catalog store resides.
2. Copy the catalog store to the directory.

   Sample Code

   In this example, the credentials, port, engine are of the coordinator. The folder is where copy of the catalog store will reside.

   On Windows:

   ```
   dbbackup -y -d -c "uid=<user_name>;pwd=<password>;links=tcpip{port=2763};
   eng=mpxnode_c1" -o btup.ot y:\mpxnode_w1
   ```

   On UNIX:

   ```
   dbbackup -y -d -c "uid=<user_name>;pwd=<password>;links=tcpip{port=2763};
   eng=mpxnode_c1" -o btup.ot '/mpxnode_w1'
   ```

3. (UNIX Only) If it exists, remove the transaction log files in the directory where the copy of the catalog store resides.

   Sample Code

   ```
   rm -rf mpxtest.log
   ```

4. Set the transaction log file path.

   Sample Code

   ```
   dblog -r -t mpxtest.log mpxtest.db
   ```

5. Restart the secondary node:

Related Information

Shared Store Path Requirement [page 10]
Updates on IQ_SYSTEM_MAIN [page 61]
Adding a Catalog Dbspace [page 69]
Adding Space to IQ_SYSTEM_MAIN on a Coordinator [page 64]
Changing an IQ_SYSTEM_MAIN File Path [page 62]
Replacing the Coordinator (Manual Failover) [page 48]

4.9 Multiplex Login Management

Login policies control the capabilities of users and connections to a database.

Login management commands that you execute on any multiplex server are automatically propagated to all servers in the multiplex. For best performance, execute these commands, or any DDL, on the coordinator.

For an overview of SAP IQ login policies, see *SAP IQ Administration: User Management and Security*.

In this section:

Related Information

Login Policies [page 96]

4.9.1 Counting User Connections

To avoid exceeding the maximum number of connections, periodically check the current number of user connections.

Procedure

From the coordinator, run the `DB_PROPERTY` system function:

```
SELECT db_property('conncount')
```

The `DB_PROPERTY` function returns the number of client connections and the number of INC connections made by the secondary nodes to the coordinator.

The INC connections do not affect the number of connections (set by startup parameter `-gm`) that can be made to the coordinator node.

4.10 High Availability

SAP IQ Multiplex is a highly available system.

As with any system, failures are possible on any node. A multiplex with many nodes improves service availability if one or more nodes fail, because the work load can transfer to other participating nodes.

Many SAP IQ multiplex features can help you build and maintain systems with high service availability. Among these are coordinator failover, transaction resiliency, and the use of logical servers to isolate read-only query workload.

In this section:

4.10.1 Coordinator Failure

Clients on other nodes can be affected if the current coordinator node fails. SAP IQ provides automatic coordinator failover to minimize disruption.

Automatic failover includes an unconditional restart of the failover node as coordinator. For more requirements, see Coordinator Failure and Restart [page 51].

If SAP IQ Cockpit is unavailable when the coordinator fails, restart the node immediately or promote another server to be the coordinator. See Coordinator Failure [page 45].

During coordinator failure, read-write operations on the failed node roll back. Clients connected to a failed coordinator experience an outage. When the clients try to reconnect, they can be redirected to an available node by means of the login redirection feature or a third-party redirector. Depending on the severity of the failure, the failed node can be restarted if it is a software issue or restarted after fixing a hardware or disk issue.

Client Location	Result
Reader node	Not affected by coordinator failure.

Client Location	Result
Writer node	The clients experience a pause until the coordinator is brought back up or failed over. If the coordinator cannot be brought back up or failed over within a user controlled time out period, then these read-write transactions roll back and the clients experience an outage. The clients on writer nodes that are doing read-write operations periodically need more space in shared main dbspaces or require global locks on tables they modify. When that happens, these transactions suspend. These dependencies make it critical that the coordinator always stays up.

In this section:

Related Information

ALTER LS POLICY Statement
DROP MULTIPLEX SERVER Statement
Dropping a Secondary Server from a Multiplex [page 35]
Global Transaction Resiliency [page 71]
MPX_LIVENESS_TIMEOUT Option

4.10.1.1 What Is a Designated Failover Node?

A multiplex requires a designated failover node to take over as coordinator if the current coordinator is not running.

Use the `sp_iqmpxinfo` procedure to display the designated failover node. See *sp_iqmpxinfo Procedure* in SAP IQ SQL Reference.

During multiplex creation, the first secondary server created becomes the designated failover node, but you can later designate any other secondary server as failover node.

The designated failover node manages automatic coordinator failover and restart.

If your system is ineligible for automatic failover, you can perform manual failover from the command line or from SAP IQ Cockpit.

SAP IQ must be installed on the designated failover node for failover to be successful.

Related Information

Designating a Failover Node [page 47]
sp_iqmpxinfo Procedure

4.10.1.2 Designating a Failover Node

You designate a failover node to continue the coordinator role if that the current coordinator is unable to continue.

Prerequisites

- Requires the MANAGE MULTIPLEX system privilege.
- The coordinator is running.

Context

The designated failover node defaults to the first multiplex server added to the multiplex but any secondary node can be used. Use a reader for the designated failover node, if possible. Readers have no pending writable transactions, which makes failover easier. Run the `sp_iqpmxinfo` procedure to review the current designated failover node. See *sp_iqmpxinfo Procedure* in SAP IQ SQL Reference

.

Procedure

1. Start Interactive SQL and connect to the coordinator.

   ```
   dbisql
   ```

2. Assign the failover server. It must be included and part of the multiplex

```
ALTER MULTIPLEX SERVER <server-name> ASSIGN AS FAILOVER SERVER
```

Related Information

ALTER MULTIPLEX SERVER Statement
sp_iqmpxinfo Procedure
What Is a Designated Failover Node? [page 46]

4.10.1.3 Replacing the Coordinator (Manual Failover)

Make sure that the coordinator is no longer running before you replace it.

Prerequisites

- The coordinator process must be dead before manual failover.

Context

> **Note**
>
> In a worst-case scenario, the former coordinator computer might be running but either disconnected from the network or in a hardware hibernation state. In this situation, you cannot log into the coordinator computer, but the coordinator computer could start functioning normally without warning. Ideally, you should shut down the computer on which the coordinator was running during the manual failover process.

Procedure

1. Ensure that the coordinator process is dead and that ENABLE_AUTOMATIC_FAILOVER logical server policy option is OFF.

   > **Caution**
   >
   > Never initiate manual failover while the former coordinator process is alive or when automatic failover is enabled. Database corruption may result.

If there were any read-write transactions running on secondary nodes when the original coordinator was shut down, these transactions roll back. Ideally if the coordinator is running on dedicated server hardware, that computer should be shut down during the failover process.

Option	Description
UNIX	1. Log into the coordinator machine and make sure that the environment variables are set. 2. Issue the command to stop the appropriate `iqsrv17` process: `stop_iq`
Windows	1. Log into the coordinator machine. 2. Start Task Manager and look for the process name `iqsrv16.exe`. 3. Stop the `iqsrv16.exe` process.

2. Connect to the designated failover node and run `COMMIT`, then `BEGIN TRANSACTION` to ensure that this node is up to date with the latest TLV log.

   Shut down the designated failover node cleanly, using the `dbstop` utility.
3. At the command line, restart the intended coordinator using the failover switch (`-iqmpx_failover 1`) on the server start up utility:

   Sample Code

   For Windows:

   ```
   start_iq z:\mpxnode_c1\mpxtest.db -iqmpx_failover 1 -n mpxnode_c1 -x
   tcpip{port=2757}
   ```

   For UNIX:

   ```
   start_iq /mpxnode_c1/mpxtest.db -iqmpx_failover 1 -n mpxnode_c1 -x
   "tcpip{port=2757}"
   ```

   Once the server starts, the failover process is complete and the designated failover node is the new coordinator node. After failover, on the next transaction, other secondary servers recognize the new coordinator and connect to it for all read-write transactions. The former coordinator becomes a reader and can be started as a regular secondary node once you synchronize it against the new coordinator.
4. Restart the failover node normally, without the `-iqmpx_sn 1` switch.

Related Information

ALTER AGENT Statement
ALTER LS POLICY Statement
CREATE AGENT Statement
CREATE LS POLICY Statement
DROP AGENT Statement
Server Startup Utility (start_iq) [page 108]
SYSIQMPXSERVERAGENT System View
Coordinator Failure and Restart [page 51]

Synchronizing Multiplex Servers [page 42]

4.10.1.4 Synchronizing the Former Coordinator

Before starting a former coordinator, synchronize it against the new coordinator.

Prerequisites

- Requires the BACKUP DATABASE and SERVER OPERATOR system privileges.

Context

Perform the following steps on the computer where the former coordinator ran.

Procedure

1. On the former coordinator, switch to the directory where the copy of the catalog store will reside.
2. Copy the catalog store to the directory:

   Sample Code

   In this example, the credentials, port, engine are of the new coordinator. The folder is where copy of the catalog store will reside.

   On Windows:

   ```
   dbbackup -y -d -c "uid=<user_name>;pwd=<password>;links=tcpip{port=2957};
       eng=mpxnode_w1" -o btup.ot z:\mpxnode_r3
   ```

   On UNIX:

   ```
   dbbackup -y -d -c "uid=<user_name>;pwd=<password>;links=tcpip{port=2957};
       eng=mpxnode_w1" -o btup.ot '/mpxnode_r3'
   ```

3. (UNIX Only) If it exists, remove the transaction log files from the directory where the copy of the catalog store resides:

   Sample Code

   ```
   rm -rf mpxtest.log
   ```

4. Set the transaction log file path:

Sample Code

```
dblog -r -t mpxtest.log mpxtest.db
```

5. Start the former coordinator as a secondary node.

Sample Code

For Windows:

```
start_iq  z:\mpxnode_r31\mpxtest.db -n mpxnode_r3 -x tcpip{port=2763}
```

For UNIX:

```
start_iq  /mpxnode_r3/mpxtest.db -n mpxnode_r3 -x 'tcpip{port=2763}'
```

4.10.1.5 Coordinator Failure and Restart

If the coordinator restarts during a global transaction, due to shutdown, failover or server failure, transaction behavior depends on the user-defined timeout and the command being executed.

In the following cases, the communication to the coordinator resumes before timeout.

Command Status	Command Behavior	Result
Actively executing command	Command suspends and transfers control to server, except for ROLLBACK, which executes locally on writer.	Commands succeed
New DML command	Command suspends and transfers control to server. ROLLBACK and ROLLBACK TO SAVEPOINT execute locally instead of suspending.	Commands succeed

In the following cases, the communication failure exceeds the timeout.

Command Status	Command Behavior	Result
Suspended DML command on connection	The suspended command fails and returns an error.	Transaction rolls back
No suspended DML command on connection	The next command returns an error.	Transaction rolls back

For example, if a transaction suspends and the write server becomes inaccessible, you can roll back the transaction to release resources held by the coordinator for suspended transactions belonging to that writer.

4.10.2 Restarting a Secondary Server

The `start_iq` utility can restart secondary servers in a high-availability environment.

