Sybase IQ Multiplex の使用

Sybase IQ 15.4

ドキュメント ID： DC01141-01-1540-01
改訂： 2011 年 11 月
Copyright © 2011 by Sybase, Inc. All rights reserved.
このマニュアルは Sybase ソフトウェアの付属マニュアルであり、新しいマニュアルまたはテクニカル・ノートで特に示されないかぎりは、後続のリリースにも付属します。このマニュアルの内容は予告なしに変更されることがあります。このマニュアルに記載されているソフトウェアはライセンス契約に基づいて提供されるものであり、無断で使用することはできません。
アップグレードは、ソフトウェア・リリースの所定の日時に定期的に提供されます。 このマニュアルの内容を弊社の書面による事前許可を得ずに、電子的、機械的、手作業、光学的、またはその他のいかなる手段によっても、複製、転載、翻訳することを禁じます。
Sybase の商標は、Sybase の商標リスト (http://www.sybase.com/detail?id=1011207) で確認できます。Sybase およびこのリストに掲載されている商標は、米国法人 Sybase, Inc. の商標です。® は、米国における登録商標であることを示します。
このマニュアルに記載されている SAP、その他の SAP 製品、サービス、および関連するロゴは、ドイツおよびその他の国における SAP AG の商標または登録商標です。
Java および Java 関連のすべての商標は、米国またはその他の国での Oracle およびその関連会社の商標または登録商標です。
Unicode と Unicode のロゴは、Unicode, Inc. の登録商標です。
このマニュアルに記載されている上記以外の社名および製品名は、当該各社の商標または登録商標の場合があります。
Use, duplication, or disclosure by the government is subject to the restrictions set forth in subparagraph (c)(1)(ii) of DFARS 52.227-7013 for the DOD and as set forth in FAR 52.227-19(a)-(d) for civilian agencies.
Sybase, Inc., One Sybase Drive, Dublin, CA 94568.

目次

マルチプレックスの概要 1
対象読者 1
Sybase IQ マルチプレックスの概要 1
Sybase IQ Multiplex のアーキテクチャ 2
マルチプレックス・アーキテクチャの図 2
データ記憶領域 3
マルチプレックス設定 4
共有ディスク・アレイ上のファイル 7
通信インフラストラクチャ 8
Sybase IQ マルチプレックス機能の変更点 10
マルチプレックスの作成 13
マルチプレックス記憶領域の要件 13
メイン・ストアの要件 13
ハードウェア要件 14
マルチプレックスでの Java 外部環境 14
マルチプレックス計画ワークシート 14
データベースのマルチプレックスへの変換 15
データベースのマルチプレックスへの変換 (Interactive SQL) 16
データベースのマルチプレックスへの変換 (Sybase Central) 17
データベースのマルチプレックスへの変換 (Sybase Control Center) 18
マルチプレックス・データベース・ファイル 18
マルチプレックス・サーバの管理 21
マルチプレックスの管理 (Interactive SQL) 21
コマンド・ラインからのマルチプレックス・サーバの起動 21

サーバ・ステータスのチェック (Interactive SQL) 22
マルチプレックス・サーバの追加 (Interactive SQL) 22
マルチプレックス・サーバの名前の変更 (Interactive SQL) 23
マルチプレックス・サーバを変更するための要件 23
データベース・ファイル・パスの変更 (Interactive SQL) 24
サーバの役割の変更 (Interactive SQL) 24
ホストとポートの変更 (Interactive SQL) 24
サーバの包含または除外 (Interactive SQL) 24
フェールオーバ・ノードの指定 (Interactive SQL) 24
マルチプレックス・サーバを削除するための要件 25
マルチプレックス・サーバの削除 (Interactive SQL) 25
サーバの同期 (コマンド・ライン) 26
ユーザ接続のカウント 26
マルチプレックスの管理 (Sybase Central) 27
マルチプレックス・サーバの起動 (Sybase Central) 27
サーバ・ステータスのチェック (Sybase Central) 28
Sybase IQ Agent 29
Windows での Sybase Central の起動 35
UNIX システムでの Sybase Central の起動 35
サーバの管理 (Sybase Central) 35
マルチプレックス・サーバの同期 37
サーバを起動するための params.cfg ファイルの編集 (Sybase Central) 38

マルチプレックス・サーバの追加 (Sybase Central) 39
マルチプレックス・サーバの名前の変更 (Sybase Central) 40
データベース・ファイル・パスの変更 (Sybase Central) 40
サーバの役割の変更 (Sybase Central) 40
ホストとポートの変更 (Sybase Central) 40
サーバの包含または除外 (Sybase Central) 40
フェールオーバ・ノードの指定 (Sybase Central) 41
マルチプレックス・サーバの削除 (Sybase Central) 41
マルチプレックスの管理 (Sybase Control Center) 41
管理用権限の使用 42
MULTIPLEX ADMIN 権限 42
マルチプレックス・プロシージャを使用したセキュリティの調整 42
関連ストアド・プロシージャを実行するためのパーミッションをユーザに付与する 43
マルチプレックスのログイン管理 44
コーディネータの障害 44
指定済みのフェールオーバ・ノード 44
フェールオーバの要件 45
コーディネータの置き換え (Sybase Central) 46
コーディネータの置き換え (コマンド・ライン) 46
コーディネータの置き換え (Sybase Control Center) 48
マルチプレックス・トランザクション 49
マルチプレックス・トランザクションの概要 49
ローカル・トランザクション 49
グローバル・トランザクション 49

グローバル・トランザクション処理時のコーディネータの障害 50
DML コマンド 50
グローバル DML コマンド 50
テーブル・データ・スコープ 51
ジョイン・インデックス内のテーブル 51
DDL コマンド 52
DDL コマンドのスコープ 52
役割制限 54
マルチプレックス内の DB 領域の更新 55
動的衝突 63
論理サーバ 65
論理サーバの概要 65
組み込みの論理サーバ 65
有効な論理サーバ設定 67
論理サーバ・メンバとしてのコーディネータ 67
論理サーバ・ポリシー 68
論理サーバの管理 (Interactive SQL) 68
論理サーバの作成 (Interactive SQL) 68
論理サーバの変更 (Interactive SQL) 69
ルート論理サーバ・ポリシーの変更 (Interactive SQL) 69
論理サーバへのコメントの追加 (Interactive SQL) 69
論理サーバの削除 (Interactive SQL) 70
論理サーバの管理 (Sybase Central) 71
論理サーバの作成 (Sybase Central) 71
論理サーバの変更 (Sybase Central) 71
ルート論理サーバ・ポリシーの変更 (Sybase Central) 72
論理サーバへのコメントの追加 (Sybase Central) 72

論理サーバの削除 (Sybase Central) 72
論理サーバの管理 (Sybase Control Center) 73
論理サーバへのアクセス 73
ノード・メンバシップの重複 73
論理サーバ割り当ての継承 74
接続の論理サーバ・コンテキスト 74
論理サーバを作成したときの論理サーバ割り当ての設定 (Sybase Central) 75
論理サーバを変更したときの論理サーバ割り当ての設定 (Sybase Central) 75
論理サーバを変更したときの論理サーバ割り当ての設定 (Interactive SQL) 76
ログイン・ポリシー・オプション 76
locked オプション 76
max_connections オプション 77
dqp_enabled ログイン・ポリシー・オプション 77
分散クエリ処理オプション 77
dqp_enabled テンポラリ・データベース・オプション 77
論理サーバ設定 78
論理サーバと新しいシンプレックス・データベース 78
マルチプレックス・データベースのシンプレックス・データベースへの変換 78
シンプレックスでの接続 79
分散クエリ処理 81
リーダ・ノードとワーカ・ノード 82
クエリ部分 82
分散クエリのパフォーマンス 82
Sybase Central のパフォーマンスと統計のモニタ 85
トポロジ・ビュー 85

マルチプレックス・トポロジの表示 85
マルチプレックス・トポロジの印刷 86
グラフの保存 86
マルチプレックス・ステータス・モニタの更新頻度
の変更 86
ノード情報の表示 87
ノード間通信の状態 87
サーバ・モード 87
サーバの状態 88
サーバ・ステータス 88
パフォーマンス・モニタへのアクセス 88
データ収集レートの設定 88
[収集レートの設定] ダイアログ 89
モニタリング対象のサーバの変更 89
チャートの保存 90
チャートの印刷 90
チャート・ビューの切り替え 90
チャートのカスタマイズ 90
統計のカテゴリー 91
マルチプレックスでのバックアップ操作とリストア操作 99
データベースの一貫性のチェック 99
バックアップ要件 99
IQ ストアとカタログ・ストアのバックアップ 100
リストア前の作業 100
異なる場所にデータベースをリストアする場合の IQ
ストア・データのリストア 101
同じ場所にデータベースをリストアする場合の IQ ス
トア・データのリストア 103
マルチプレックス環境での選択的なリストア操作 105
コーディネータ用の読み取り専用バックアッ
プのリストア 105

マルチプレックス・リファレンス 107
SQL 文 107
ALTER LOGICAL SERVER 文 107
ALTER LOGIN POLICY 文 108
ALTER LS POLICY 文 113
ALTER MULTIPLEX RENAME 文 114
ALTER MULTIPLEX SERVER 文 114
COMMENT ON LOGICAL SERVER 文 116
CREATE LOGICAL SERVER 文 117
CREATE MULTIPLEX SERVER 文 118
DROP LOGICAL SERVER 文 120
DROP MULTIPLEX SERVER 文 121
データベース・オプション 122
DQP_ENABLED オプション 122
MPX_AUTOEXCLUDE_TIMEOUT オプション . 123
MPX_HEARTBEAT_FREQUENCY オプション 123
MPX_IDLE_CONNECTION_TIMEOUT オプション 124
MPX_MAX_CONNECTION_POOL_SIZE オプション 124
MPX_MAX_UNUSED_POOL_SIZE オプション 125
MPX_WORK_UNIT_TIMEOUT オプション 126
システム・テーブル 126
ISYSIQINFO システム・テーブル 126
ISYSIQLOGICALSERVER システム・テーブル 128
ISYSIQLOGINPOLICYLSINFO システム・テーブル 128
ISYSIQLSLOGINPOLICYOPTION システム・テーブル 128