Procedure

At the command line, run the start server utility, `start_iq`, specifying the full path the location of the parameters:

```
start_iq @/<full-path-to-params>/params.cfg /<full-path-to-params>/database.db
```

For example:

```
start_iq @/system1/IQ16/mydb/params.cfg /system1/IQ16/mydb/database.db
```

> **Note**
>
> When restarting a secondary server in a high-availability environment, `<full-path-to-params>` is required.

4.10.3 SAN or Shared Disk Failure

These failures are the most serious because they can cause outage for all multiplex nodes.

Check the disk vendor support for high availability for this kind of failure.

4.11 Releasing Secondary Node Shared Temporary Space

Use MPX_SHTEMP_ALLOC_LEASE_TIME either alone, or with MPX_SHTEMP_INTSPACE_CLEAN, to improve multiplex performance.

The MPX_SHTEMP_ALLOC_LEASE_TIME database option lets you control precisely how long shared temporary dbspace memory block allocations are held before SAP IQ releases the last allocation unit from the allocation unit chain, freeing memory. The default lease time is 20 minutes, but you can specify a shorter lease time to improve query processing performance.

MPX_SHTEMP_INTSPACE_CLEAN offers a second means of releasing unused secondary node shared temporary dbspace allocations by freeing up even more space. When you set MPX_SHTEMP_INTSPACE_CLEAN to ON for the coordinator, SAP IQ scans the secondary node shared temporary dbspace allocation chains, and releases all unused shared temporary space (except the first allocation unit in the chain) back to the coordinator. Cleaning the shared temporary dbspace can improve query processing speeds.

Related Information

MPX_SHTEMP_ALLOC_LEASE_TIME Option
MPX_SHTEMP_INTSPACE_CLEAN Option

5 Manage Transactions

Multiplex transactions that modify shared objects behave according to certain rules.

Local Transactions

A local transaction does not modify a shared object.

A local transaction may be read-only or read-write but modify data in local objects only (SA tables or temp tables). Local transactions may be performed on any multiplex node, and the effects of the transaction are only visible to that node.

All multiplex transactions begin as local transactions.

Global Transactions

A global transaction modifies data in shared objects or changes the schema of any persistent object. Global transactions may only be performed on the coordinator node or a writer node. The effects of a global transaction are visible on all nodes of the multiplex.

A transaction only becomes global when the first read-write operation (such as an insert) is initiated on a shared IQ object. When a global transaction is initiated from a secondary writer node, the coordinator node must be running, because the writer node must communicate with the coordinator node to coordinate the global aspects of the transaction.

In a writer-initiated global transaction, the writer node CPU and local disks are used to do the work of the read-write operation, while the coordinator node handles the global bookkeeping for the transaction, such as the assignment of global transaction IDs, global lock management, and writing the transaction-level versioning (TLV) log.

In this section:

DML read-write transactions on multiplex writer nodes survive temporary communication failures between coordinator and writer nodes and temporary failure of the coordinator due to server failure, shutdown or failover.

Troubleshoot Transactions [page 72]
Commands are unaffected by many communication or coordinator failures, but certain cases require user action.

5.1 DML Commands

DML commands that modify tables in the shared IQ store are global DML commands. Any transaction that contains a global DML command becomes a global transaction.

You can run global transactions from the coordinator and any write server.

In this section:

Global DML Commands [page 55]
Global DML commands behave as if they were executed on the coordinator, and obey the same locking rules as on a single server.

Table Data Scope [page 55]
When running DML commands in multiplex, the visibility of the table rows differs for different table types.

5.1.1 Global DML Commands

Global DML commands behave as if they were executed on the coordinator, and obey the same locking rules as on a single server.

For example, if one transaction on any server has modified a shared IQ table, no other transaction may modify that table until the original transaction commits or rolls back. Whenever a global transaction commits, whether it runs on a writer node or the coordinator, the metadata for that global transaction is communicated to all servers in the multiplex through the TLV log.

5.1.2 Table Data Scope

When running DML commands in multiplex, the visibility of the table rows differs for different table types.

There are three types of row visibility in multiplex:

Global scope All connections on all servers can see the rows.

Server scope All connections on a particular multiplex server can see the rows.

Connection scope Only a single connection on a particular multiplex server can see the rows.

Table Type	Data Scope
IQ base	Global
IQ temporary	Connection
Global temporary table	Connection
SA catalog (table created IN SYSTEM)	Server
SA temporary (table created IN SYSTEM)	Connection

SAP IQ copies data in SA catalog tables on the coordinator node to the secondary node catalog store whenever a secondary node is synchronized. The data in catalog store tables is otherwise not related between secondary server and coordinator server instances. After synchronization, local SA table data on the secondary server is lost, because table data from the coordinator overwrites it.

> **Note**
>
> Because CIS proxy tables point to data on remote servers, the data scope is external. Data management for such tables is done by the remote server.

5.2 DDL Commands

Command type and object type affect the scope of Data Definition Language (DDL) commands in a multiplex.

Local Scope

Local commands execute on the local server and affect only the local catalog store or local temporary store.

Local commands affect these object types:

- Local temporary tables
- Local procedures
- Temporary options

Global Scope

Global commands execute on the coordinator and affect the shared IQ store and global catalog store. The coordinator writes statements with global scope to the TLV log on commit.

DDL commands that create, alter, or drop a permanent object ID in the `ISYSOBJECT` table are global.

Global commands affect these object types:

- Table – includes SA base tables, IQ base tables, and global temporary tables

- View
- Materialized view (SA tables only)
- Column
- Index
- Procedure
- Event
- User
- Publication
- Remote type
- Login mapping
- JAR
- Java class
- Service
- Database options (minus locally scoped option exceptions)

In this section:

5.2.1 DDL Command Dependencies Example 1

You cannot run global DDL commands if the coordinator node is not running. Attempting to do so results in the error `SQLCODE: -1004011, SQLSTATE QIA11: Coordinator node not responding`.

For example, if you created this temporary function or stored procedure:

```
CREATE TEMPORARY FUNCTION f1() RETURNS INT
BEGIN
RETURN 1;
END
```

Next, if you tried to create a view that is dependent on the temporary function:

```
CREATE VIEW v1 AS SELECT * FROM f1()
```

You would receive the error `Procedure 'f1' not found` since it is not a permanent procedure. SAP IQ does not allow such operations in a multiplex environment.

5.2.2 DDL Command Dependencies Example 2

When creating objects that are global, make sure that they do not depend on objects that are local.

This example creates a global object with a dependency on a local object. Assume that you create the `lineitem` temporary table on a secondary node:

```
DECLARE LOCAL TEMPORARY TABLE #lineitem (
  l_orderkey       integer,
  l_partkey        integer     iq unique(20000000),
  l_suppkey        integer     iq unique(20000000),
  l_linenumber     integer,
  l_quantity       integer     iq unique(50),
  l_extendedprice  double,
  l_discount       double      iq unique(11),
  l_tax            double      iq unique(9),
  l_returnflag     char(1)     iq unique(3),
  l_linestatus     char(1)     iq unique(2),
  l_shipdate       date        iq unique(270),
  l_commitdate     date        iq unique(256),
  l_receiptdate    date        iq unique(300),
  l_shipinstruct   char(25),
  l_shipmode       char(10)    iq unique(7),
  l_comment        char(44)
)
```

Next, you create indexes—which are global objects—on the columns of the `lineitem` temporary table using the `BEGIN PARALLEL IQ` command:

```
BEGIN PARALLEL IQ
CREATE HG INDEX LFIQ_IDX_TXXX_CXX_L_PK on #lineitem (l_partkey);
CREATE HG INDEX LFIQ_IDX_TXXX_CXX_L_OK on #lineitem (l_orderkey);
END PARALLEL IQ
```

SAP IQ returns the error `Table 'lineitem' not found` because the `BEGIN PARALLEL IQ` command is a global command sent to the coordinator node, but the `lineitem` table is a local temporary table on the secondary node.

5.2.3 Role Restriction

Some statements are restricted to nodes of certain roles.

The following statements run only on a coordinator node:

- All `CREATE/ALTER/DROP DBSPACE` commands operating on IQ main store dbspaces
- `BACKUP DATABASE`
- `sp_iqemptyfile`

The following statements run on a writer or coordinator node:

- `LOCK TABLE`
- All DDL commands that affect objects in the IQ main store dbspaces. This includes `ALTER/DROP` of:
  - Tables
  - Single and multicolumn indexes
  - Table constraints

Unlisted statements are unrestricted, and run on any node.

5.2.4 Preserving Rows

A global temporary table created with ON COMMIT PRESERVE ROWS cannot be altered or dropped from a secondary node if the instance of the global temporary table on the connection of the secondary node executing the drop table has data.

Truncate the table and try again or alter or drop the table from the coordinator node. If the global temporary table is created with ON COMMIT DELETE ROWS, you may alter or drop the table even if it has rows.

For example, connect to a secondary node:

```
CREATE GLOBAL TEMPORARY TABLE
foo_gtt(c1 int)
ON COMMIT PRESERVE ROWS;
INSERT INTO foo_gtt VALUES(200);
COMMIT;
DROP TABLE foo_gtt;
```

The drop statement fails with the following error:

```
Operation not allowed on global temporary
table foo_gtt as it is in use. Please reconnect and
try again. SQLCODE=1287. ODBC 3
State="HY000"
```

5.3 Dbspace Updates in Multiplex

Dbspace updates on multiplex differ slightly from those on SAP IQ databases.

In general, when updating shared dbspaces on a multiplex:

- Only the coordinator can manipulate shared IQ dbspaces.
- Before you can drop a shared dbspace file, you must alter the file to read-only mode. All servers must be at the version corresponding to the `ALTER READONLY` statement, and must have returned all reserved blocks on the file.
- `CREATE DBSPACE`, `ALTER DBSPACE`, and `DROP DBSPACE` commands for main and catalog dbspaces are allowed only on the coordinator.

Due to TLV propagation and version maintenance, there may be a delay of a few minutes before you can drop an empty dbfile or dbspace. You see a `Command not replayed` error. To perform the `DROP DBSPACE`

or `ALTER DBSPACE DROP FILE` commands, the `OkToDrop` column reported by the `sp_iqdbspace` and `sp_iqfile` procedures must indicate Y.

Start the coordinator in single-node mode (`-iqmpx_sn -1`) before dropping files from IQ_SHARED_TEMP. The first file made read-write in IQ_SHARED_TEMP must be the last file dropped. You may also drop files in IQ_SHARED_TEMP in an SAP IQ server.

> **Note**
>
> Do not start the coordinator in single-node mode with `-iqro`, or adding or dropping a file returns the error `SQL Anywhere Error -757: Modifications not permitted for read-only database`.

In this section:

5.3.1 Updates on IQ_SYSTEM_TEMP

Familiarize yourself with requirements before updating IQ_SYSTEM_TEMP dbspaces.

When updating IQ_SYSTEM_TEMP:

- Create, alter, and drop temporary dbspace files only from the node where they reside. The syntax for these commands is the same as for SAP IQ temporary dbspaces.
- Secondary servers must be synchronized before you run `ALTER DBSPACE ALTER FILE RENAME PATH`. Synchronization happens automatically through TLV log replay, but there may be a time delay of up to two minutes between the previous dbspace operation on this dbspace (create or alter) and when you can run `ALTER DBSPACE ALTER FILE RENAME PATH`. If secondary servers are not synchronized, you may see a "Command not replayed" error.