ISYSIQLSMEMBER システム・テーブル 128
ISYSIQLSPOLICY システム・テーブル 128
ISYSIQLSPOLICYOPTION システム・テーブル 129
ISYSIQMPXSERVER システム・テーブル 129
システム・ビュー 129
SYSIQLOGICALSERVER システム・ビュー 129
SYSIQLOGINPOLICYLSINFO システム・ビュー 130
SYSIQLSMEMBER システム・ビュー 130
SYSIQLSMEMBERS 統合システム・ビュー 131
SYSIQLSLOGINPOLICIES システム・ビュー ..131
SYSIQLSLOGINPOLICYOPTION システム・ビュー 132
SYSIQLSPOLICYOPTION システム・ビュー ...132
SYSIQMPXSERVER システム・ビュー 133
SYSIQLSPOLICY システム・ビュー 134
システム・プロシージャ 134
sp_iqcheckdb プロシージャ 134
sp_iqconnection プロシージャ 135
sp_iqdbsize プロシージャ 138
sp_iqdbspace プロシージャ 139
sp_iqfile プロシージャ 139
sp_iqindexinfo プロシージャ 139
sp_iqmpxincconnpoolinfo プロシージャ 139
sp_iqmpxcheckdqpconfig プロシージャ 140
sp_iqmpxdumptlvlog プロシージャ 142
sp_iqmpxfilestatus プロシージャ 143
sp_iqmpxincheartbeatinfo プロシージャ 144
sp_iqmpxinfo プロシージャ 145
sp_iqmpxvalidate プロシージャ 147
sp_iqmpxversioninfo プロシージャ 148

sp_iqsharedtempdistrib プロシージャ 148
sp_iqspaceinfo プロシージャ 150
sp_iqspaceused プロシージャ 150
sp_iqstatus プロシージャ 150
sp_iqtransaction プロシージャ 151
sp_iqversionuse プロシージャ 154
起動ユーティリティとデータベース管理ユーティリティ 156
バックアップ・ユーティリティ (dbbackup).....156
サーバ起動ユーティリティ (start_iq) 156
索引 159

マルチプレックスの概要

Sybase® IQ マルチプレックスのサーバとトランザクションを管理するには、マルチプレックス・アーキテクチャのデータ記憶領域と通信についての基本的な知識が必要です。

対象読者

このマニュアルは、複数のノードにわたる大きなクエリ負荷を管理する Sybase® IQ マルチプレックス機能のユーザを対象にしています。

マルチプレックス操作をサポートする構文、システム・オブジェクト、ユーティリティについては、このマニュアルの付録と他の Sybase IQ リファレンス・マニュアルを併せて参照してください。

Sybase IQ マルチプレックスの概要

Sybase IQ のマルチプレックスは、スケーラビリティの高い共有ディスク・グリッド・テクノロジです。マルチプレックスを使用すると、共有データ・ソースに接続された独立のデータ処理ノードを介して、データのロードとクエリを同時に実行できます。

IQ マルチプレックス・テクノロジは、次のメリットをもたらします。

- パフォーマンス
  IQ マルチプレックスでは、クエリとロードのパフォーマンスが実証されている標準 IQ サーバ・エンジンを使用します。
- 安価な総所有コスト
  IQ マルチプレックスでは、標準的で低コストのハードウェアとオペレーティング・システムを使用します。
- 簡単で柔軟性の高いセットアップと設定
  IQ マルチプレックスは、SQL インタフェースまたは使いやすい Sybase Central GUI を使用して簡単に設定できます。
- スケーラビリティ
  IQ マルチプレックスを多数のノードに適用すると、何万人ものユーザをサポートできます。ノードはマルチプレックスに追加されるため、データ・ロードとクエリのパフォーマンスの低下はほとんどありません。
- 高可用性

ノードの障害が発生しても、他のノード上でのクエリ・ジョブには影響しません。

IQ マルチプレックスは、企業全体への基幹アプリケーション配備のために理想的なプラットフォームを提供します。

Sybase IQ Multiplex のアーキテクチャ

Sybase IQ マルチプレックスでは、共有記憶領域とローカル記憶領域の両方を使用する、ハイブリッド・クラスタ・アーキテクチャが採用されています。

共有記憶領域は永続的な IQ データと、分散クエリ処理用の共有テンポラリ・データのために使用されます。ローカル記憶領域は、カタログ・メタデータ、テンポラリ・データ、トランザクション・ログのために使用されます。

各サーバには、それぞれ独自のテンポラリ・ストアとカタログ・ストアがあります。

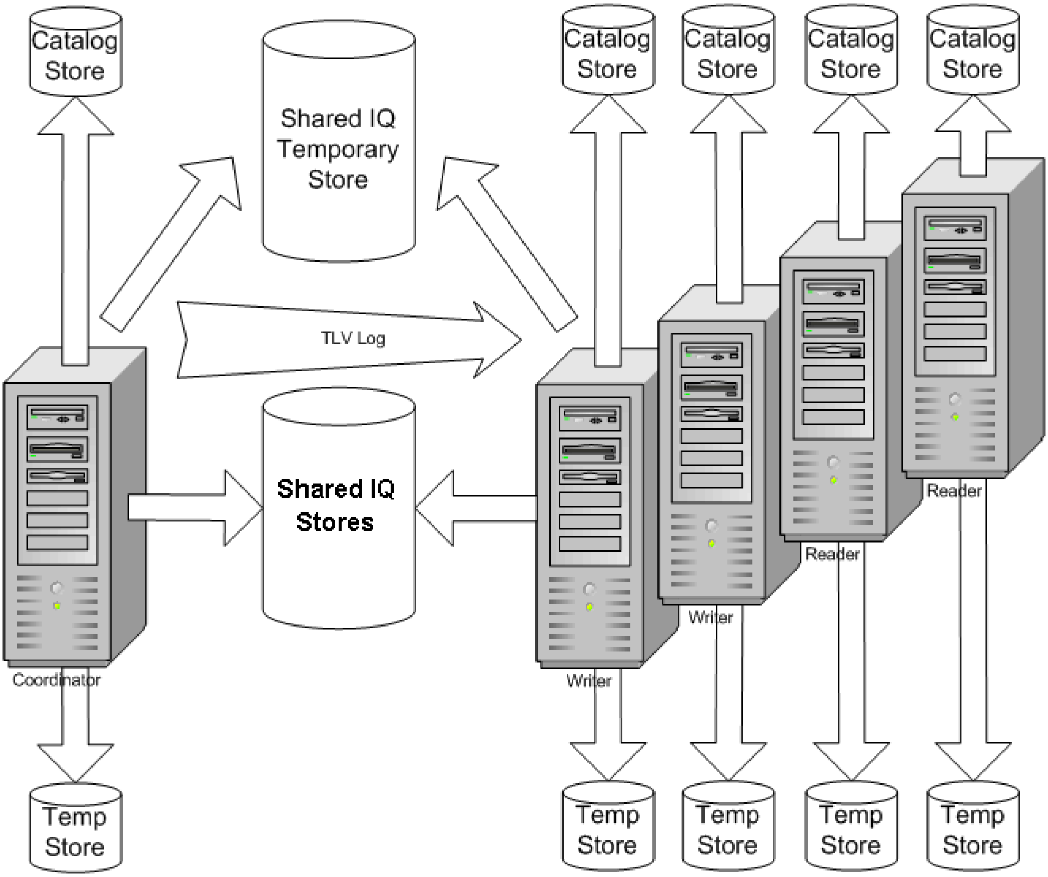
共有 IQ ストアと共有テンポラリ・ストアは、すべてのサーバに共通です。

マルチプレックス・アーキテクチャの図

「すべて共有」または「何も共有しない」のいずれかのアーキテクチャに従う通常の他のデータベース・クラスタ・アーキテクチャと異なり、Sybase IQ のマルチプレックス・ノードは、異なる役割、異なる機能を持つことができます。

設定は、「非対称クラスタ」とも呼ばれます。

図 1 : IQ マルチプレックス・アーキテクチャ



データ記憶領域

Sybase IQ のみによって管理されるデータと、基になる SQL Anywhere® ソフトウェアによって管理されるデータの違いを理解します。

『システム管理ガイド：第 1 巻』の「Sybase IQ システム管理の概要」>「Sybase IQ 内のデータ記憶領域」と『システム管理ガイド：第 1 巻』の「Sybase IQ システム管理の概要」>「SQL Anywhere と Sybase IQ」を参照してください。

表 1 : DB 領域の管理

IQ による管理	**SA による管理**
IQ_SYSTEM_MAIN IQ_SYSTEM_MSG IQ_SYSTEM_TEMP IQ_SHARED_TEMP IQ ユーザ・メイン DB 領域	システム・カタログ SA テンポラリ DB 領域 SA カタログ DB 領域

IQ_SYSTEM_MAIN、IQ_SHARED_TEMP、IQ ユーザ・メイン DB 領域はすべてのマルチプレックス・サーバによって共有されており、すべてのサーバが同じ物理ファイルにアクセスする必要があります。アドレス形式については、「共有ディスク・アレイ上のファイル」を参照してください。

SQL Anywhere によって管理されるデータは共有されません。各ノードに、そのデータの個別のコピーが必要となります。

各サーバは、IQ_SYSTEM_TEMP および IQ_SYSTEM_MSG に独自のファイル・セットを持っています。

注意： 同じ物理ホスト上にあるサーバを使用した小規模なテスト用のマルチプレックスの場合は、次のガイドラインに従うことをおすすめします。IQ によって管理される DB 領域のアドレス指定には、相対パスではなく絶対パスを使用します。SA によって管理されるカタログ DB 領域のアドレス指定には、絶対パスではなく相対パスを使用します。

参照：

- UNIX または Linux での共有アレイ・パス定義 (7 ページ)
- Windows での共有ディスク・アレイ・パス定義 (8 ページ)

マルチプレックス設定

Sybase IQ マルチプレックスを設定するには、静的コンポーネントと動的コンポーネントが必要です。

- *静的設定*には、マルチプレックスを構成するノードとそれらのプロパティが含まれます。
- *動的設定*には、ランタイム・プロセス、接続ステータス、テーブル・バージョンの使用状況が含まれます。

  Sybase IQ は、IQ テーブルの DB 領域にあるテーブル・メタデータとテーブル・バージョン (TLV) ログを格納します。このログはメイン・ストア上で共有され

るエンティティです。 使用不可になっていたノードが再び稼働状態になると、マルチプレックスでは格納済みのテーブル・バージョンが同期されます。

コーディネータ・ノード

各マルチプレックス設定には、「コーディネータ・ノード」が必要です。

既存のシンプレックス・サーバは、マルチプレックスに変換されると、コーディネータ・ノードになります。

コーディネータ・ノードの機能は次のとおりです。

- 読み取り専用操作と読み取り／書き込み操作を共有 IQ オブジェクトに対して実行する。
- IQ メイン DB 領域を管理する。
- SQL Anywhere システムおよびユーザ・テーブルでローカル・データを操作する。
- 共有 IQ オブジェクトでのすべての読み取り／書き込み操作をコーディネートする。これらの操作には、次の操作が含まれます。
  - 共有 IQ テーブルのロック
  - 共有 IQ ストアの管理
  - 共有 IQ オブジェクトに関連する読み取り／書き込みトランザクション用のグローバル・トランザクション ID の提供
  - グローバル・カタログの維持管理
- セカンダリ・サーバのカタログ同期化を制御する。
- 共有 IQ ストア・オブジェクトでスキーマ変更を実行する。
- SQL Anywhere ストア・オブジェクトでスキーマ変更を実行する。
- オブジェクト・バージョンの管理とクリーンアップを行う。