Before updating dbspaces, see the overview of dbspaces and dbfiles in *Data Storage* in SAP IQ Administration: Database.

5.3.2 Updates on IQ_SYSTEM_MAIN

The IQ_SYSTEM_MAIN dbspace manages important database structures including the free list, which lists blocks in use.

Before adding space to IQ_SYSTEM_MAIN, shut down secondary nodes. Active secondary nodes shut down automatically if a dbfile is added to IQ_SYSTEM_MAIN. The IQ message file for the secondary node reports:

```
Multiplex secondary node shutting down
due to a file added to the IQ_SYSTEM_MAIN dbspace.
This node must be synchronized and restarted.
```

This behavior applies only to the IQ_SYSTEM_MAIN dbspace. Other dbspace operations cause no disruption and all nodes in the multiplex continue to run.

When updating IQ_SYSTEM_MAIN:

- If any shared IQ main dbspace files have paths inaccessible from a secondary node, the secondary node cannot access the file or any contents of that file until the path is corrected.
- `ALTER DBSPACE ALTER FILE RENAME PATH` is prohibited on IQ_SYSTEM_MAIN.

After adding new files to IQ_SYSTEM_MAIN, synchronize and restart secondary nodes.

In this section:

Related Information

Shared Store Path Requirement [page 10]
Adding Space to IQ_SYSTEM_MAIN on a Coordinator [page 64]
Synchronizing Multiplex Servers [page 42]

5.3.2.1 Changing an IQ_SYSTEM_MAIN File Path

If the paths of all dbfiles in IQ_SYSTEM_MAIN are not visible to the secondary nodes, the secondary nodes do not start, and report `Error opening DBFILE 'filepath'` messages.

Prerequisites

- Requires the MANAGE ANY DBSPACE system privilege.
- At least two dbfiles exist in IQ_SYSTEM_MAIN.

Context

The IQ_SYSTEM_MAIN dbspace manages important database structures including the freelist, which tracks which blocks are in use. Shut down secondary nodes before adding space to IQ_SYSTEM_MAIN. If a dbfile is added to IQ_SYSTEM_MAIN, all running secondary nodes shut down automatically and the IQ message file for the secondary node reports:

```
Multiplex secondary node shutting down
due to a file added to the IQ_SYSTEM_MAIN dbspace.
This node must be synchronized and restarted.
```

Changing the path requires that the IQ_SYSTEM_MAIN dbfile be read-only. Since at least one dbfile must always be read-write, you need at least two dbfiles to complete this task.

Procedure

1. Shut down all the servers in the multiplex.
2. Start the coordinator in single node mode using the `-iqmpx_sn 1` switch.

   Sample Code

   On Windows:

   ```
   start_iq z:\mpxnode_c1 -n mpxnode_c1 -x tcpip{port=2764} mpxtest.db
   ```

   On UNIX:

   ```
   start_iq /mpxnode_c1 -n mpxnode_c1 -x tcpip{port=2764} mpxtest.db
   ```

3. Verify that you have at least two dbfiles in `IQ_SYSTEM_MAIN` by executing `sp_iqfile`.

DBSpaceName	DBFileName	Path	SegmentType	RWMode
IQ_SYSTEM_MAIN	IQ_SYSTEM_MAIN	z:\mpxnode_c1\mydb1.iq	MAIN	RW
IQ_SYSTEM_MAIN	myfile1	z:\mpxnode_c1\myfile1.iq	MAIN	RO
IQ_SYSTEM_TEMP	IQ_SYSTEM_TEMP	mydb1.iqtmp	TEMPORARY	RW

4. If only one dbfile exists in IQ_SYSTEM_MAIN, create a second dbfile and make it read-write.

   Sample Code

   ```
   ALTER DBSPACE IQ_SYSTEM_MAIN ADD FILE myfile2 'z:\mpxnode_c1\myfile2.iq'
   SIZW 200 MB;
   ```

   ```
   ALTER DBSPACE IQ_SYSTEM_MAIN ALTER FILE myfile2 READWRITE;
   ```

5. If multiple dbfiles exist beside IQ_SYSTEM_MAIN, ensure that at last one is read-write.
6. Set the IQ_SYSTEM_MAIN dbfile to READONLY.

   Sample Code

   ```
   ALTER DBSPACE IQ_SYSTEM_MAIN ALTER FILE IQ_SYSTEM_MAIN READWRITE
   ```

7. Empty the dbfile.

   ```
   sp_iqemptyfile IQ_SYSTEM_MAIN
   ```

8. Drop the IQ_SYSTEM_MAIN dbfile.

   ```
   ALTER DBSPACE IQ_SYSTEM_MAIN DROP FILE IQ_SYSTEM_MAIN
   ```

9. Add the IQ_SYSTEM_MAIN dbfile using the new path visible to all secondary nodes. Specify both the logical name (IQ_SYSTEM_MAIN) and the physical file name ending in `.iq`.

   Sample Code

   For Windows:

   ```
   ALTER DBSPACE IQ_SYSTEM_MAIN ADD FILE IQ_SYSTEM_MAIN
   'y:\mpxnode_c1\mydb1.iq'
   ```

   For UNIX:

   ```
   ALTER DBSPACE IQ_SYSTEM_MAIN ADD FILE IQ_SYSTEM_MAIN '/mpxnode_c1/mydb1.iq'
   ```

10. Set the IQ_SYSTEM_MAIN dbfile back to read-write.

    ```
    ALTER DBSPACE IQ_SYSTEM_MAIN ALTER FILE IQ_SYSTEM_MAIN READWRITE
    ```

11. If necessary, drop the dbfile create in step 4.
12. Restart the coordinator normally, without the `-iqmpx_sn 1` switch.
13. Start all secondary nodes and synchronize. See Synchronizing Multiplex Servers [page 42].

5.3.2.2 Adding Space to IQ_SYSTEM_MAIN on a Coordinator

If the coordinator runs out of space in IQ_SYSTEM_MAIN, including reserve space, it may abort to prevent database corruption; you may be unable to start the coordinator normally.

Prerequisites

- Requires the MANAGE ANY DBSPACE system privilege.

Context

When the coordinator aborts due to lack of space, start the coordinator in a single node mode using the `-iqmpx_sn` startup switch, and add more space by adding file(s) to IQ_SYSTEM_MAIN. Restart the coordinator normally and synchronize all secondary nodes, as in the following procedure.

Perform these steps when the coordinator needs space in IQ_SYSTEM_MAIN.

Procedure

1. Shut down all servers in the multiplex.
2. Start the coordinator in single node mode using `-iqmpx_sn 1`.

   Sample Code

   On Windows:

   ```
   start_iq z:\mpxnode_c1\mytest.db -iqmpx_sn 1 -n mpxnode_c -x
   tcpip{port=<2763>}
   ```

   On UNIX

   ```
   start_iq /mpxnode_c1/mytest.db -iqmpx_sn 1 -n mpxnode_c -x
   "tcpip{port=<2763>}"
   ```

3. To add more space to IQ_SYSTEM_MAIN with a new file on a raw device, use syntax like the following, where `mymainfile` is the logical or chosen logical name of the new dbfile and `PhysicalDrive3.iq` is the physical file name.

   Sample Code

   For Windows:

   ```
   ALTER DBSPACE IQ_SYSTEM_MAIN ADD FILE <mymainfile> '\\.\PhysicalDrive3.iq'
   ```

For UNIX:

```
ALTER DBSPACE IQ_SYSTEM_MAIN ADD FILE <mymainfile> '/dev/rdsk/
PhysicalDrive3.iq'
```

> **Note**
>
> Raw devices aren't available in Linux. See SAP Note 3483103 The RAW_DRIVER, and support for devices located in `/dev/raw/`, was deprecated in:
>
> - SuSE Linux Enterprise Server (SLES) 15.4
> - Red Hat Enterprise Linux 9

4. Restart the coordinator normally, without the `-iqmpx_sn 1` switch.
5. Restart the secondary nodes and synchronize.

Related Information

Synchronizing Multiplex Servers [page 42]

5.3.2.3 Replacing Dbfiles in IQ_SYSTEM_TEMP on a Coordinator

To replace a damaged dbfile, drop the file, restart the server, and add the dbfile in the coordinator.

Prerequisites

- Requires the MANAGE ANY DBSPACE system privilege.

Procedure

1. Use the `-iqnotemp` parameter to start the database.

   The only temporary file operation allowed on a database while running with `-iqnotemp` is to drop one or more of the temporary file definitions.
2. Drop the last file in IQ_SYSTEM_TEMP.

   ```
   ALTER DBSPACE IQ_SYSTEM_TEMP DROP FILE <filename>
   ```
3. Stop the database.
4. Start the database in single-node mode with no files to empty the checkpoint log.

Sample Code

On Windows:

```
start_iq z:\mpxnode_c1\mytest.db -iqmpx_ov 1 -iqmpx_sn 1 -n mpxnode_c -x
tcpip{port=<2763>}
```

On UNIX

```
start_iq /mpxnode_c1/mytest.db -iqmpx_ov 1 -iqmpx_sn 1 -n mpxnode_c -x
"tcpip{port=<2763>}"
```

5. Add a "dummy" dbfile to the coordinator. You'll need to drop this file and add it again in coordinator mode, because files added in single-node mode have a null server ID; they're owned by the coordinator instead of the server that added the file. To add more space to IQ_SYSTEM_MAIN with a new file on a raw device, use syntax like the following, where `<tempfile>` is the logical or chosen logical name of the new dbfile.

Note

Raw devices aren't available in Linux. See SAP Note 3483103 The RAW_DRIVER, and support for devices located in `/dev/raw/`, was deprecated in:

- SuSE Linux Enterprise Server (SLES) 15.4
- Red Hat Enterprise Linux 9

Sample Code

For Windows:

```
ALTER DBSPACE IQ_SYSTEM_TEMP ADD FILE <tempfile> '\\.\PhysicalDrive3'
```

For UNIX:

```
ALTER DBSPACE IQ_SYSTEM_TEMP ADD FILE <tempfile> '/dev/rdsk/
c4t600A0B80005A7F5D0000024'
```

6. Stop and restart the server.
7. Drop the dummy file.

5.3.3 Updates on IQ_SHARED_TEMP

When you update IQ_SHARED_TEMP dbspaces, certain restrictions apply.

These rules affect IQ_SHARED_TEMP dbspace updates:

- Only the coordinator can manipulate shared IQ dbspaces.
- Start the coordinator in single-node mode before dropping files from IQ_SHARED_TEMP. The first file made read-write in IQ_SHARED_TEMP must be the last file dropped. You may also drop files in IQ_SHARED_TEMP in an SAP IQ server.
- Updates on the IQ_SHARED_TEMP store require the MANAGE ANY DBSPACE system privilege.
- You cannot execute `ALTER FILE RENAME PATH` on the IQ_SHARED_TEMP dbspace.

- You cannot execute `ALTER DBSPACE ALTER FILE READONLY` on the IQ_SHARED_TEMP dbspace.

> **Note**
>
> If you start the coordinator in single-node mode with `-iqro`, adding or dropping a file returns the error `SQL Anywhere Error -757: Modifications not permitted for read-only database`.