セカンダリ・ノード

Sybase IQ マルチプレックス設定には、1 つまたは複数のセカンダリ・ノードを含めることができます。

1 つのセカンダリ・ノードは、指定されたフェールオーバ・ノードとして機能します。このノードは第 1 選択ノードとして、現在のコーディネータの継続が不可能になった場合、コーディネータの役割を担います。

サポートされているセカンダリ・ノードの数は、次のように、購入済みのライセンスによって異なります。

- Demo/Trial Edition：セカンダリ・ノード数は無制限
- Small Business Edition：なし (マルチプレックスは使用できません)
- Single Application Server Edition：1 つのセカンダリ・ノード

- Enterprise Edition：セカンダリ・ノード数は無制限 (ノードごとにライセンスが必要)

セカンダリ・ノード：

- 読み取り専用ノード (リーダ・ノード) または読み取り／書き込みノード (ライタ・ノード) のいずれかとなります。

ライタ・ノード：

- 読み取り専用操作と読み取り／書き込み操作を共有 IQ オブジェクトに対して実行できます。
- テンポラリ・テーブルと SA ベース・テーブルのローカル・データを操作できます。

リーダ・ノード：

- 読み取り操作のみを共有 IQ オブジェクトに対して実行できます。
- テンポラリ・テーブルと SA ベース・テーブルのローカル・データを操作できます。

パラレル・ロード用の複数のライター

マルチプレックスには 1 つのコーディネータがあり、複数のライターとリーダーを使用できます。

このアーキテクチャは、

- マルチプレックス内のコーディネータおよび書き込み用サーバからの読み取り／書き込みトランザクションをサポートします。
- 同時に複数のサーバで、異なるテーブルに対するロードを実行できます。

これにより、複数のコンピュータのローカル CPU、メモリ、ディスク・リソースを活用できるため、ロードが高速化されます。

パラレル・ロードは異なるテーブルに対して実行する必要があります。同じテーブルに対してはロードを同時に実行できません。

テーブルのロードが別々の書き込み用サーバで実行される場合、複数のライターを設定すると、パラレル・テーブル・ロードのパフォーマンスがほぼリニアに向上します。 ロードするデータのサイズが大きい場合、ノード間の通信で発生するオーバーヘッドは無視できる程度です。 ただし、データのサイズが小さい読み取り／書き込み操作 (1 つのローに対する挿入や更新など) の場合は、単独の DML 操作と比べてノード間の通信によるオーバーヘッドが大きくなります。そのため、ライター・ノードでの負荷が高くなる、小さなデータ・サイズの読み取り／書き込み更新は、実行しないようにすることをおすすめします。

小さなバッチ処理とは、1 分未満で完了するロードのことを示します。データ・サイズの小さいトランザクションを実行するライターでは、持続的に発生する

オーバーヘッドのために、より長い時間がかかります。 実行時間の長いトランザクションの場合、一定のオーバーヘッドの占めるコストの割合は小さくなります。

参照：

- MPX_MAX_CONNECTION_POOL_SIZE オプション (124 ページ)
- MPX_MAX_UNUSED_POOL_SIZE オプション (125 ページ)

共有ディスク・アレイ上のファイル

Sybase IQ マルチプレックスが機能するには、すべてのノードから共有ストア内の DB 領域内のファイルにアクセスするのに同じファイル・パスを使用する必要があります。

共有ストア内のすべての DB 領域内のすべてのファイル (つまり、IQ_SYSTEM_MAIN と共有 DB 領域内のすべてのファイル) は、すべてのノードからまったく同じ方法でアクセスできる必要があります。共有テンポラリ・ストア内のファイルは、マルチプレックス内のすべてのノードからアクセスできる必要があります。

Sybase IQ 12.7 では、マルチプレックス内のすべてのノードに独自のデータベース・パスがあります。このパスは*エイリアス*と呼ばれ、共有 IQ メイン・ストア、テンポラリ・ストアにあるファイル、`.iqmsg` ファイルを開くときに使用されます。

Sybase IQ では、共有 IQ ストア用のエイリアスはサポートされなくなりました。IQ_SYSTEM_TEMP および IQ_SYSTEM_MSG では、独立したファイルはノードごとに維持管理されます。 これらのファイルへのパスは、ファイルを所有するノードでアクセスできる必要があります。

参照：

- UNIX または Linux での共有アレイ・パス定義 (7 ページ)
- Windows での共有ディスク・アレイ・パス定義 (8 ページ)

UNIX または Linux での共有アレイ・パス定義

UNIX または Linux で共有ディスクにアクセスする方法には、次の 2 つの方法があります。

共有ストアへの絶対パスを使用する方法。次に例を示します。

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

共有ストアへのソフト・リンクを使用する方法。次に例を示します。

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

これらの各ソフト・リンクは、絶対パスの例で示したロー・デバイスをポイントします。

参照：

- データ記憶領域 (3 ページ)
- 共有ディスク・アレイ上のファイル (7 ページ)
- メイン・ストアの要件 (13 ページ)

Windows での共有ディスク・アレイ・パス定義

Windows で共有ディスクにアクセスする方法には、次の 2 つの方法があります。

ディスクの物理番号を使用する方法。次に例を示します。

```
¥¥¥¥.¥¥PhysicalDrive1 ¥¥¥¥.¥¥PhysicalDrive2 ¥¥¥¥.¥¥PhysicalDrive3
```

ドライブ文字を使用した絶対パスを使用する方法。次に例を示します。

```
x:¥main y:¥userdb1 z:¥userdb2
```

ドライブ文字を使用した場合、ディスク数は最大 26 個に制限されます。このため、最初の方法を使用することをおすすめします。

参照：

- データ記憶領域 (3 ページ)
- 共有ディスク・アレイ上のファイル (7 ページ)
- メイン・ストアの要件 (13 ページ)

通信インフラストラクチャ

マルチプレックスに参加しているサーバは、2 つのフレームワークを使用してメッセージとデータを共有します。

ノード間通信 (INC)

INC は、コーディネータ・ノードとセカンダリ・ノードとの間のトランザクション通信を提供します。

Sybase Central の [トポロジ・ビュー] タブには、マルチプレックスのノード間の関係が表示されます。コーディネータからセカンダリ・ノードへ、およびセカンダリ・ノードからコーディネータへの INC リンクは存在しますが、セカンダリ・ノード間のリンクは存在しません。

セカンダリ・サーバは、DDL 文と読み取り／書き込み DML 文を実行するために INC を使用してコーディネータと通信します。コーディネータは、特定のストアド・プロシージャがコーディネータ上で実行された場合に INC を使用してセカンダリ・サーバと通信します。セカンダリ・サーバどうしが INC を使用して通信することはありません。

INC リンクは、ハートビート接続とプール済み接続から構成されています。

ハートビート接続

各セカンダリ・ノードには、「ハートビート接続」があります。これは、コーディネータ・ノードへの専用接続です。この接続は、セカンダリ・ノード起動時に確立され、そのセカンダリ・ノードがアクティブな間はアクティブに維持されます。

ハートビート接続は、コーディネータとセカンダリ・ノードの両方によってモニタされます。この接続が切断されると、ノードはオフラインを宣言します。ハートビート接続が一定の時間内に再度確立されない場合に、コーディネータが自動的にセカンダリ・サーバを除外するように設定することもできます。この設定には、オプション `MPX_AUTOEXCLUDE_TIMEOUT` を使用します。

参照：

- ノード間通信の状態 (87 ページ)

プール済み接続

各セカンダリ・ノードには、INC 接続プールがあります。接続プールは、トランザクション通信のために INC が使用するセカンダリ・ノードからコーディネータ・ノードへの接続を管理します。

INC 接続プールは、新しい接続を開く代わりに既存の接続を再度使用することで通信オーバヘッドを削減します。また、同時に使用できるセカンダリ・ノードからコーディネータへの INC 接続数を制御します。

セカンダリ・サーバからの INC 接続の特性は、次の 2 つのデータベース・オプションによって制御されます。

- セカンダリ・ノードからコーディネータに開くことができる接続の最大数を指定する `MPX_MAX_CONNECTION_POOL_SIZE`
- 接続が未使用プールに戻されたときに有効な状態で維持できる未使用の接続の最大数を指定する `MPX_MAX_UNUSED_POOL_SIZE`

参照：

- MPX_MAX_CONNECTION_POOL_SIZE オプション (124 ページ)
- MPX_MAX_UNUSED_POOL_SIZE オプション (125 ページ)
- ノード間通信の状態 (87 ページ)

マルチプレックス・プロセス間通信 (MIPC： Multiplex Interprocess Communication)

MIPC はすべてのマルチプレックス・ノードを接続し、分散クエリ処理と高度な可用性をサポートします。

マルチプレックス・サーバは起動すると、MIPC 接続を確立します。

MIPC は完全にメッシュ化された通信フレームワークで、パブリックとプライベートの両方の相互接続設定で動作します。パブリック相互接続設定は必須で、プライベート相互接続設定はオプションです。

プライベート高速相互接続設定は、分散クエリ処理向けです。現時点では、プライベート相互通信設定は、TCP/IP プロトコルをサポートする物理ネットワークに制限されています。

プライベート相互通信設定が利用できない場合、MIPC は、システム・テーブルの SYSIQMPXSERVER.conn_info カラムで指定されている従来のパブリック相互接続設定を使用します。このホスト・アドレスとポート・アドレスの組み合わせは、外部ユーザ接続、INC 接続、内部 MIPC 接続で共有されます。

冗長ネットワークは、より堅牢なクラスタを実現します。パブリックとプライベートの両方の相互接続設定が存在する場合、MIPC はプライベート接続が利用できる場合はプライベート接続を使用し、プライベート・ネットワークに障害が発生した場合はトラフィックをパブリック接続に経路変更します。マルチプレックスはノード間のメッセージをモニタし、障害を検出します。

可能性のある相互接続リンクの障害には、以下のものがあります。

- ケーブルの切断や破損などの物理的な障害
- ネットワーク・インフラストラクチャ機器などの電源障害
- ネットワーク・スタック内のソフトウェア障害

分散クエリ処理向けのネットワーク設定を計画する場合の推奨事項については、『インストールおよび設定ガイド』を参照してください。

Sybase IQ マルチプレックス機能の変更点

Sybase IQ15.4 では、外部ユーザ定義テーブル関数の機能が拡張されています。

Sybase IQ15.4 では、次の機能が提供されます。

- 外部環境の UDF サポート - 開発者は Java 環境を使用してストアド・プロシージャを作成することができます。開発者は標準 Java API を使用して、引数を処理したり、値 (結果セット) をサーバに返したりすることができるようになります。Sybase IQ ではマルチプレックス機能が拡張され、SQL の管理下から大き

なコンピュータ・クラスタを利用できるようになります。その一方で、エンド・ユーザは SQL 以外の言語で計算のロジックを記述することが可能です。

- テーブル UDF のサポート - テーブル関数は、入力データを処理してリレーショナル・セットを出力として生成するロジックをエンド ユーザが実装できるようにすることで、集合ベースのカスタマイズされた計算を容易にできます。出力されたリレーショナル・セットは、SQL クエリでテーブル式であるかのように使用される可能性があります。
- パラメータ化されたテーブル関数 - Sybase IQ は、従来のテーブル関数の方法を拡張して任意のローセットを入力として処理するようにし、ユーザ指定のパーティションおよび順序付けの要件をテーブル関数呼び出しの引数として使用ができるようにします。Sybase IQ ではマルチプレックス機能が拡張され、SQL の管理下から大きなコンピュータ・クラスタを利用できるようになります。その一方で、エンド・ユーザは SQL 以外の言語で計算のロジックを記述することが可能です。

マルチプレックスの作成

単一の Sybase IQ サーバをマルチプレックスに変換します。

1. 要件を満たすように、メイン・ストアとハードウェアを設定します。
2. データベースをマルチプレックスに変換します。

マルチプレックス記憶領域の要件

マルチプレックス・ストアを適切なデバイスに作成します。

ストア	サポートされるデバイス
IQ 共有ストア (IQ_SYSTEM_MAIN、IQ_SHARED_TEMP、ユーザ DB 領域)	ロー・デバイスが必要です。
IQ 非共有ストア (IQ_SYSTEM_TEMP)	ロー・デバイスと O/S ファイルの両方がサポートされます。
カタログとトランザクション・ログ	O/S ファイルが必要です。

メイン・ストアの要件

メイン・ストアが、マルチプレックス操作の前提条件を満たす必要があります。

メイン・ストア・ファイルのパスは、マルチプレックスのすべてのサーバがアクセスできるような絶対パスまたは相対パスである必要があります。

- 『Sybase IQ システム管理ガイド：第 1 巻』の「データベース・オブジェクトの管理」>「データベースの定義」を参照して、IQ データベースを作成してください。
- 以前のリリースの Sybase IQ からデータベースを移行する方法については、『インストールおよび設定ガイド』の「データベースのアップグレード」 を参照してください。

参照：

- UNIX または Linux での共有アレイ・パス定義 (7 ページ)
- Windows での共有ディスク・アレイ・パス定義 (8 ページ)

ハードウェア要件

Sybase IQ マルチプレックスでは、IQ_SYSTEM_MAIN、IQ_SHARED_TEMP、ユーザ DB 領域のデータを格納するために、共有記憶領域が必要です。

マルチプレックスに参加しているサーバを実行しているすべてのマシンに、最新バージョンの Sybase IQ がインストールされている必要があります。アップグレードの手順については、『インストールおよび設定ガイド』の「インストールの計画」>「以前のバージョンからのアップグレード」 を参照してください。

既存のシンプレックス・サーバは、マルチプレックスに変換されると、コーディネータになります。セカンダリ・サーバについては、コーディネータが使用しているパスとまったく同一のパスを使用して共有 IQ ストアにアクセスできるようにコンピュータを設定します。

Sybase IQ は以下をサポートしません。

- 異なるリリース・レベルの Sybase IQ サーバのマルチプレックス。
- 異機種間マルチプレックス (UNIX サーバと Windows サーバを使用する混合マルチプレックス)。コーディネータ・サーバとセカンダリ・サーバは、同じオペレーティング・システムかつ同じハードウェア・プラットフォームである必要があります。
- 仮想マシン上のマルチプレックス・サーバ・インスタンス。

マルチプレックスでの **Java** 外部環境

マルチプレックス設定で Java 外部環境 UDF を使用できるようにするには、Java クラス ファイルか JAR ファイルを、UDF を必要とするマルチプレックスの各ノードにインストールしてください。

Sybase Control Center、Sybase Central、または Interactive SQL の **INSTALL JAVA** 文を使用して、Java クラス・ファイルと JAR をインストールします。

マルチプレックス計画ワークシート

マルチプレックスを作成する前に、参加が予定されているオブジェクトのパスが要件を満たすことを確認します。

プラットフォーム上のデータベース・パスは、ロー・デバイスまたはオペレーティング・システム・ファイルのいずれの場合も、128 バイトに制限されています。Sybase IQ は、以下をサポートします。

- IQ 共有ストア (IQ_SYSTEM_MAIN、IQ_SHARED_TEMP、ユーザ DB 領域) 用のロー・デバイス
- 非共有 IQ ストア (IQ_SYSTEM_TEMP) 用のロー・デバイスとオペレーティング・システム・ファイル
- カタログ用とトランザクション・ログ用のみのオペレーティング・システム・ファイル

マルチプレックス作成のための Sybase Central ダイアログおよび SQL 文では、次の一部またはすべての値が必要となります。

表 2 : マルチプレックス・データベースの要件

ダイアログ項目	データ型／長さ	注意	値
ホスト名	CHAR 128	データベース・エンジンを実行するマシンの名前。	
サーバ名	CHAR 128	コーディネータのサーバ名 (このサーバ名はローカル・エリア・ネットワーク内でユニークである必要があります)。	
データベース・パス	CHAR 128	データベース・ファイルは、リモート・ロケーションではなく、必ずローカル・ディスク上に作成します。 データベース作成ウィザードでは、DB ファイルへのパスが要求されます。サーバの起動場所は指定できません。	
IQ ストア・パス (テンポラリ・ストアおよびメイン・ストア)	CHAR 128	IQ_SYSTEM_MAIN のすべてのファイルと共有のユーザ DB 領域は、すべてのノードから同じファイル・パスを使用してまったく同じ方法でアクセスできる必要があります。IQ メイン DB 領域へのパスはすべてのノードで共有されますが、テンポラリ DB 領域と `.iqmsg` DB 領域へのパスは、所有しているノードのみで有効です。	
データベース名	CHAR 70	最大 70 文字までのデータベース名。パスに含まれます。	

データベースのマルチプレックスへの変換

Sybase Central、Sybase® Control Center、または Interactive SQL を使用して、マルチプレックス・サーバを作成します。

シンプレックス・データベースをマルチプレックスに変更するには、シンプレックス・データベースに接続し、マルチプレックス・サーバを追加します。

シンプレックス・データベース・サーバはコーディネータとなり、ユーザが追加したサーバはセカンダリ・サーバと呼ばれます。

データベースのマルチプレックスへの変換 (Interactive SQL)

シンプレックス・データベースをマルチプレックスに変更するには、シンプレックス・データベースに接続し、CREATE MULTIPLEX SERVER を使用します。

前提条件

マルチプレックス環境では、すべての IQ メイン・ストア・ファイルへのパスは、マルチプレックスのすべてのサーバがアクセスできるような絶対パスまたは相対パスである必要があります

手順

1. 使用するシステムが、ハードウェアの前提条件を満たしていることを確認します。メイン・ストアの DB ファイルは、セカンダリ・ノードから参照できる必要があります。
2. Sybase IQ サーバを起動し、共有メイン・ストアのあるデータベースに接続します。 このサーバが、マルチプレックスのコーディネータになります。

   ```
   start_iq @params.cfg -n mpxnode_c -x
   "tcpip{host=host1;port=2763}" mpxtest.db
   ```
3. Interactive SQL を使用して接続します。

   ```
   dbisql
   ```
4. Interactive SQL で、**CREATE MULTIPLEX SERVER** コマンドを実行します。

   ```
   CREATE MULTIPLEX SERVER mpxnode_w1 DATABASE 'host2/mpx/
   mpxtest.db' HOST 'host2' PORT 2957 ROLE WRITER STATUS INCLUDED
   ```

   このコマンドは、接続されているサーバ `mpxnode_c` をコーディネータに変換し、セカンダリ・ライタ・サーバ `mpxnode_w1` の定義を作成します。
5. マルチプレックス・コーディネータとしてサーバを再度初期化するためにサーバが停止します (最初のセカンダリ・ノード作成時のこの停止では、Interactive SQL が接続切断 (-308) エラーを返す場合がありますが、このエラーに対する対処のアクションを実行する必要はありません)。
6. コーディネータを再起動します。

   ```
   start_iq @params.cfg -n mpxnode_c -x
   "tcpip{host=host1;port=2763}" mpxtest.db
   ```
7. カタログ・ストアをセカンダリ・サーバが実行されるディレクトリにバックアップします。

   セカンダリ・ノードが実行されるコンピュータから、このバックアップ・コマンドを実行します。これは、`.db` ファイルがローカル・ディスクにある必要があるためです。たとえば、セカンダリ・ノードの `.db` ファイルが存在するディレクトリから次のコマンドを実行します。

```
dbbackup -y -d -c
"uid=DBA;pwd=sql;links=tcpip{port=2763};eng=mpxnode_c" .
```

セカンダリ・ノードの .db ファイルが配置されているディレクトリに移動している場合は、**dbbackup** コマンドでパスの代わりにピリオド (.) を指定できます。

8. セカンダリ・サーバが実行されるディレクトリ内のトランザクション・ログ・ファイルを削除します。

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

9. **dblog** を次のように実行して、トランザクション・ログをリセットします。

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

10. セカンダリ・サーバを起動します。

```
start_iq -STARTDIR /host2/mpx @params.cfg -n mpxnode_w1 -x
"tcpip{port=2957}" mpxtest.db
```

11. Interactive SQL を起動し、セカンダリ・マルチプレックス・ノードに接続します。

```
dbisql -c "uid=DBA;pwd=sql;eng=mpxnode_w1;links=tcpip{port=2957}"
```

12. **ALTER DBSPACE** 文を使用して、テンポラリ DB ファイルを Interactive SQL に追加します

(セカンダリ・サーバでは **CREATE DBSPACE** は許可されません)。

```
ALTER DBSPACE IQ_SYSTEM_TEMP ADD FILE mpxnode_w1_temp
'w1_temp1.iqtmp' SIZE 500
```

サーバを起動するとメッセージ・ログが自動的に作成されますが、テンポラリ・ストア・ファイルはユーザが作成する必要があります。セカンダリ・サーバに接続して、そのサーバ用のテンポラリ・ストア・ファイルを追加します。テンポラリ・ストア・ファイルをマルチプレックスに追加すると、特に接続しているサーバにファイルが追加されます。サーバは起動できますが、テンポラリ・ストアを追加するまで IQ クエリは実行できません。

コマンドの参照の詳細については、『リファレンス：文とオプション』の「SQL 文」>「ALTER DBSPACE 文」を参照してください。

データベースのマルチプレックスへの変換 (Sybase Central)