In this section:

5.3.3.1 Shared File Status Data

When opening databases and executing certain commands, multiplex nodes maintain a file status for each shared dbspace file.

The coordinator maintains status data for all nodes, and each secondary node maintains its own file status data.

All included servers must return valid read-write status for a new IQ_SHARED_TEMP file, or the statement returns an error. If all nodes can access the newly added file, they return a `Valid status` message. All secondary servers return the file status for all shared files to the coordinator. SAP IQ maintains shared file status information for dbfiles of all shared dbspaces, but uses the information only in dbspace update commands that target the IQ_SHARED_TEMP dbspace.

5.3.3.2 Adding Dbfiles to Shared Dbspaces

Use `ALTER DATABASE ADD FILE` statement in Interactive SQL to add dbfiles to shared dbspaces.

Prerequisites

- Requires the MANAGE ANY DBSPACE system privilege.

Procedure

1. Start Interactive SQL and connect to the coordinator.

> ⚠ Caution
>
> Don't start the coordinator in single-node mode (`-iqmpx_sn 1`) with `-iqro`, or adding a file returns the error `SQL Anywhere Error -757: Modifications not permitted for read-only database`.

```
dbisql
```

2. Add the file. In the `ADD FILE` clause, specify either a full path to a raw device, or a soft link. You need not synchronize or restart any secondary servers.

> ⓘ Note
>
> Raw devices aren't available in Linux. See SAP Note 3483103 The RAW_DRIVER, and support for devices located in `/dev/raw/`, was deprecated in:
>
> - SuSE Linux Enterprise Server (SLES) 15.4
> - Red Hat Enterprise Linux 9

> {·} Sample Code
>
> For Windows:
>
> ```
> ALTER DBSPACE IQ_SHARED_TEMP ADD FILE mydbfilename
> '\\\\.\\PhysicalDrive2'
> ```
>
> The same command, using a soft link:
>
> ```
> ALTER DBSPACE IQ_SHARED_TEMP ADD FILE mydbfilename
> '/dev/rdsk/c4t600A0B80005A7F5D0000024'
> ```
>
> For UNIX:
>
> ```
> ALTER DBSPACE IQ_SHARED_TEMP ADD FILE mydbfilename
> 'store/userdb1'
> ```

3. Confirm that the file is visible to all secondary nodes.

```
sp_iqmpxfilestatus
```

4. Shared files on IQ_SYSTEM_MAIN and user main dbspaces are implicitly read-write. Dbfiles on IQ_SHARED_TEMP are created as read only. If the dbfile is on a shared temporary dbspace, alter the new file to read-write status.

> {·} Sample Code
>
> ```
> ALTER DBSPACE IQ_SHARED_TEMP ALTER FILE mydbfilename READWRITE
> ```

Results

If an error is returned regarding the file status issues from one or more nodes, run the `sp_iqmpxfilestatus` procedure to troubleshoot and correct problems. You can force read-write status on IQ_SHARED_TEMP dbfiles if you need to postpone problem correction:

Sample Code

```
ALTER DBSPACE IQ_SHARED_TEMP ALTER FILE mydbfilename FORCE READWRITE
```

Use of the `FORCE READWRITE` clause returns an error on IQ_SYSTEM_MAIN and user main dbfiles.

5.3.4 Adding a Catalog Dbspace

Under normal circumstances, you don't need to add catalog dbspaces; they automatically grow as needed. However, if your catalog dbspace file is on constrained storage, you may need to add a new catalog dbspace to accommodate catalog data.

Prerequisites

- Requires the MANAGE ANY DBSPACE system privilege.

Procedure

1. Shut down all servers in the multiplex.
2. Start the coordinator in single node mode using `-iqmpx_sn 1`.

   Sample Code

   On Windows:

   ```
   start_iq z:\mpxnode_c1\mytest.db -iqmpx_sn 1 -n mpxnode_c -x
   tcpip{port=<2763>}
   ```

   On UNIX

   ```
   start_iq /mpxnode_c1/mytest.db -iqmpx_sn 1 -n mpxnode_c -x
   "tcpip{port=<2763>}"
   ```
3. Create the dbspace, where`dspcat2` is the logical filename and `sadb2.db` is the physical file name.

   Sample Code

   ```
   CREATE DBSPACE dspcat2 AS 'sadb2.db' CATALOG STORE
   ```

4. Restart the coordinator without the `-iqmpx_sn 1` switch.
5. Restart and synchronize all secondary servers in the multiplex.

Related Information

Dropping a Catalog Dbspace [page 70]
Synchronizing Multiplex Servers [page 42]

5.3.5 Dropping a Catalog Dbspace

Under normal circumstances, you don't need to remove catalog dbspaces.

Prerequisites

- Requires the MANAGE ANY DBSPACE system privilege.

Procedure

1. Shut down all the servers in the multiplex.
2. Start the coordinator in single node mode using the `-gm` and `-iqmpx_sn 1` switches.

   Sample Code

   For Windows:

   ```
   start_iq z:\mpxnode_c1 -iqmpx_sn 1 -gm -n mpxnode_c1 -x tcpip{port=2764}
   mpxtest.db
   ```

   For UNIX:

   ```
   start_iq /mpxnode_c1 -iqmpx_sn 1 -gm -n mpxnode_c1 -x tcpip{port=2764}
   mpxtest.db
   ```

3. Drop the dbspace.

   Sample Code

   ```
   DROP DBSPACE dspcat2
   ```

4. Restart the coordinator normally, without the `-gm` or `-iqmpx_sn 1` switches.
5. Restart and synchronize all secondary servers in the multiplex.

Related Information

Dropping a Catalog Dbspace [page 70]
Adding a Catalog Dbspace [page 69]
Adding a Catalog Dbspace [page 69]

5.4 Global Transaction Resiliency

DML read-write transactions on multiplex writer nodes survive temporary communication failures between coordinator and writer nodes and temporary failure of the coordinator due to server failure, shutdown or failover.

When a user connects to a writer node and executes read-write DML commands against shared objects, the writer starts a global transaction on the coordinator. The transaction starts on an internal internode communication (INC) connection from writer to coordinator.

For example, INSERT or LOAD commands on shared database objects are global transactions. If a failure occurs, the global transaction and corresponding INC connection is suspended.

If the temporary failure resolves within a user-defined timeout period, the global transaction continues as if there was no failure. The user can commit, roll back, or continue the transaction. Use the MPX_LIVENESS_TIMEOUT option to set the timeout period, which defaults to an hour (default value 3600).

If the failure persists longer than the user-defined timeout period, the global transaction cannot resume and the user must roll back the whole transaction.

If there is a DML command actively executing while the failure happens, the command behavior depends on the user-defined timeout and the command type.

To check connection status (active or suspended), use the `sp_iqconnection` system procedure on a writer node or `sp_iqmpxsuspendedconninfo` system procedure on a coordinator. Run `sp_iqmpxincstatistics` for a snapshot of the aggregate statistics of the INC status since server startup.

This feature does not affect transactions initiated on the coordinator.

Related Information

ALTER LS POLICY Statement
DROP MULTIPLEX SERVER Statement
Dropping a Secondary Server from a Multiplex [page 35]
MPX_LIVENESS_TIMEOUT Option
sp_iqconnection Procedure
sp_iqmpxincstatistics Procedure
sp_iqmpxsuspendedconninfo Procedure
Troubleshoot Transactions [page 72]

5.5 Troubleshoot Transactions

Commands are unaffected by many communication or coordinator failures, but certain cases require user action.

In this section:

5.5.1 Communication Failure or Coordinator Failure and Restart During Global Transaction

If internode communication (INC) fails or the coordinator fails or is shut down during a writer-initiated global transaction, transactions suspend and resume automatically if the INC is restored before a user-specified timeout expires.

Delays in command execution may indicate INC suspend and resume operations. If INC is interrupted, the coordinator suspends a global transaction for an hour. The transaction resumes successfully as soon as INC is

restored. If the timeout value elapses, the transaction fails. Set the `MPX_LIVENESS_TIMEOUT` database option to change the timeout period.

The following cases describe the behavior of writer nodes if the communication to the coordinator resumes before timeout.

Writer Command Status	Command Behavior	Result
Actively executing command	Command suspends, except for ROLLBACK, which executes locally on writer.	Command succeeds.
New DML command	Command suspends and resumes, except for ROLLBACK and ROLLBACK TO SAVEPOINT, which execute locally on the writer.	If communication is restored, resumed commands succeed.

The following cases describe the behavior of writer nodes if the communication failure exceeds the timeout period.

Writer Command Status	Command Behavior	Result
Suspended DML command on connection	The suspended command fails and returns an error about the non-recoverable state of the transaction.	You must roll back the transaction. Rollback happens automatically if the suspended command is COMMIT or ROLLBACK to SAVEPOINT.
No suspended DML command on connection	The next command returns an error about the non-recoverable state of the transaction.	You must roll back the transaction.

To check connection status, use the `sp_iqconnection` system procedure on a writer node or the `sp_iqmpxsuspendedconninfo` system procedure on a coordinator.

Run `sp_iqmpxincstatistics` for a snapshot of the aggregate statistics of the INC status since server startup.

> **Note**
>
> If a global transaction initiated from a writer node modifies both global and local persistent objects (for example, an SA base table and an IQ base table), and the coordinator fails during commit, global object changes may be committed while local object changes are lost. This is consistent with a scenario that updates both local and proxy tables in the same transaction, where “best effort” is used to commit both local and global components of a transaction.

Related Information

Coordinator Failover and Global Transactions [page 74]

5.5.2 Reclaiming Resources Held for Global Transactions

If a writer node with one or more active global transactions becomes inaccessible, the coordinator does not immediately release resources held for global transactions.

Context

The coordinator waits for a period of `2 * Mpx_Liveness_Timeout` for the writer to resume the global transaction. If the writer does not resume by the end of the period, the coordinator releases resources. To free the resources held for the global transactions immediately, kill the corresponding suspended INC connections on the coordinator.

Procedure

1. Run the stored procedure `sp_iqmpxsuspendedconninfo` to locate the suspended connection:

   ```
   call sp_iqmpxsuspendedconninfo
   ```

   ```
   ConnName             ConnHandle      GlobalTxnId
   ==================   =============   =============
   'IQ_MPX_SERVER_P5'               15            3920
   MPXServerName        TimeInSuspendedState
   ==================   ======================
   'HP1_12356_IQ_mpx'                     50
   SuspendTimeout
   ==============
             7200
   ```

2. On the server specified in `MPXServerName`, issue a drop command that specifies the `ConnHandle` for the connection:

   ```
   call "DROP CONNECTION 15"
   ```

5.5.3 Coordinator Failover and Global Transactions

If the coordinator fails over during a writer-initiated global transaction before a user-specified timeout expires, transactions suspend and resume automatically.

Delays in command execution may indicate internode communication (INC) suspend and resume operations.

Related Information

Communication Failure or Coordinator Failure and Restart During Global Transaction [page 72]

5.5.4 Dynamic Collisions

A dynamic collision occurs when a schema change is committed on a table, procedure, function, or view while a secondary server is querying the same object. The collision results from the TLV replay of shared IQ object schema updates.

Dynamic collisions only occur on secondary nodes. In an interactive setting, a dynamic collision results in a forced disconnect of a user connection from a secondary node.