サーバの作成ウィザードを使用して、シンプレックス・サーバでマルチプレックスを有効にします。

前提条件

次の手順を実行するには DBA 権限が必要です。

手順

1. 使用するシステムが、ハードウェアの前提条件を満たしていることを確認します。メイン・ストアの DB ファイルは、セカンダリ・ノードから参照できる必要があります。
2. 使用しているプラットフォームに合った方法で、Sybase Central を起動します。
3. UNIX の場合、IQ Agent が実行していることを確認します。
   `stop_iq -agent`
4. Start Server Wizard を使用して Sybase IQ サーバを起動します。
5. IQ サーバに接続します。
6. サーバ名を右クリックし、[Multiplex に変換] ([Alt]+[M]) を選択します。DBA 権限がない場合、[Multiplex に変換] のメニュー項目は無効になっています。
7. サーバ作成ウィザードの画面に表示される手順を順に実行します。

データベースのマルチプレックスへの変換 (Sybase Control Center)

データベースをマルチプレックスに変換する操作に Web ベースのツールを使用したい場合は、Sybase Control Center を使用します。

SCC の『Sybase Control Center for Sybase IQ オンライン・ヘルプ』または http://sybooks.sybase.com/nav/summary.do?prod=10680 を参照してください。

マルチプレックス・データベース・ファイル

マルチプレックス内の各サーバは、データ、メタデータ、設定情報を格納するために、いくつかのファイルを使用します。

注意： これらのファイルの中には、Sybase Central の自動作成機能のみで作成できるファイルがあります。

表 3 : マルチプレックス・データベース・ディレクトリの内容

フォルダ、ディレクトリまたはファイル名	目的
`dbname`.db	カタログ・ストアを格納するファイル。ロー・デバイスは使用できません。
`dbname`.iqmsg	Sybase IQ からのメッセージを格納するファイル。

フォルダ、ディレクトリまたはファイル名	目的
`dbname`.iqtmp	これは、IQ テンポラリ・ストアです。このファイルは、OS ファイルとしてテンポラリ・ファイルが追加された場合のみに存在します。IQ_SYSTEM_TEMP ストアは、ロー・デバイスまたはシステム・ファイルに配置できます。
`dbname`.log	データベース・トランザクション・ログを格納するファイル。ロー・デバイスは使用できません。
`dbname`.lmp	ライセンス管理プロパティ・ファイル。このファイルは、データベースの作成時に自動的に生成されます。データベースを削除すると、このファイル以外のすべてのデータベース・ファイルが削除されます。
`params.cfg`	このサーバの起動パラメータを格納するファイル。パフォーマンスや特定の要件に合わせてカスタマイズできます。このファイルは、Sybase Central を使用した場合のみに自動作成されます (Sybase Central によって作成された場合は、`params.cfg` にはサーバ名または tcpip パラメータは含まれていません)。
`start_server`	これらのスクリプトは Sybase Central によって生成されます。マルチプレックスの作成に Sybase Central を使用する場合は自動生成されます。また、Interactive SQL を使用してマルチプレックスを作成してから Sybase Central を使用して生成することもできます。
`stop_server`	
`sync_server`	

参照：

• 管理シェル・スクリプト (36 ページ)

マルチプレックス・サーバの管理

マルチプレックス・サーバを管理するには、Sybase Central、Sybase® Control Center、またはコマンド・ライン・ユーティリティと Interactive SQL の組み合わせを使用します。

マルチプレックスの管理 (Interactive SQL)

柔軟性のあるスクリプトを使用したい場合や、他の製品よりも Interactive SQL を使い慣れている場合は、Interactive SQL を使用します。

Interactive SQL はコマンド・ライン・インタフェースです。Interactive SQL を使用することによって、SQL 文を個々に入力したり、コマンド・スクリプトで実行したりして、Sybase IQ を含む Sybase サーバに対するクエリ、変更またはロードを行うことができます。

参照：

- CREATE MULTIPLEX SERVER 文 (118 ページ)
- マルチプレックス・サーバの追加 (Interactive SQL) (22 ページ)

コマンド・ラインからのマルチプレックス・サーバの起動

IQ マルチプレックス・サーバを対話形式で起動および停止するには、Sybase Central またはサーバ起動ユーティリティ **start_iq** を使用します。スクリプトから IQ マルチプレックス・サーバを起動するには、コマンド・ライン・パラメータを使用します。

start_iq ユーティリティは、シンプレックス・サーバまたはマルチプレックス・サーバを起動します。**-n** <エンジン名> スイッチを指定する必要があります。エンジン名は、マルチプレックス・サーバの作成時に使用したサーバ名と一致している必要があります。**-x** (接続文字列) 値は、マルチプレックス・サーバの作成時に指定した tcpip 接続パラメータに一致している必要があります。データベース・ファイル・パスは、マルチプレックス・サーバの作成時に指定したデータベース・ファイル・パスと一致している必要があります。アプリケーション用には、**-gn** 値は、予想される同時要求数に 5 (内部イベントおよび接続用) を足した数より多い値に設定する必要があります。起動パラメータの完全なリストについては、『ユーティリティ・ガイド』の「start_iq データベース・サーバ起動ユーティリティ」>「start_iq サーバ・オプション」を参照してください。

次のコマンドは、サーバを起動し、`host1_test1_writer` という名前を付けます。

```
start_iq @/host1/mpxdevices/params.cfg -n host1_test1_writer -x
"tcpip{host=host1;port=2763}" /host1/mpxdevices/test1.db
```

サーバ・ステータスのチェック (Interactive SQL)

応答していないサーバやマルチプレックスに含まれていないサーバが存在しないことを確認したり、フェールオーバ時にどのサーバが新しいコーディネータになるのかを確認したりするために、サーバのステータスをチェックします。

1. DBA 権限のあるユーザ、または **sp_iqmpxinfo** プロシージャの EXECUTE パーミッションが割り当てられているユーザとして、コーディネータに接続します。
2. **sp_iqmpxinfo** を実行します。

参照：

- sp_iqmpxinfo プロシージャ (145 ページ)

マルチプレックス・サーバの追加 (Interactive SQL)

Interactive SQL で次の手順に従って、マルチプレックス・サーバを追加します。

1. サーバのステータスをチェックします。セカンダリ・サーバを追加するには、コーディネータが実行されている必要があります。
2. Interactive SQL で、**CREATE MULTIPLEX SERVER** コマンドを実行します。

   次に例を示します。

   ```
   CREATE MULTIPLEX SERVER mpxnode_w2 DATABASE 'host1/mpx/
   mpxtest.db' HOST 'host1' PORT 2957 ROLE WRITER STATUS INCLUDED
   ```

   コマンドの参照の詳細については、「マルチプレックス・リファレンス」>「SQL 文」>「CREATE MULTIPLEX SERVER 文」を参照してください。

   このコマンドは、セカンダリ・ライタ・サーバ `mpxnode_w2` の定義を作成します。
3. 「マルチプレックスの作成」>「データベースのマルチプレックスへの変換」>「データベースのマルチプレックスへの変換 (Interactive SQL)」の手順 7 ～ 12 を実行します。

セカンダリ・サーバの追加後、初めてマルチプレックスを起動したときに、コーディネータが次のようなエラーを返す場合があります。

```
I. 12/11 12:50:08. Trying to start TCPIP link ...
 I. 12/11 12:50:08. Multiplex environment incorrect for this server
 I. 12/11 12:50:08. Please connect and run procedure sp_iqmpxvalidate
for help
```

このエラーは、セカンダリ・サーバの IQ_SYSTEM_TEMP にテンポラリ・ファイルがないために発生します。新しく追加したサーバの IQ_SYSTEM_TEMP にテンポラリ DB ファイルを追加する必要があります。「マルチプレックスの作成」>「データベースのマルチプレックスへの変換」>「データベースのマルチプレックスへの変換 (Interactive SQL)」の手順 11 〜 12 を参照してください。このファイルを追加した後に、**sp_iqmpxvalidate** を実行して、このサーバにエラーが報告されていないことを確認します。

参照：

- マルチプレックスの管理 (Interactive SQL) (21 ページ)
- CREATE MULTIPLEX SERVER 文 (118 ページ)

マルチプレックス・サーバの名前の変更 (Interactive SQL)

Interactive SQL を使用して、マルチプレックス・サーバの名前を変更できます。

前提条件

ALTER MULTIPLEX SERVER コマンドはマルチプレックス内のすべてのサーバから実行できますが、すべての DDL と同様に、コマンドはコーディネータ上で実行することをおすすめします。役割をリーダからライタに変更する場合を除いて、変更がコミットされると指定したサーバは自動的に停止します。

手順

サーバを起動し、接続して、次の形式でコマンドを実行します。

```
ALTER MULTIPLEX SERVER oldname RENAME newname
```

このコマンドを発行すると、指定したサーバの名前が変更され、シャットダウンされます。

または、サーバを右クリックし、コンテキスト・メニューで [コントロール] - [含める／除外] を選択します。

参照：

- ALTER MULTIPLEX SERVER 文 (114 ページ)

マルチプレックス・サーバを変更するための要件

要件と結果をよく理解してから、マルチプレックス・サーバを変更してください。

コーディネータが実行されている必要があります。

コーディネータから **ALTER MULTIPLEX SERVER** コマンドを実行します。

役割をリーダからライタに変更する場合を除いて、変更がコミットされると指定したサーバは自動的に停止します。

データベース・ファイル・パスの変更 (Interactive SQL)

データベースを、たとえばより多くの領域のあるディスクに移動する必要がある場合、Interactive SQL を使用してファイル・パスを変更できます。

サーバを起動し、接続して、次の形式でコマンドを実行します。

```
ALTER MULTIPLEX SERVER server-name DATABASE 'new db file path'
```

このコマンドを発行すると、指定したサーバがシャットダウンされます。

サーバの役割の変更 (Interactive SQL)

Interactive SQL を使用して、リーダ・サーバをライタ・サーバに変更できます。

サーバを起動し、接続して、次の形式でコマンドを実行します。

```
ALTER MULTIPLEX SERVER server name ROLE {WRITER|READER}
```

コーディネータのロールは変更できません。ライタ・サーバをリード・サーバに変更すると、サーバは自動的にシャットダウンされます。

ホストとポートの変更 (Interactive SQL)

Interactive SQL を使用して、マルチプレックス・サーバのホストとポートの情報を変更できます。

サーバを起動し、接続して、次の形式でコマンドを実行します。

```
ALTER MULTIPLEX SERVER server name HOST 'hostname' PORT portnumber
```

このコマンドを発行すると、指定したサーバがシャットダウンされます。

サーバの包含または除外 (Interactive SQL)

マルチプレックス・セカンダリ・サーバを長期間にわたり停止する場合は、そのサーバをマルチプレックスから除外します。

1. サーバを起動し、接続して、次の形式でコマンドを実行します。

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

2. 除外または包含するサーバが実行されている場合、操作の前に停止することを強くおすすめします。停止しなかった場合、最終的にはサーバ自体によって停止されますが、除外または包含の前に停止を計画および実行してください。サーバを除外するとそのサーバは停止されます。サーバを包含した後は、そのサーバを同期し、再起動する必要があります。

フェールオーバ・ノードの指定 (Interactive SQL)

現在のコーディネータが処理を継続できなくなった場合にコーディネータの役割を継続させるために、フェールオーバ・ノードを指定する必要があります。

現在のコーディネータが処理を継続できなくなった場合にコーディネータの役割を継続させるために、フェールオーバ・ノードを指定する必要があります。

1. サーバを起動し、接続して、次の形式でコマンドを実行します。

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

2. 指定したフェールオーバ・ノードが、マルチプレックスに追加された最初のマルチプレックス・サーバにデフォルトとして設定されます。

参照：

- 指定済みのフェールオーバ・ノード (44 ページ)
- sp_iqmpxinfo プロシージャ (145 ページ)

マルチプレックス・サーバを削除するための要件

要件と結果をよく理解してから、マルチプレックス・サーバを削除してください。

削除するサーバが実行されている場合、削除する前に停止することを強くおすすめします。停止しなかった場合、最終的にはサーバ自体によって停止されますが、削除の前の停止を計画および実行してください。コーディネータ・ノードまたは指定したフェールオーバ・ノードは、最後のセカンダリ・ノードでない場合、削除できません。最後のセカンダリ・サーバが削除されると、マルチプレックスはシンプレックスに変換され、コーディネータが停止します。フリー・リストを保存しているライタ・サーバは削除できません。

ライト・サーバの通常の再起動を実行すると、そのフリー・リストが解放されるため、サーバを停止して削除できます。ライタが起動できない場合でプロセスが非稼動状態であることが明白な場合には、**-iqmpx_reclaimwriterfreelist** スイッチを使用してコーディネータを再起動できます。このコマンドは、コーディネータがライタのフリー・リストを要求するため、ライタを後で削除できます。この起動フラグは、ライタ・プロセスが非稼動であることと起動できないことが保証できる場合のみに使用してください。ライタ・プロセスがデータベースに書き込みを実行している場合にコーディネータがライタのフリー・リストを要求すると、データベースの破損が生じる可能性があります。

参照：

- DROP MULTIPLEX SERVER 文 (121 ページ)
- マルチプレックス・サーバの削除 (Interactive SQL) (25 ページ)

マルチプレックス・サーバの削除 (Interactive SQL)

セカンダリ・サーバを削除すると、マルチプレックス設定からも削除されます。

1. コーディネータに接続します。
2. コマンドを次の形式で発行します。

```
DROP MULTIPLEX SERVER server-name
```

参照：

- マルチプレックス・サーバを削除するための要件 (25 ページ)
- コーディネータの障害 (44 ページ)
- DROP MULTIPLEX SERVER 文 (121 ページ)

サーバの同期 (コマンド・ライン)

同期処理では、コーディネータのバージョンのデータベース・カタログがセカンダリ・サーバにコピーされます。

前提条件

セカンダリ・サーバが実行されているコンピュータで次の手順を実行します。

注意： Sybase Central でのマルチプレックス・サーバの作成時に [Admin スクリプトの生成] をオンにした場合、**dbbackup** の代わりに `sync_server` スクリプトを実行できます。

手順

1. セカンダリ・サーバ・ディレクトリの `.LOG` トランザクション・ログ・ファイルを削除します。
2. セカンダリ・サーバが実行されるディレクトリにカタログ・ストアをバックアップします。

   ```
   dbbackup -y -d -c
   "uid=dba;pwd=sql;links=tcpip{port=2763};eng=mpxnode_c" /host1/
   mpx/
   ```
3. ログ・ファイル・パスを次のように設定します。

   ```
   dblog -r -t mpxtest.log mpxtest.db
   ```
4. **start_iq** コマンドを使用してセカンダリ・ノードを起動します。
5. マルチプレックス内の各セカンダリ・サーバに対して、ここまでの手順を繰り返します。

参照：

- フェールオーバの要件 (45 ページ)
- IQ_SYSTEM_MAIN の更新 (56 ページ)

ユーザ接続のカウント

接続数が最大数を超えるのを防ぐために、定期的に、現在のユーザ接続数を確認します。

DB_PROPERTY 関数では、クライアント接続数とセカンダリ・ノードが行ったコーディネータへの INC 接続の数が返されます。

INC 接続は、起動パラメータ **-gm** によって設定される、コーディネータ・ノードに対する接続数には影響しません。

コーディネータから、システム関数 **DB_PROPERTY** を実行します。

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

マルチプレックスの管理 (Sybase Central)

マルチプレックスに詳しくないか、グラフィカル・インタフェースの使用を希望する場合は、Sybase Central を使用します。

Sybase Central は、Sybase IQ を含む Sybase の多数のサーバ製品を管理および監視するグラフィカル・ツールです。

マルチプレックス・サーバの起動 (Sybase Central)

スクリプトから IQ マルチプレックス・サーバを起動するには、コマンド・ライン・パラメータを使用します。

マルチプレックス・サーバを対話形式で起動するには、Sybase Central を使用します。

1. DBA 権限のあるアカウントを使用してログインし、使用しているプラットフォームに合った方法で Sybase Central を起動します。
2. Sybase Central の左側のウィンドウ枠で、[Sybase IQ 15] を選択します。
3. [Tools] - [Sybase IQ 15] - [Start Server] を選択します。

   デフォルトでは、ウィザードは単一のサーバを起動します。すべてのサーバを起動するには、[Multiplex のすべてのサーバ] オプション・ボタンをオンにします。
4. ウィザードの手順に従います。

   マルチプレックス内のすべてのサーバを起動する場合は、コーディネータの情報を指定します。各サーバのデフォルトのポート番号は、使用されていない別の番号に必ず変更してください。

注意： Sybase Central でのマルチプレックス・サーバの作成時に [Admin スクリプトの生成] をオンにすると、`start_server` スクリプトと `sync_server` スクリプトが作成されます。start_iq および dbbackup の代わりにこれらのスクリプトをコマンド・ラインから実行できます。

サーバ・ステータスのチェック (Sybase Central)

最初のセカンダリ・サーバを追加すると、データベースをマルチプレックスで使用できるようになります。必要に応じて、すべてのマルチプレックス・ノードのステータスをチェックできます。

1. マルチプレックス・フォルダを開きます。
2. [サーバ] タブを選択して、サーバのステータスを表示します。

表 4 : [サーバ] タブの内容

カラム	説明
Name	サーバ名。
Role	コーディネータ、リーダ、またはライタ。
IsConnected	サーバが接続している場合は Yes。それ以外の場合は No。Sybase Central がサーバに接続しているかどうかを示します。Sybase Central は、マルチプレックス内の 1 台のサーバへの接続のみを必要とします。ただし、Sybase Central を複数のサーバに接続することもできます。
Failover node	このノードが指定済みのフェールオーバ・ノードである場合は Yes。それ以外の場合は No。
Sync State	同期済み、非同期 (有効な状態ではありません)、除外済み、コーディネータ、または実行されていません。
As Of	表示された情報の最終更新時刻。次に例を示します。"8/5/08 7:01:18 PM"

3. ステータス表示はその瞬間のステータスを正しく表しているとは限らず、ネットワークの遅延時間に依存します。ステータス表示をリフレッシュするには、メイン・メニュー・バーから [ビュー] - [フォルダの再表示] を選択します。

注意： マルチプレックス環境では、RESOURCE 権限のあるユーザには、[マルチプレックス] フォルダではなく [サーバ] フォルダにコーディネータが表示されます。

参照：

- sp_iqmpxinfo プロシージャ (145 ページ)

Sybase IQ Agent

Sybase Central では、Sybase IQ Agent を介して、Sybase IQ サーバを管理するのに必要な機能を実行できます。

「エージェント」は、クライアントの代わりに動作するプロセスで、多くの場合、リモート・マシンで実行します。

Sybase IQ Agent を使用すると Sybase Central の以下のような機能を使用できます。

- サーバの起動／停止
- ログ・ファイルへのアクセス
- システム関数の実行

参照：

- UNIX システムでの Sybase Central の起動 (35 ページ)
- UNIX 上での Sybase IQ Agent の実行 (34 ページ)
- Windows での Sybase Central の起動 (35 ページ)

Sybase IQ Agent ログのロケーション

Sybase IQ Agent ログのパスは、プラットフォームによって異なります。

UNIX システムでは、デフォルトの Sybase IQ Agent ポート (1099) が使用されている場合のログ名は `$IQDIR15/logfiles/SybaseIQAgent1099.nnn.log` です。ここで、*nnn* は、ディレクトリを最後に消去した後に Agent を起動した回数です。 たとえば、最後の消去後に Agent を 3 回起動した場合は、`$IQDIR15/logfiles/SybaseIQAgent1099.003.log` となります。

Windows システムでは、ユーザが Sybase IQ Agent のポートを明示的に設定した場合のログ名は、デフォルトで `%ALLUSERSPROFILE%¥SybaseIQ¥logfiles¥SybaseIQAgent_pppp.nnn.log` になります。ここで、*pppp* はポート番号で、*nnn* は、ディレクトリを最後に消去した後に Agent を起動した回数です。 例を示します。`C:¥Documents and Settings¥All Users¥SybaseIQ¥logfiles¥SybaseIQAgent_7899.001.log`.