To resolve a dynamic collision during DDL execution, the secondary server finds the connections responsible and disconnects them. The disconnect is logged in the `.iqmsg` file with a message similar to:

```
Shared IQ Store update DDL statement:
drop table DBA.gtt44
Disposition: SQLSTATE:42W21 --
dropped 1 connection(s) for table:
DBA.gtt44 Retry successful
```

To avoid dynamic collisions, schedule schema changes when the multiplex is not heavily loaded.

5.5.5 Resolving INC Failure

If Internode Communication (INC) between the coordinator and a secondary node has been disrupted and returns an error, you may need to drop the connection before retrying the DML command.

Context

A DML command that runs from a writer processes as a global transaction on a coordinator via INC. After the coordinator is restarted, a DML command that starts a global transaction from the same connection on a writer may fail with the following error before INC resumes:

Sample Code

```
I. 12/20 10:12:36. 0000228231 Exception Thrown from dblib/
db_catalog.cxx:824, Err# 5, tid 5 origtid 5
I. 12/20 10:12:36. 0000228231    O/S Err#: 0, ErrID: 12291
(inc_CommandHandlerException);
 SQLCode: -1006274, SQLState: 'QBC74', Severity: 14
I. 12/20 10:12:36. 0000228231 [22036]:
Transaction cannot continue due to unrecoverable INC failure.
Roll back current transaction.
--dblib/db_catalog.xx 824
```

After INC resumes, DML commands on the writer may continue to return the same error. If you encounter this problem, proceed as follows.

Procedure

Drop the connection using one of the following:

Option	Description
`DROP CONNECTION`	Drops the client connection that returned the error, or exit from the client that was connected to the writer.

`sp_iqconnection`	Locates a connection on the coordinator that matches the following condition, and drop that connection.
	Sample Code
	`Name : IQ_MPX_SERVER_Pxxx`
	`Reqtype : EXEC`
	`LastIQCmdTime : 0001-01-01 00.00.00.0`
	`ConnCreateTime : Time before ConnCreateTime of IQ_MPX_SERVER_H0`

5.5.6 Dropping Global Transactions After Timeout

On rare occasions, the coordinator may fail during a global transaction and restart after the MPX_LIVENESS_TIMEOUT period has elapsed.

Context

Before the writer's connection can perform new write operations, you must locate and drop the old global transaction started on the writer before the coordinator failed.

Procedure

1. To locate the transaction, run `sp_iqconnection` on the coordinator and find the ConnHandle of the connection with a name like IQ_MPX_SERVER_PXXX.
2. To drop the old transaction, enter a `DROP CONNECTION <ConnHandle>` command on the coordinator.

5.5.7 Secondary Server Shuts Down After CREATE TEXT INDEX

A secondary server may perform an emergency shutdown after running CREATE TEXT INDEX.

An emergency shutdown happens under these conditions:

- The secondary server is started with the `-sf external_library_full_text` flag, which disables the loading of external libraries
- The user issues a `CREATE TEXT INDEX` statement on the coordinator with a text configuration that uses external libraries.

To avoid this, do not use the `-sf external_library_full_text` flag when starting secondary nodes in the multiplex.

5.5.8 Physical File Not Deleted by DROP DBSPACE

Within a multiplex environment on Windows, if you execute `DROP DBSPACE <dbspace_name>` on a coordinator, the dbspace file may not be physically removed, if the file is locked by the secondary nodes.

No error is reported to the client when this occurs. The IQ message file (`.iqmsg`) of the coordinator records this error:

```
I. 02/25 11:20:58. 0000000060 DropDBSpacePhysicalFile
I. 02/25 11:20:58. 0000000060 Exception Thrown from
hos_bio.cxx:1228, Err# 6, tid 398 origtid 398
I. 02/25 11:20:58. 0000000060  O/S Err#: 26, ErrID:
518 (hos_bioexception); SQLCode: -1006062, SQLState:
'sQBA62', Severity: 14
I. 02/25 11:20:58. 0000000060 [20300]: OS error 26
reported on file C:\users\mpx_tablespaces\mpx\size_dbsp_0tb.
```

To resolve this issue, manually delete the file from the file system after dropping the dbspace.

6 Manage Resources Through Logical Servers

You must use logical servers to access multiplex servers.

Logical servers provide resource provisioning for the IQ multiplex by grouping a subset of computing resources (multiplex servers) as a logical entity.

When you connect to a logical server and run a query, SAP IQ distributes the query execution to only those nodes that are members of the logical server. You can dynamically add or drop member nodes for a logical server to accommodate the changing resource needs of applications.

Access to SAP IQ multiplex nodes is provided only via logical servers. You can create multiple logical servers to serve different groups of applications or users. When you create a logical server, explicitly select one or more multiplex servers to add them as members. A multiplex node can be a member of more than one logical server. A logical server can have both reader and writer nodes. You can also make the coordinator node a member of a logical server, although specific rules govern coordinator membership.

A workload directed to a logical server can only be processed by member servers. Only queries that can be parallelized are eligible for work distribution; all other operations are confined to the node of initial connection. Users must connect to a node that has the role (reader or writer) required for the intended operation.

> **Note**
>
> SERVER is the default logical server for `dbbackup` and `dbstop` commands.

Not all member nodes of a logical server may be available at all times due to failure or exclusion from the multiplex.

The effective logical server configuration represents the current dynamic constitution of the logical server consisting of all member nodes that are actually available for use. The effective logical server configuration is based on the static logical server configuration and dynamic state of the multiplex.

In this section:

To simplify administration, comment on user-defined logical servers.

Altering Membership of a Logical Server [page 87]
Alter the membership or login policy of a user-defined logical server using Interactive SQL.

Dropping a Logical Server [page 87]
Drop a user-defined logical server using Interactive SQL.

Connecting to a Logical Server [page 88]
Use the `LogicalServer` and `NodeType` connection parameters to establish the logical server context for a new user connection.

Configure Logical Server Policies [page 89]
A logical server policy is associated with each logical server. Configure logical server policy options to control behavior of all associated logical servers.

Altering Root Logical Server Policy [page 90]
Alter the root logical server policy of a user-defined logical server using Interactive SQL.

Manage Logical Server Membership [page 91]
You can only access multiplex servers by using logical servers. Login policies control user access to the logical servers.

Redirecting Logins [page 94]
SAP IQ provides load balancing when a user tries to log in to an overloaded node by redirecting the attempted login to a node that is less loaded in the same logical server.

Disabling Login Redirection [page 96]
You can disable login redirection for all logical servers governed by a named logical server policy, or at the connection level.

Login Policies [page 96]
A login policy defines the rules that SAP IQ follows to establish user connections. Each login policy is associated with a set of options called login policy options.

6.1 Logical Servers and SAP IQ Configurations

Only multiplexes support logical servers, but information about built-in logical servers and logical server policies can remain, unused, in the catalog in an SAP IQ server environment.

In this section:

Logical Servers and New SAP IQ Databases [page 80]
When you create a new SAP IQ database, catalogs are created with the information for built-in logical servers ALL, AUTO, COORDINATOR, NONE, OPEN, and SERVER.

Multiplex Database Conversion into an SAP IQ Database [page 80]
Before converting a multiplex into an SAP IQ database, drop all user-defined logical servers.

Connection in an SAP IQ Database [page 80]
Connecting to an SAP IQ database is unaffected by the login policy setting of logical server assignments. The connections have no logical server context.

6.1.1 Logical Servers and New SAP IQ Databases

When you create a new SAP IQ database, catalogs are created with the information for built-in logical servers ALL, AUTO, COORDINATOR, NONE, OPEN, and SERVER.

The root logical server policy is also automatically created. The OPEN logical server is assigned to the root login policy.

> **Note**
>
> Executing DDL commands to create a new logical server returns an error.

6.1.2 Multiplex Database Conversion into an SAP IQ Database

Before converting a multiplex into an SAP IQ database, drop all user-defined logical servers.

Use the `DROP MULTIPLEX SERVER` command with the WITH DROP LOGICAL SERVER clause to drop all user-defined logical servers.

The catalog retains this information:

- Built-in logical servers
- Root logical server policy
- Login policy assignments to built-in logical servers
- Logical server level overrides for the login policy option. These could exist only for the built-in logical servers.

The information is reused if you convert the SAP IQ database back to multiplex.

6.1.3 Connection in an SAP IQ Database

Connecting to an SAP IQ database is unaffected by the login policy setting of logical server assignments. The connections have no logical server context.

The login policy `locked` option is still applied before connections are accepted by the server.

6.2 Coordinator as a Logical Server Member

Logical server memberships that are defined using the multiplex server name of the current coordinator server are ineffective.

These memberships become effective again when the multiplex server no longer acts as the current coordinator.

In this section:

6.2.1 Logical Membership of the Coordinator

To specify logical membership for the coordinator in a user-defined logical server, use the `FOR LOGICAL COORDINATOR` clause instead of the name of the current coordinator.

The logical membership of the coordinator means the coordinator is always available to the logical server as its member, regardless of which multiplex node plays the coordinator role.

> **Note**
>
> The coordinator role can move from one multiplex server to another, for example, during a failover. If a logical server has logical membership of the coordinator, the new coordinator server becomes part of the effective configuration of the logical server.

The current coordinator node is available only to those logical servers that have the coordinator as its logical member, and not to those logical servers that have membership to the current coordinator node using the multiplex node name. If the current coordinator is both a logical member and a named member of a logical server, then it is available to the logical server, but only by virtue of its logical membership. The named membership is still considered ineffective.

The coordinator node membership rules ensure predictability of the workload on the coordinator. The coordinator is available to a known set of logical servers and that does not change as the coordinator fails over from one multiplex node to another.

6.3 Use Built-in Logical Servers

OPEN and SERVER are built-in logical servers that always exist; their membership is implicitly defined, and is never stored in the catalog.

In this section:

OPEN Logical Server [page 84]
A node that is not a part of any user-defined logical server is implicitly a member of the OPEN logical server. The OPEN logical server consists of all multiplex nodes that are not members of any user-defined logical servers.

SERVER Logical Server [page 84]
On each multiplex server, the SERVER logical server has implicit membership to that server only, allowing certain privileged users to connect to any node of the multiplex and execute server-level administrative tasks without knowing which logical servers include that node.

6.3.1 ALL Logical Server

The ALL logical server allows access to all logical servers.

When you specify the ALL logical server, there is no need to list the names as you add or drop servers.

Parent topic: Use Built-in Logical Servers [page 81]

Related Information

AUTO Logical Server [page 82]
COORDINATOR Logical Server [page 83]
NONE Logical Server [page 83]
OPEN Logical Server [page 84]
SERVER Logical Server [page 84]

6.3.2 AUTO Logical Server

Specify the AUTO logical server to prevent login redirection.

If the node belongs to multiple logical servers, using the AUTO logical server returns an error. If the node belongs to a single logical server, AUTO uses the logical server context of the current node.

Parent topic: Use Built-in Logical Servers [page 81]

Related Information

ALL Logical Server [page 82]
COORDINATOR Logical Server [page 83]
NONE Logical Server [page 83]

OPEN Logical Server [page 84]
SERVER Logical Server [page 84]

6.3.3 COORDINATOR Logical Server

COORDINATOR is a built-in logical server that consists of the current coordinator node.

Clients can use the COORDINATOR logical server to connect to the current coordinator without knowing its name. This simplifies connection because the coordinator role may pass from one node to another, for example, during failover.

You cannot drop the COORDINATOR logical server.

Parent topic: Use Built-in Logical Servers [page 81]

Related Information

ALL Logical Server [page 82]
AUTO Logical Server [page 82]
NONE Logical Server [page 83]
OPEN Logical Server [page 84]
SERVER Logical Server [page 84]

6.3.4 NONE Logical Server

The NONE logical server is defined to be always empty.

Parent topic: Use Built-in Logical Servers [page 81]

Related Information

ALL Logical Server [page 82]
AUTO Logical Server [page 82]
COORDINATOR Logical Server [page 83]
OPEN Logical Server [page 84]
SERVER Logical Server [page 84]

6.3.5 OPEN Logical Server

A node that is not a part of any user-defined logical server is implicitly a member of the OPEN logical server. The OPEN logical server consists of all multiplex nodes that are not members of any user-defined logical servers.

You cannot directly manipulate membership of a multiplex node into the OPEN logical server; it changes implicitly when you add multiplex nodes to, or drop them from, user-defined logical servers. The OPEN logical server may be empty when each server of a multiplex is a member of one or more logical servers.

> **Note**
>
> When there are no user-defined logical servers, the OPEN logical server consists of all multiplex servers.

The OPEN logical server supports these use cases:

- A large multiplex deployment is likely to have some applications that require provisioning of their own resources. It is also likely to have other applications that are used in an ad hoc manner, and do not require dedicated resources. These ad hoc applications can be served by the OPEN logical server, rather than by user-defined logical servers.
- In a utility computing environment where resources are used on a chargeback basis, you can keep the OPEN logical server empty as all applications and users access the multiplex with their assigned logical servers.

Parent topic: Use Built-in Logical Servers [page 81]

Related Information

ALL Logical Server [page 82]
AUTO Logical Server [page 82]
COORDINATOR Logical Server [page 83]
NONE Logical Server [page 83]
SERVER Logical Server [page 84]

6.3.6 SERVER Logical Server

On each multiplex server, the SERVER logical server has implicit membership to that server only, allowing certain privileged users to connect to any node of the multiplex and execute server-level administrative tasks without knowing which logical servers include that node.

SERVER is the default logical server for `dbbackup` and `dbstop` commands.

When you connect using the SERVER logical server context, SAP IQ ignores the NodeType connection parameter.

You must have the ACCESS SERVER LS system privilege to connect to a multiplex with the SERVER logical server context.

Parent topic: Use Built-in Logical Servers [page 81]

Related Information

ALL Logical Server [page 82]
AUTO Logical Server [page 82]
COORDINATOR Logical Server [page 83]
NONE Logical Server [page 83]
OPEN Logical Server [page 84]

6.4 Creating a Logical Server

Create a user-defined logical server using Interactive SQL.

Prerequisites

- Requires the MANAGE MULTIPLEX system privilege.

Procedure

1. Start Interactive SQL and connect to the coordinator.

   ```
   dbisql
   ```

2. Create the logical server.

   {·} Sample Code

   This example creates a user-defined logical server `ls1` with three multiplex nodes — `n1`, `n2`, and `n3` — as its members:

   ```
   CREATE LOGICAL SERVER ls1 MEMBERSHIP (n1, n2, n3);
   ```

   See *CREATE LOGICAL SERVER Statement* in *SAP IQ SQL Reference*.

Related Information

CREATE LOGICAL SERVER Statement

6.5 Commenting on Logical Servers

To simplify administration, comment on user-defined logical servers.

Prerequisites

- Requires the MANAGE MULTIPLEX system privilege.

Procedure

1. Start Interactive SQL and connect to the coordinator.

   ```
   dbisql
   ```

2. Add a comment to the logical server.

   {·} Sample Code

   This example adds a comment to a user-defined logical server `ls1`:

   ```
   COMMENT ON LOGICAL SERVER ls1 IS 'ls1: Primary Logical Server';
   ```

   See *COMMENT Statement* in *SAP IQ SQL Reference*.

Related Information

COMMENT Statement

6.6 Altering Membership of a Logical Server

Alter the membership or login policy of a user-defined logical server using Interactive SQL.

Prerequisites

- Requires the MANAGE MULTIPLEX system privilege.

Procedure

1. Start Interactive SQL and connect to the coordinator.

   ```
   dbisql
   ```

2. Alter the logical server.

   Sample Code

   This example alters a user-defined logical server `ls1` by adding multiplex nodes `n4` and `n5`:

   ```
   ALTER LOGICAL SERVER ls1 ADD MEMBERSHIP (n4, n5) ;
   ```

   See *ALTER LOGICAL SERVER Statement* in *SAP IQ SQL Reference.*

Related Information

ALTER LOGICAL SERVER Statement

6.7 Dropping a Logical Server

Drop a user-defined logical server using Interactive SQL.

Prerequisites

- Requires the MANAGE MULTIPLEX system privilege.

Context

Dropping a user-defined logical server results in dropping all node membership definitions.

Each login policy that has an explicit assignment to the logical server drops the logical server assignment from the login policy. However, if the logical server is the only one that is assigned to the login policy, then the logical server assignment for the login policy is set to NONE.

Existing connections to a logical server remain unaffected when it is dropped. Ensure that there are no active connections for a logical server when it is being dropped.

Procedure

1. Start Interactive SQL and connect to the coordinator.

   ```
   dbisql
   ```

2. Drop the logical server.

   {·} Sample Code

   This example drops a user-defined logical server `ls1`:

   ```
   DROP LOGICAL SERVER ls1
   ```

   See *DROP LOGICAL SERVER Statement* in *SAP IQ SQL Reference.*

Related Information

DROP LOGICAL SERVER Statement

6.8 Connecting to a Logical Server

Use the `LogicalServer` and `NodeType` connection parameters to establish the logical server context for a new user connection.

You can enter a connect statement from the Interactive SQL command line or include it in a configuration file.

Connections fail if:

- The current node is not a member of any logical server assigned to the user's login policy.
- The current login policy assigns SERVER logical server, and the user lacks sufficient system privilege.
- The current login policy assigns NONE as logical server.
- The current login policy assigns COORDINATOR as the logical server, and the user connects to a secondary server without enabling login redirection.

In a connect statement, specify the target logical server name and role by adding the `LogicalServer=<target-logical-server>` and `NodeType={ READER | WRITER | ANY }` clauses. For example, to connect an application that specifically needs to execute user-defined functions on member nodes with the READER role, use:

Sample Code

```
dbisql -c
"uid=DBA;pwd=<password>;eng=host4_iqdemo;LogicalServer=LogSvr1;NodeType=READER
```

When you connect using the SERVER logical server context, SAP IQ ignores the `NodeType` connection parameter.

For new and upgraded databases, the `default_logical_server` login policy is AUTO. User-defined login policies use the value from the root login policy, which defaults to AUTO. When the default logical server policy is AUTO, login redirection never occurs, even if the LOGIN REDIRECTION logical server policy option. is set ON.

Under the AUTO setting, if the connection string fails to specify `LogicalServer`, SAP IQ automatically determines logical server context as follows:

Current Physical Node Status	Result
Node belongs to multiple logical servers assigned to a single login policy. A user belonging to that login policy logs in to the common node, but SAP IQ cannot determine which logical server to use.	Connection refused and error raised due to overlapping server scenario.
Node belongs to a single logical server.	Connection succeeds and context is that logical server.

6.9 Configure Logical Server Policies

A logical server policy is associated with each logical server. Configure logical server policy options to control behavior of all associated logical servers.

An SAP IQ database includes a built-in root logical server policy that applies to all logical servers. You cannot drop the root logical server policy.

See `CREATE LS POLICY` and `ALTER LS POLICY` for valid logical server policy options.

Related Information

Altering Root Logical Server Policy [page 90]
ALTER LS POLICY Statement
CREATE LS POLICY Statement

6.10 Altering Root Logical Server Policy

Alter the root logical server policy of a user-defined logical server using Interactive SQL.

Prerequisites

- Requires the MANAGE MULTIPLEX system privilege.

Procedure

1. Start Interactive SQL and connect to the coordinator.

   ```
   dbisql
   ```

2. Alter the policy:

Option	Description
To alter the root policy:	Sample Code `ALTER LS POLICY root TEMP_DATA_IN_SHARED_TEMP=OFF;`

To alter a user-defined logical server policy:	Sample Code
	`CREATE LS POLICY lsp1 TEMP_DATA_IN_SHARED_TEMP=OFF;`
	`ALTER LS POLICY lps1 TEMP_DATA_IN_SHARED_TEMP=ON;`

   See `ALTER LS POLICY` for a complete list of logical server policy options.

   See *ALTER LS POLICY Statement* and *CREATE LS POLICY Statement*in *SAP IQ SQL Reference.*

Related Information

ALTER LS POLICY Statement
CREATE LS POLICY Statement

6.11 Manage Logical Server Membership

You can only access multiplex servers by using logical servers. Login policies control user access to the logical servers.

You can assign one or more logical servers to a login policy. All users using the login policy can only access those multiplex servers that are effective members of the assigned logical servers. You can make one of these assignments to a login policy:

ALL

Allows access to all logical servers, so that you don't need to specify server names when you add or drop them.

AUTO

Prevents login redirection. Returns an error if the node belongs to multiple logical servers. If the node belongs to a single logical server, uses the logical server context of the current node.

COORDINATOR

Allows access to the current coordinator node, so that, should the coordinator role pass from one server to another, you don't need to specify the new server name.

Name of logical server

One or more existing user-defined logical servers.

OPEN

Allows access to all multiplex nodes that are not members of any user-defined logical servers.

SERVER

Allows access to all multiplex nodes, subject to the semantics of the SERVER logical server.

NONE

Denies access to any multiplex server.

> **Note**
>
> Do not combine SERVER or NONE with other logical server assignments.

In this section:

When multiple logical servers are assigned to a single login policy, no overlap in node membership is allowed among those logical servers, including the logical membership of the coordinator.

6.11.1 Logical Server Assignment Inheritance

A login policy without a logical server assignment inherits assignments from the root login policy.

By default, the logical server assignment of the root login policy includes only the OPEN logical server. This ensures that all multiplex servers are accessible when default configuration is in use.

The effective logical server assignment of a login policy is either from the logical server assignment that is made explicitly to the login policy or from the inheritance of logical server assignments of the root login policy.

> **Note**
>
> Logical server assignment of NONE is different from the case when there are no logical server assignments.

6.11.2 Lock-Down Prevention

To prevent a lock-down situation due to login policy logical server access configuration, DBA connections will not fail even if the login policy prevents access to the node.

When the connection cannot be accepted based upon login policy's logical server access configuration, then the DBA connection is accepted with SERVER logical server context.

6.11.3 Changing Logical Server Assignments

Alter logical server assignments using Interactive SQL.

Prerequisites

- Requires the MANAGE ANY LOGIN POLICY system privilege.

Procedure

1. Start Interactive SQL and connect to the coordinator.

   ```
   dbisql
   ```

2. Alter the login policy. The `LOGICAL SERVER` clause lets you configure values of certain login policy options on a per logical server basis.