UNIX 上での Sybase IQ Agent ログの移動

IQ Agent ログ・ファイルを異なるディレクトリに生成するには、ログ・ディレクトリ環境変数を設定します。
UNIX システムでは、$IQLOGDIR15 環境変数を変更先のディレクトリのフル・パスに設定します。たとえば、C シェル (csh) の場合は、次のように入力します。

```
setenv IQLOGDIR15 /work/mydir/agentlogs
```

Windows 上での Sybase IQ Agent ログの移動

Agent ログ・ファイルを異なるディレクトリに生成するには、ログ・ディレクトリ環境変数を設定します。

1. [コントロール パネル] - [システム] - [詳細設定] - [環境変数] を使用します。
2. [システム環境変数] にある [新規] をクリックします。
3. *[変数名]*に IQLOGDIR15、*[変数値]*にディレクトリのフル・パスを入力します。
4. [OK] をクリックします。

Sybase IQ Agent ログの表示 (Sybase Central)

Sybase Central で何らかの問題が発生する場合は、Agent のログ・ファイルにエラーまたはメッセージが記録されていないかどうかを確認します。
サーバ名を右クリックし、[エージェント] - [エージェント・ログの表示] を選択します。

Windows 上での Sybase IQ Agent の起動のトラブルシューティング

Sybase IQ Agent が実行されていないことを示すエラーが表示された場合は、Agent を起動します。

1. [コントロール パネル] の [サービス] ユーティリティを開きます。
2. Sybase IQ Agent の状態が [開始] でない場合は、[開始] をクリックします。
3. Agent を起動してから、Sybase Central を再起動します。

UNIX 上での Sybase IQ Agent の起動のトラブルシューティング

UNIX 上で Agent が実行されているかどうかを確認するには、**stop_iq** ユーティリティを実行します。

1. **-agent** 引数を指定して **stop_iq** コマンドを入力します。

```
stop_iq -agent
```

```
Checking system ...

The following 1 agent(s) are owned by 'fiona'
## Owner     PID     Started  CPU Time Additional Information
-- --------- ------- -------- -------- ----------------------
- 1: fiona 6669 Sep.01 5:11 PORT:1100 java
-Diq.agent=/work/sybase15/
IQ-15_3/java/IQAgent15.jar
-Di q.agent_log=/

-- Do you want to stop the agent displayed above <Y/N>?
```

この出力は、ユーザ `fiona` が Agent を実行していることを示しています。

2. Agent を停止するには、次のように、「はい」を意味する "Y" を入力します。`Y`
3. Agent が実行されていないことを示すエラーが表示された場合は、ディレクトリを `$IQDIR15/bin64` (またはプラットフォームに応じて `$IQDIR15/bin32`) に切り替えて、`S99SybaseIQAgent15` と入力して Agent を再起動します。

Windows 上での Sybase IQ Agent のパーミッションの設定

Windows では、デフォルトでシステム・アカウントが Agent を起動するので、Sybase IQ Agent の所有者を変更する必要があります。

Agent を起動するユーザは、マルチプレックス・データベースとサーバ・ディレクトリの作成者かつ所有者で、マルチプレックス内のすべてのサーバについて書き込み権限を所有している必要があります。

1. [管理ツール] - [サービス] を選択します。
2. [Sybase IQ Agent] を右クリックし、[開始] を選択します。
3. [全般] タブで、[スタートアップの種類] の [自動] を選択します。
4. [ログオン] タブで、[ログオン] オプションを [ローカル システム アカウント] から [アカウント] に変更します。
5. または、適切な権限を持つアカウントをブラウズして選択し、[OK] をクリックします。
6. そのアカウントのパスワードを入力し、再確認して、[OK] をクリックします。

UNIX 上での Sybase IQ Agent のパーミッション

Sybase IQ Agent は、適切な権限を持つアカウントから起動する必要があります。

`S99SybaseIQAgent15` スクリプトの所有者は、すべてのサーバを実行するときに使用する UID と同じにしてください。root ユーザ・アカウントは使用しないでください。

Windows 上での Sybase IQ Agent のポート番号の上書き

特定のホスト上の Sybase IQ Agent は、ユニークなポート番号を必要とします。Windows 上でサービス マネージャーを使用して、デフォルトの Sybase IQ Agent ポート番号を上書きします。

1. Sybase IQ Agent 15 という名前を選択します。
2. メニュー・バーから [操作] - [プロパティ] を選択します。

   [サービスの状態] の下の [停止] をクリックし、エージェントを停止します。

   [開始パラメータ] テキスト・ボックスに `-port nnnn` と入力します。nnnn はポート番号です。

[開始] をクリックしてエージェントを再開します。

Windows 上でエージェントが開始しない場合は、イベント・ログで診断情報を確認します。

UNIX 上での Sybase IQ Agent のポート番号の上書き

特定のホスト上の Sybase IQ Agent は、ユニークなポート番号を必要とします。

次の例のように、Sybase Central の起動コマンドで -DIQPORT パラメータを指定します。

```
scjview -DIQPORT=3356
```

ポート番号を変更することにより、特定のホストで複数の現在のバージョンの Sybase IQ Agent を実行したり、同じホスト上で Sybase IQ 12.7、15.0、15.1、15.2、15.3、15.4 の Agent を実行したりできます。

Sybase IQ Agent Windows サービスのアンインストールと再インストール

Windows 上の Sybase IQ Agent で問題が発生している場合は Sybase IQ Agent Windows サービスのアンインストールと再インストールを試してみます。

1. Sybase IQ Agent 15 Windows サービスをアンインストールするには、次のコマンドを実行します。

   ```
   SybaseIQagent15.exe -u "SybaseIQAgent15"
   ```

2. Sybase IQ Agent 15 Windows サービスを再インストールするには、次のコマンドを実行します。

   ```
   SybaseIQagent15.exe -i "SybaseIQAgent15"
   ```

Windows 上でのリモート・ファイル参照の無効化

Sybase IQ Agent では、デフォルトでリモート・ファイル参照がサポートされています。 セキュリティ上の不安がある場合は、参照機能を無効にしてください。

1. DOS シェルで、環境変数 **IQAGENTNOBROWSE** を任意の値に設定します。

   ```
   SET IQAGENTNOBROWSE=1
   ```

   ```
   SET IQPORT=2525
   ```

2. DOS シェルを起動したウィンドウで Agent を起動します。

   ```
   SybaseIQAgent15.exe -r "SybaseIQAgent15"
   ```

UNIX 上の Sybase IQ Agent でのリモート・ファイル参照の無効化

Sybase IQ Agent では、デフォルトでリモート・ファイル参照がサポートされています。 セキュリティ上の不安がある場合は、参照機能を無効にしてください。

UNIX 上で参照機能を無効化するには、次の手順を実行します。

-nobrowse パラメータを指定して Agent を起動します。

```
$IQDIR15/bin64/S99SybaseIQAgent15 -nobrowse
```

IQAGENTNOBROWSE の設定

Sybase IQ Agent では、デフォルトでリモート・ファイル参照がサポートされています。

セキュリティに不安がある場合は、IQAGENTNOBROWSE 環境変数を設定して、参照を無効にします。この方法は、**-nobrowse** 起動パラメータの代替策として使用できます。

Windows 上での IQ Agent の設定

Windows システムでは、Sybase IQ のインストールによって Windows サービス Sybase IQ Agent 15 がインストールされますが、エージェント・サービスを正しく機能させるには、マシンを再起動する必要があります。

UNIX での Sybase IQ Agent の自動起動の設定

システムを再起動するたびに、Sybase IQ Agent が自動的に起動するように設定できます。

Sybase IQ Agent の起動操作を繰り返し行わなくても済むようにするために、自動起動を有効にします。

1. Sybase IQ Agent の自動起動を有効にするには、次のファイルを UNIX 起動ディレクトリ (通常は `/etc/rc*`) に配置します。

   ```
   $IQDIR15/bin64/S99SybaseIQAgent15
   ```

2. Sybase IQ をインストールし、ファイルが配置されると、システムをリブートするたびに Agent が自動的に起動します。

リモートの Sybase IQ Agent

Sybase IQ Agent がリモート・サーバに存在する場合、Agent を起動するにはそのサーバのホスト名を指定する必要があります。

スクリプトを使用して Sybase IQ Agent を起動する場合は、オプションの **-host** パラメータを **S99SybaseIQAgent15** コマンドに使用してホスト名を明示的に指定します。

uname -n で返されるホスト名を使用して Agent を起動するには、次のように、オプションの **-host** パラメータを使用します。

```
S99SybaseIQAgent15 -host
```

-host パラメータを完全に省略した場合は、この動作がデフォルト動作となります。

ホストのエイリアスを使用して Agent を起動するには、次のように入力します。

```
S99SybaseIQAgent15 -host <foo>
```

ここで、*foo* は `/etc/hosts` ファイル内のエイリアスです。

1 つの引数の代わりに、代替ホスト名または IP アドレスのリストをカンマ区切りで指定することもできます。このリストでは、スペースを使用できません。1 つのホストがユニークなアドレスを持つ複数のネットワーク・カードをサポートしている場合、複数の IP エイリアスを使用できます。次に例を示します。

```
S99SybaseIQAgent15 -host "host1.sybase.com,10.20.30.40,
host2,12.20.30.41"
```

問題を回避するには、短いホスト名のみを使用するか、3 つの構造体すべてを使用して Sybase IQ Agent を起動します。 たとえば、

```
S99SybaseIQAgent15 -host host2 -port 2099
```

または

```
S99SybaseIQAgent15 -host "ciaran,ciaran.sybase.com,
10.50.7.70"  -port 2199
```

Agent は次の順番でバインドします。

1. ローカル・ホスト名
2. ホストの IP アドレス
3. **-host** コマンド・ライン・パラメータで指定された項目

UNIX 上での Sybase IQ Agent の実行

マルチプレックス・データベースがある場合、リモート IQ サーバを完全に管理するには、*マルチプレックス内の各マシン*で Sybase IQ Agent が実行されている必要があります。

前提条件

Agent の起動中および停止中でも Sybase Central の実行は継続できます。

手順

次のコマンドを、コマンド・ラインまたはスクリプトで入力します。

```
S99SybaseIQAgent15
```

参照：

- Sybase IQ Agent (29 ページ)

Windows での Sybase Central の起動

マルチプレックスを管理するには、Sybase Central を起動する必要があります。Windows 上で Sybase Central を起動するには、[プログラム] メニューから [Sybase] - [Sybase IQ15.4] - [Sybase Central v6.1] を実行します。

参照：

- Sybase IQ Agent (29 ページ)

UNIX システムでの Sybase Central の起動

マルチプレックスを管理するには、Sybase Central を起動する必要があります。

1. Sybase IQ Agent を起動していない場合は、「UNIX での Sybase IQ Agent の実行」を参照してください。
2. UNIX 上で Sybase Central を起動するには、コマンド・プロンプトに次のコマンドを入力します。

```
% scjview
```

参照：

- Sybase IQ Agent (29 ページ)

サーバの管理 (Sybase Central)

Sybase Central は、Sybase IQ マルチプレックス・サーバを管理するためのグラフィカル・インタフェースです。

1. 目的のサーバを右クリックします。
2. メニューから目的の操作を選択します。

   ポップアップ・メニューから次の操作を実行できます。

   - IQ Agent のプロパティ (バージョン、ポート番号、タイムアウト設定) または Agent ログの表示。
   - 選択したサーバの切断または削除。
   - Open Interactive SQL。

     このオプションを使用すると、ログイン情報の入力を要求せずにユーザを接続します。
   - 選択したサーバの名前の変更、停止、または同期処理。サーバがコーディネータの場合、ホストの ping、名前の変更、またはサーバの停止ができます。
   - 管理スクリプトの生成。

このオプションを使用すると、サーバの起動または停止のためのオプションのスクリプトを作成または更新できます。 Sybase IQ の新しいリリースまたは更新をインストールしてからスクリプトを生成してください。

- 選択したサーバのプロパティの表示。
- コンソール・ログの表示。

注意： コーディネータ上でウィザード (起動、停止、および同期) を使用すると、マルチプレックス内のすべてのサーバに影響を与えることができます。

参照：

- 管理シェル・スクリプト (36 ページ)
- マルチプレックス・リファレンス (107 ページ)