   Sample Code

   This example alters a logical server assignment by assigning logical server `ls1` to the login policy `lp1`:

   ```
   ALTER LOGIN POLICY lp1 ADD LOGICAL SERVER ls1
   ```

   See *ALTER LOGIN POLICY Statement* in *SAP IQ SQL Reference.*

Related Information

ALTER LOGIN POLICY Statement

6.11.4 Removing All Logical Server Assignments

Remove all existing logical server assignments from a user-defined login policy.

Prerequisites

- Requires the MANAGE ANY LOGIN POLICY system privilege.

Context

To remove existing logical server assignments from a user-defined login policy, set the logical server assignment to DEFAULT. Setting the logical server assignment to DEFAULT in the root login policy restores the default assignment of OPEN logical server.

Procedure

1. Start Interactive SQL and connect to the coordinator.

   ```
   dbisql
   ```

2. Alter the login policy to use DEFAULT.

For example, to alter a logical server assignment by assigning logical server `DEFAULT` to the user-defined login policy `lp1`, enter:

```
ALTER LOGIN POLICY lp1 ADD LOGICAL SERVER DEFAULT
```

6.11.5 Node Membership Overlap

When multiple logical servers are assigned to a single login policy, no overlap in node membership is allowed among those logical servers, including the logical membership of the coordinator.

An automatic membership overlap check is enforced during changes to:

Logical server membership

If one or more member nodes are added to a logical server, each login policy that allows access to this logical server is examined to ensure that the new membership does not overlap with memberships of other logical servers assigned to the login policy. Attempt to change a logical server membership fails if such an overlap occurs.

Logical server assignment of a login policy

If one or more user-defined logical servers are added to a login policy, a check ensures that the new logical server assignment of the login policy does not cause a membership overlap among assigned logical servers. Attempt to change logical server assignment for a login policy fails if such an overlap occurs.

> ⓘ Note
>
> Logical servers can be defined with overlapping memberships as long as they are not assigned to the same login policy.

6.12 Redirecting Logins

SAP IQ provides load balancing when a user tries to log in to an overloaded node by redirecting the attempted login to a node that is less loaded in the same logical server.

Prerequisites

- Requires the MANAGE MULTIPLEX system privilege.

Context

Control login redirection using the logical server policy options LOGIN_REDIRECTION and REDIRECTION_WAITERS_THRESHOLD. See related reference *CREATE LS POLICY Statement*.

Procedure

1. Create a logical server policy where login redirection is enabled.

   Sample Code

   ```
   CREATE LS POLICY lsp1 LOGIN_REDIRECTION=ON
   ```

2. Associate the new policy with a logical server:

   Sample Code

   ```
   ALTER LOGICAL SERVER ls1 POLICY lsp1
   ```

3. Set the REDIRECTION_PREFERENCE option for your database. See related reference *REDIRECTION_PREFERENCE Database Option*.

Results

If login redirection is enabled and a connection is allowed, SAP IQ redirects connections when:

- The initial connection node is not a member of the target logical server.
- The initial connection node is a member of the target logical server, but has a role other than that requested.
- The initial node is a member of the target logical server and has the requested role, but the user has reached the limit of maximum connections on the current logical server member node. Maximum connection here refers to the option `MAX_CONNECTIONS` specified during CREATE LOGIN POLICY which is associated with the current logical server and assigned to the user.

A direct connection to a server that has reached its connection limit (`-gm`) is refused, not redirected, even if that node participates in a logical server.

Related Information

Connecting to a Logical Server [page 88]
start_iq Database Server Startup Utility
Disabling Login Redirection [page 96]
ALTER LS POLICY Statement
CREATE LS POLICY Statement
REDIRECTION_PREFERENCE Option

6.13 Disabling Login Redirection

You can disable login redirection for all logical servers governed by a named logical server policy, or at the connection level.

Prerequisites

- Requires the MANAGE MULTIPLEX system privilege.

Procedure

1. Disable login redirection at the logical server level. You cannot enable login redirection at the connection level after you disable it at the logical server level:

   Sample Code

   ```
   ALTER LS POLICY mypolicy LOGIN_REDIRECTION=OFF
   ```

2. Disable login redirection at the connection level if you do not need to affect all servers associated with a particular logical server policy. When disabled, no redirection occurs for the connection, and the connection fails if the node specified cannot satisfy connection requirements of the target logical server and requested role.

   Sample Code

   For example, if an application needs to target specific nodes within a logical server that contains data in tables not shared between nodes, such as SQL Anywhere tables, disable at the connection level:

   ```
   dbisql -c
   "uid=dba;pwd=<password>;eng=mpxnode_w1;links=tcpip{host=<host_name>;port=<p
   ort>};redirect=off"
   ```

6.14 Login Policies

A login policy defines the rules that SAP IQ follows to establish user connections. Each login policy is associated with a set of options called login policy options.

Login management commands that you execute on any multiplex server are automatically propagated to all servers in the multiplex. For best performance, execute these commands, or any DDL, on the coordinator.

Each new database is created with a default login policy, called ROOT. When you create a user account, if you don't specify a login policy, then the user is assigned the ROOT login policy. The ROOT login policy can be modified, but not deleted.

Each user can be associated with only one login policy. You can create as many user-defined login policies with as many combinations of login policy and LDAP login policy options as needed. A new user-defined login policy starts with the default values as defined in the ROOT policy. Add only those policy options that require different values from the default value to the user-defined policy. To restore an option value in a user-defined login policy back to the default value, set the value to DEFAULT in the policy.

Use a login policy to permanently lock a user account. This prevents a user from connecting without having to delete the user account. Create a login policy with the LOCK option set to ON. Then, assign the login policy to any user to be personalty locked. To unlock a user account, assign a login policy with the LOCK option set to OFF or DEFAULT to the user.

To control how many failed login attempts are allowed and temporarily lock the account once the maximum attempts is exceeded, create a login policy with the MAX_FAILED_LOGIN_ATTEMPTS option. Once an account is temporarily locked, to define a wait period before the user than try logging in again, add the AUTO_UNLOCK_TIME option to the same login policy. If the AUTO_UNLOCK_TIME option is not added to the policy, the user account remains temporarily locked until it is manually unlocked using the ALTER USER statement. The default value in ROOT for the MAX_FAILED_LOGIN_ATTEMPTS is unlimited. SAP strongly recommends that a login policy with a fixed failed attempts limit be implanted.

In this section:

Related Information

Login Policy Options
ALTER LOGIN POLICY Statement
CREATE LOGIN POLICY Statement

6.14.1 Modifying the Root Login Policy

You can modify the option values for the root login policy, but you cannot drop the policy.

Prerequisites

The MANAGE ANY LOGIN POLICY system privilege.

Context

Each new database is created with a default login policy, called the root policy. When you create a user account without specifying a login policy, the user becomes part of the root login policy.

Procedure

Modify the options of the root login policy by executing:

```
ALTER LOGIN POLICY ROOT {<login_policy_options>}
```

Related Information

ALTER LOGIN POLICY Statement
Managing LDAP User Authentication Login Policy Options
Login Policy Options
LDAP Login Policy Options
Multiplex Login Policy Configuration

6.14.2 Creating a New Login Policy

Any options that are not explicitly set when creating a login policy inherit their values from the root login policy.

Prerequisites

- Requires the MANAGE ANY LOGIN POLICY system privilege.

Context

Login policy names are unique. An error appears if the login policy name already exists.

Procedure

Execute:

```
CREATE LOGIN POLICY <policy_name> {<login_policy_options>}
```

Example

This statement creates the `Test1` login policy with PASSWORD_LIVE_TIME option set to 60 days:

```
CREATE LOGIN POLICY Test1 password_life_time=60
```

Related Information

CREATE LOGIN POLICY Statement
Login Policy Options
LDAP Login Policy Options
Multiplex Login Policy Configuration

6.14.3 Modifying an Existing Login Policy

Modify options within an existing login policy.

Prerequisites

- Requires the MANAGE ANY LOGIN POLICY system privilege.

Procedure

1. (Optional) Review the policy options currently assigned to the login policy:

```
SELECT SYSLOGINPOLICY.login_policy_name,
SYSLOGINPOLICYOPTION.login_option_name,
SYSLOGINPOLICYOPTION.login_option_value
   FROM SYSLOGINPOLICYOPTION
   JOIN SYSLOGINPOLICY ON
SYSLOGINPOLICY.login_policy_id=SYSLOGINPOLICYOPTION.login_policy_id
   WHERE login_policy_name='<policy-name>';
```

2. Alter the options of an existing login policy by executing:

```
ALTER LOGIN POLICY <policy-name> <policy_option> [...]
```

Example

This statement alters the LOCKED and MAX_CONNECTIONS options on the `Test1` login policy:

```
ALTER LOGIN POLICY Test1 locked=on max_connections=5;
```

This example alters the value of `LOCKED` back to the default value of `OFF` in login policy `Test1`:

```
ALTER LOGIN POLICY Test1 LOCKED=DEFAULT;
```

Related Information

ALTER LOGIN POLICY Statement
Login Policy Options
LDAP Login Policy Options
Multiplex Login Policy Configuration

6.14.4 Displaying a List of Users Assigned a Login Policy

Before you can drop a login policy, ensure that it is not currently assigned to any users.

Procedure

To display the users assigned to a login policy, select from the SYSUSER system view based on the login policy identifier.

Example

This statement lists users assigned to the `lp1` login policy:

```
SELECT user_name FROM sysuser WHERE login_policy_id = ( SELECT login_policy_id
FROM
    sysloginpolicy WHERE login_policy_name='lp1' )
```

6.14.5 Deleting a Login Policy

You cannot delete the root login policy, or one that is currently assigned to a user.

Prerequisites

- Requires the `MANAGE ANY LOGIN POLICY` system privilege.

Procedure

1. Verify that no users are currently assigned the login policy to be dropped.

   ```
   SELECT SYSUSER.user_name, sysloginpolicy.login_policy_name
   FROM SYSLOGINPOLICY
   JOIN SYSUSER on SYSUSER.login_policy_id=SYSLOGINPOLICY.login_policy_id
   WHERE SYSLOGINPOLICY.login_policy_name='<policy-name>';
   ```

2. To drop the login policy, execute:

   ```
   DROP LOGIN POLICY <policy-name>;
   ```

Related Information

DROP LOGIN POLICY Statement

6.14.6 Assigning a Login Policy When Creating a New User

If you do not assign a login policy when creating a user account, the account is assigned the root login policy.

Prerequisites

The MANAGE ANY LOGIN POLICY system privilege.

Context

Assign a login policy other than the root login policy when creating a new user. A user can be assigned only one login policy at a time.

Procedure

Execute:

```
CREATE USER <userID> [ IDENTIFIED BY <password> ]
    [ LOGIN POLICY <policy-name> ]
```

You cannot specify multiple user IDs in the same `CREATE USER` command when assigning a login policy to users.

This statement creates a user called `Joe` with the password `welcome`, and assigns the login policy `Test2`:

```
CREATE USER Joe IDENTIFIED BY welcome LOGIN POLICY Test2
```

Related Information

CREATE USER Statement

6.14.7 Assigning a Login Policy to an Existing User

Assign a login policy to an existing SAP IQ user.

Prerequisites

- Requires the MANAGE ANY LOGIN POLICY system privilege.

Procedure

1. Execute:

   ```
   ALTER USER <userID> LOGIN POLICY <policy_name>
   ```

2. Have the user log out and back in to apply the new login policy.

Related Information

ALTER USER Statement

7 Multiplex Reference

Certain SQL statements, utilities, database options, and system objects have special syntax for multiplex functionality.

In this section:

7.1 Database Options

On a multiplex, database options control connections, distributed query processing and multiplex internode communications.

Related Information

MPX_AUTOEXCLUDE_TIMEOUT Option
MPX_HEARTBEAT_FREQUENCY Option
MPX_IDLE_CONNECTION_TIMEOUT Option
MPX_LIVENESS_TIMEOUT Option
MPX_MAX_CONNECTION_POOL_SIZE Option
MPX_MAX_UNUSED_POOL_SIZE Option
MPX_MIPC_TIMEOUT Option

7.2 System Tables

Certain system tables support multiplex functionality.

See *SAP IQ SQL Reference >System Tables* for syntax and usage information for the following:

- ISYSIQINFO system table
- ISYSIQLOGICALSERVER system table
- ISYSIQLOGINPOLICYLSINFO system table
- ISYSIQLSLOGINPOLICYOPTION system table
- ISYSIQLSMEMBER system table
- ISYSIQLSPOLICY system table
- ISYSIQLSPOLICYOPTION system table
- ISYSIQMPXSERVER system table
- ISYSIQMPXSERVERAGENT system table

Related Information

ISYSIQINFO System Table
ISYSIQLOGICALSERVER System Table
ISYSIQLOGINPOLICYLSINFO System Table
ISYSIQLSLOGINPOLICYOPTION System Table
ISYSIQLSMEMBER System Table
ISYSIQLSPOLICY System Table
ISYSIQLSPOLICYOPTION System Table
ISYSIQMPXSERVER System Table
ISYSIQMPXSERVERAGENT System Table

7.3 System Views

Certain system views support multiplex functionality.

See *SAP IQ SQL Reference > System Tables and Views > System Views > Alphabetical List of System Views* for syntax and usage information for the following:

- SYSIQLOGICALSERVER system view
- SYSIQLOGINPOLICYLSINFO system view
- SYSIQLSLOGINPOLICIES consolidated view
- SYSIQLSLOGINPOLICYOPTION system view
- SYSIQLSMEMBER system view
- SYSIQLSMEMBERS consolidated view
- SYSIQLSPOLICY system view

- SYSIQLSPOLICYOPTION system view
- SYSIQMPXSERVER system view
- SYSIQMPXSERVERAGENT system view

Related Information

SYSIQLOGICALSERVER System View
SYSIQLOGINPOLICYLSINFO System View
SYSIQLSLOGINPOLICIES Consolidated View
SYSIQLSLOGINPOLICYOPTION System View
SYSIQLSPOLICY System View
SYSIQLSPOLICYOPTION System View
SYSIQMPXSERVER System View
SYSIQMPXSERVERAGENT System View

7.4 System Procedures

Certain system procedures have special syntax or usage notes to support multiplex functionality.

Where syntax is not provided, syntax is common across SAP IQ and multiplex servers and is documented in *SAP IQ SQL Reference*.

See *SAP IQ SQL Reference* for syntax and usage information for the following:

- sp_iqcheckdb procedure
- sp_iqconnection procedure
- sp_iqdbsize procedure
- sp_iqdbspace procedure
- sp_iqdbspaceinfo procedure
- sp_iqemptyfile procedure
- sp_iqfile procedure
- sp_iqindexinfo procedure
- sp_iqmpxdumptlvlog procedure
- sp_iqmpxfilestatus procedure
- sp_iqmpxincconnpoolinfo procedure
- sp_iqmpxincheartbeatinfo procedure
- sp_iqmpxincstatistics procedure
- sp_iqmpxinfo procedure
- sp_iqmpxsuspendedconninfo procedure
- sp_iqmpxvalidate procedure
- sp_iqmpxversioninfo procedure
- sp_iqsharedtempdistrib procedure

- sp_iqspaceinfo procedure
- sp_iqspaceused procedure
- sp_iqstatus procedure
- sp_iqtransaction procedure
- sp_iqversionuse procedure

Related Information

sp_iqcheckdb Procedure
sp_iqconnection Procedure
sp_iqdbsize Procedure
sp_iqdbspace Procedure
sp_iqdbspaceinfo Procedure
sp_iqemptyfile Procedure
sp_iqfile Procedure
sp_iqindexinfo Procedure
sp_iqmpxdumptlvlog Procedure
sp_iqmpxfilestatus Procedure
sp_iqmpxincconnpoolinfo Procedure
sp_iqmpxincheartbeatinfo Procedure
sp_iqmpxincstatistics Procedure
sp_iqmpxinfo Procedure
sp_iqmpxsuspendedconninfo Procedure
sp_iqmpxvalidate Procedure
sp_iqmpxversioninfo Procedure
sp_iqsharedtempdistrib Procedure
sp_iqspaceinfo Procedure
sp_iqspaceused Procedure
sp_iqstatus Procedure
sp_iqtransaction Procedure
sp_iqversionuse Procedure

7.5 Startup and Database Administration Utilities

Certain command-line utilities have multiplex syntax or restrictions.

In this section:

7.5.1 Backup Utility (dbbackup)

The `dbbackup` utility truncates the database name to 70 characters and creates a target file with a truncated name. SAP IQ uses `dbbackup` when synchronizing secondary servers. Due to the `dbbackup` restrictions, database names must be less than 70 characters.

See dbbackup Database Administration Utility .

7.5.2 Server Startup Utility (start_iq)

Run `start_iq` at the command line to customize your multiplex server startup.

Use server startup switches with the `start_iq` command to configure SAP IQ multiplex servers at startup.

> **Note**
>
> See start_iq Database Server Startup Utility in the *SAP IQ Utility Reference* for detailed information on `start_iq` switches, server options, database options, and file parameters.

Startup Switch	Values	Description
`-iqgovern <num>`	Integer	Sets the number of concurrent queries allowed by the server. The number of concurrent queries is not the same as the number of connections. A single connection can have multiple open cursors.
`-iqfrec`	Database name	Opens database in forced recovery mode.
`-iqmpx_failover`	1	Initiates multiplex coordinator failover to establish the designated failover Secondary node as the new coordinator.
`-iqmpx_ov`	1	Performs multiplex configuration override for the current node. Used to change node properties during startup in the event that a node's location or other property has changed. Initiates multiplex coordinator failover to establish the designated failover secondary node as the new coordinator. Starting the coordinator with this option has no effect.
`-iqmpx_reclaimwriterfreelist`	Server name	This option applies only while restarting a coordinator node. The coordinator forcefully reclaims the free list of the writer node identified by the server name. Use this switch only when a writer fails and cannot be restarted.
`-iqmpx_sn`	1	Initiates multiplex. Runs the current multiplex node in single-node mode. Use single-node mode only to fix problems with the multiplex configuration. You must shut down all other nodes in the multiplex. Use single-node mode only on the coordinator.

Startup Switch	Values	Description
`-iqmsgnum <num>`	0-64 (inclusive)	Specifies the number of message log archives of the old message log maintained by the server. Default value is 0, which means that messages are wrapped in the main message log file. Takes effect only if `-iqmsgsz` or the `IQMsgMaxSize` server property is nonzero. The `IQMsgNumFiles` server property corresponds to `-iqmsgnum` and takes precedence over the value of `-iqmsgnum`. If the value is not set, the default minimum pool size is: `MIN (MAX (4, number of cores/4) , mipcmaxt (if set))`
`-iqmsgsz <size>`	Integers 0-2047 inclusive, in MB.	Limits the maximum size of the message log. The default value is 0, which specifies no limit on the size of the message file.
`-mipcmint <size>`	Integers 0-256 inclusive	Specifies the minimum number of threads allowed in the shared thread pool for MIPC request handling. Each new MIPC server connection adds two threads to the pool. The value of `-mipcmint` defaults to 0 and cannot exceed the `-mipcmaxt` value. Set this value only if advised to do so by SAP Technical Support. If the value is not set, the default minimum pool size is: `MIN (MAX (4, number of cores/4), mipcmaxt (if set))`
`-mipcmaxt <size>`	Integers 0-256 inclusive	Specifies the maximum number of threads allowed in the shared thread pool for MIPC request handling. Each new MIPC server connection adds two threads to the pool. The value of `-mipcmaxt` defaults to 0 and must exceed the `-mipcmint` value. Set this value only if advised to do so by Technical Support. If the value is not set, the default maximum pool size is: `MAX (number of cores, mipcmint)`

> **Note**
>
> The `-iqmc` and `-iqtc` switches allow different cache sizes for each node in a multiplex, but this may have adverse affects. For example, if a node worker is configured with a much smaller cache than the leader, hash joins on the leader will operate in a paging mode that disallows parallelism.

Important Disclaimers and Legal Information

Hyperlinks

Some links are classified by an icon and/or a mouseover text. These links provide additional information.
About the icons:

- Links with the icon  : You are entering a Web site that is not hosted by SAP. By using such links, you agree (unless expressly stated otherwise in your agreements with SAP) to this:
  - The content of the linked-to site is not SAP documentation. You may not infer any product claims against SAP based on this information.
  - SAP does not agree or disagree with the content on the linked-to site, nor does SAP warrant the availability and correctness. SAP shall not be liable for any damages caused by the use of such content unless damages have been caused by SAP's gross negligence or willful misconduct.
- Links with the icon  : You are leaving the documentation for that particular SAP product or service and are entering an SAP-hosted Web site. By using such links, you agree that (unless expressly stated otherwise in your agreements with SAP) you may not infer any product claims against SAP based on this information.

Videos Hosted on External Platforms

Some videos may point to third-party video hosting platforms. SAP cannot guarantee the future availability of videos stored on these platforms. Furthermore, any advertisements or other content hosted on these platforms (for example, suggested videos or by navigating to other videos hosted on the same site), are not within the control or responsibility of SAP.

Beta and Other Experimental Features

Experimental features are not part of the officially delivered scope that SAP guarantees for future releases. This means that experimental features may be changed by SAP at any time for any reason without notice. Experimental features are not for productive use. You may not demonstrate, test, examine, evaluate or otherwise use the experimental features in a live operating environment or with data that has not been sufficiently backed up.
The purpose of experimental features is to get feedback early on, allowing customers and partners to influence the future product accordingly. By providing your feedback (e.g. in the SAP Community), you accept that intellectual property rights of the contributions or derivative works shall remain the exclusive property of SAP.

Example Code

Any software coding and/or code snippets are examples. They are not for productive use. The example code is only intended to better explain and visualize the syntax and phrasing rules. SAP does not warrant the correctness and completeness of the example code. SAP shall not be liable for errors or damages caused by the use of example code unless damages have been caused by SAP's gross negligence or willful misconduct.

Bias-Free Language

SAP supports a culture of diversity and inclusion. Whenever possible, we use unbiased language in our documentation to refer to people of all cultures, ethnicities, genders, and abilities.

www.sap.com/contactsap