管理シェル・スクリプト

管理スクリプトを使用することで、通常は Sybase Central を使用して定期的に実行している操作を自動化できます。

スクリプトで次を行うことができます。

- サーバの起動、停止、または同期化
- **dbisql** の起動

たとえば、スクリプトを使用して夜間にセカンダリ・サーバを同期させることができます。

`start_server` スクリプトと `sync_server` スクリプトは、データベース・ディレクトリにあります。Windows ホストでは、これらのスクリプトは `.bat` ファイルです。UNIX では、シェル・スクリプトです。**start_iq** と **dbbackup** の代わりにこれらのスクリプトをコマンド・ラインから実行できます。

各スクリプトをテキスト・エディタで開き、使い方についてのコメントを読んでください。

パスワードを必要とするスクリプトでは、パスワードを入力パラメータとして定義します。

参照：

- マルチプレックス・リファレンス (107 ページ)
- サーバの管理 (Sybase Central) (35 ページ)
- マルチプレックス・データベース・ファイル (18 ページ)

管理シェル・スクリプトの作成

Sybase Central のナビゲーション・ウィンドウ枠から、マルチプレックス・サーバ用の管理シェル・スクリプトを作成します。

1. サーバを右クリックします。
2. ポップアップ・メニューから [Admin スクリプトの生成] を選択します。

UNIX での管理シェル・スクリプトのスケジューリング

管理スクリプトをスケジュールして、夜間に実行するか、定期的に繰り返す必要があるタスクを実行します。

crontab システム・ユーティリティを使用して、管理タスクをスケジュールできます。

Windows での管理シェル・スクリプトのスケジューリング

管理スクリプトをスケジュールして、夜間に実行するか、定期的に繰り返す必要があるタスクを実行します。

[スタート] - [設定] - [コントロール パネル] - [タスク] を使用して、管理スクリプトをスケジュールします。

マルチプレックス・サーバの同期

通常は、DDL と DML の変更内容は Sybase IQ によってセカンダリ・サーバに伝達されるため、セカンダリ・サーバの同期では特定のサーバ管理操作に関する同期のみが必要となります。

「同期」とは、コーディネータの情報に基づいて、セカンダリ・サーバを更新するプロセスです。

マルチプレックスは、自動的に変更内容をコミットします。これには、グローバルなスキーマ変更、マルチプレックス上のすべてのサーバがアクセスできる IQ テーブル内のデータに加えられた変更なども含まれます。

次の状況でのみ、同期が必要になります。

- 新規セカンダリ・サーバの作成中
- コーディネータのバックアップからのリストア
- 除外されていたセカンダリ・サーバ、または長期間停止されていたセカンダリ・サーバの再起動
- シングルノード・モードでのコーディネータの実行後
- IQ_SYSTEM_MAIN DB 領域へのファイルの追加後

参照：

- カタログ DB 領域の追加 (62 ページ)
- カタログ DB 領域の削除 (62 ページ)

- IQ_SYSTEM_MAIN ファイル・パスの変更 (57 ページ)
- コーディネータ上の IQ_SYSTEM_MAIN への領域の追加 (58 ページ)
- 同じ場所にデータベースをリストアする場合の IQ ストア・データのリストア (103 ページ)
- 異なる場所にデータベースをリストアする場合の IQ ストア・データのリストア (101 ページ)

個別のセカンダリ・サーバの同期処理 (Sybase Central)

他のサーバでクエリを実行している間に、コーディネータのバージョンのデータベース・カタログを単一のセカンダリ・サーバにコピーします。

1. コーディネータが実行されていることを確認し、コーディネータを Sybase Central に接続します。
2. セカンダリ・サーバを同期するには、そのサーバを右クリックして [コントロール] - [同期] を選択します。他のサーバではクエリの実行を継続できます。

   Sybase Central は、右クリックされたセカンダリ・サーバを停止し、カタログ・ストアを置き換えた後に、サーバを再起動します。この操作中、コーディネータは実行を継続しています。

複数のサーバの同期 (Sybase Central)

コーディネータのバージョンのデータベース・カタログをセカンダリ・サーバにコピーします。

複数のサーバを同期するには、次の手順に従います。

1. 同期ウィザードを起動するためにマルチプレックス・フォルダを右クリックします。
2. コンテキスト・メニューで [コントロール] - [同期] を選択します。
3. ウィザード・ページで、同期するサーバを選択します。

参照：

- IQ_SYSTEM_MAIN の更新 (56 ページ)

サーバを起動するための params.cfg ファイルの編集 (Sybase Central)

params.cfg ファイルには、メイン・キャッシュとテンポラリ・キャッシュの設定、エラーのログ機能など、サーバ起動時の機能を制御するスイッチの値が保存されています。

前提条件

サーバを起動する前に、サーバのデータベース・ディレクトリにある `params.cfg` ファイルを確認し、必要に応じて編集します。

Sybase Central を使用してデータベースを起動する場合は、次の手順に従います。

手順

1. マルチプレックス・データベースを起動するときに使用する `params.cfg` ファイルのすべての **-n** スイッチを削除します。Sybase Central によってポート番号の入力が求められるので、**-x tcpip{port = }** スイッチを削除します。

   設定ファイルには、サーバに名前を付ける **-n**、またはデータベース名やパスを含めないでください。
2. 設定ファイルには `params.cfg` という名前を付ける必要があります。
3. 設定ファイルはデータベース (`.db`) ファイルと同じフォルダまたはディレクトリに配置する必要があります。

起動スイッチについては、『ユーティリティ・ガイド』を参照してください。`params.cfg` の詳細については、『システム管理ガイド：第 1 巻』を参照してください。

マルチプレックス・サーバの追加 (Sybase Central)

Sybase Central で次の手順に従って、マルチプレックス・サーバを追加します。

1. サーバのステータスをチェックします。セカンダリ・サーバを追加するには、コーディネータが実行されている必要があります。
2. マルチプレックス・フォルダでサーバ名を右クリックします。
3. ポップアップから [サーバの追加] を選択します。
4. サーバ作成ウィザードの手順を順番に実行します。

セカンダリ・サーバの追加後、初めてマルチプレックスを起動したときに、コーディネータが次のようなエラーを返す場合があります。

```
I. 12/11 12:50:08. Trying to start TCPIP link ...
 I. 12/11 12:50:08. Multiplex environment incorrect for this server
 I. 12/11 12:50:08. Please connect and run procedure sp_iqmpxvalidate
for help
```

このエラーは、セカンダリ・サーバの IQ_SYSTEM_TEMP にテンポラリ・ファイルがないために発生します。新しく追加したサーバの IQ_SYSTEM_TEMP にテンポラリ DB ファイルを追加する必要があります。このファイルを追加した後に、**sp_iqmpxvalidate** を実行して、このサーバでエラーが報告されないことを確認します。

マルチプレックス・サーバの名前の変更 (Sybase Central)

Sybase Central を使用して、マルチプレックス・サーバの名前を変更できます。

1. サーバ・プロパティ・シートを開きます。
2. [設定] タブでサーバ名を変更します。

または、サーバを右クリックし、コンテキスト・メニューで [コントロール] - [含める／除外] を選択します。

データベース・ファイル・パスの変更 (Sybase Central)

データベースを、たとえばより多くの領域のあるディスクに移動する必要がある場合、Sybase Central を使用してファイル・パスを変更できます。

1. サーバのプロパティ・シートを開きます。
2. [設定] タブで、データベース・ファイルのパスを変更します。

サーバの役割の変更 (Sybase Central)

Sybase Central を使用して、リーダ・サーバをライタ・サーバに変更できます。

1. サーバ・プロパティ・シートを開きます。
2. [一般] タブで、サーバ・ロール (読み込み/書き込み) を変更します。

ホストとポートの変更 (Sybase Central)

Sybase Central を使用して、マルチプレックス・サーバのホストとポートの情報を変更できます。

1. サーバのプロパティ・シートを開きます。
2. [設定] タブで、ホスト／ポート情報を変更します。

サーバの包含または除外 (Sybase Central)

マルチプレックス・セカンダリ・サーバを長期間にわたり停止する場合は、そのサーバをマルチプレックスから除外します。

サーバを除外すると、バージョンのクリーンアップ時にコーディネータがこのサーバを無視できます。サーバを除外しない場合、コーディネータはセカンダリ・ノード停止以降の古いバージョンの IQ オブジェクトをすべて保持する必要があります。このため、余分なディスク領域が使用されます。指定されているフェールオーバ・サーバは、マルチプレックス内に最後に残ったセカンダリ・ノードでないかぎり除外できません。コーディネータには包含／除外は適用されません。

1. サーバ・プロパティ・シートを開きます。
2. [一般] タブで、サーバを包含または除外します。

または、サーバを右クリックし、コンテキスト・メニューで [コントロール] - [含める／除外] を選択します。

フェールオーバ・ノードの指定 (Sybase Central)

現在のコーディネータが処理を継続できなくなった場合にコーディネータの役割を継続させるために、フェールオーバ・ノードを指定する必要があります。

1. Sybase Central のツリー・ビューでマルチプレックスの名前を右クリックします。
2. 指定するフェールオーバを選択します。

参照：

- 指定済みのフェールオーバ・ノード (44 ページ)
- sp_iqmpxinfo プロシージャ (145 ページ)

マルチプレックス・サーバの削除 (Sybase Central)

セカンダリ・サーバを削除すると、マルチプレックス設定からも削除されます。

1. セカンダリ・サーバを削除するには、そのサーバを右クリックして、ポップアップ・メニューから [サーバの削除] を選択します。
2. サーバを削除する前にウィザードが表示されます。
3. 必要に応じて関連ファイルを削除するオプション・ボタンをクリックし、[完了] をクリックします。

削除するサーバが停止されていない場合は、そのセカンダリ・サーバが使用しているファイルは Sybase Central で削除できない場合があります。

マルチプレックスの管理 (Sybase Control Center)

Sybase IQ サーバのモニタリングと管理に Web ベースのツールを使用したい場合は、Sybase Control Center を使用します。

Sybase Control Center では、セカンダリ・サーバの追加、削除、または同期を行うことができます。また、セカンダリ・サーバをマルチプレックスに含めたり、マルチプレックスから除外したりもできます。また、Sybase Control Center を使用して、シンプレックスをマルチプレックスに変換したり、マルチプレックス・フェールオーバを実行したりできます。

SCC の『Sybase Control Center for Sybase IQ オンライン・ヘルプ』または http://sybooks.sybase.com/nav/summary.do?prod=10680 を参照してください。

管理用権限の使用

管理用権限を使用すると、適切に定義された一連のデータベース管理タスクを実行できます。

シンプレックス・サーバとマルチプレックス・サーバの両方に影響する権限の概要については、『システム管理ガイド：第 1 巻』の「ユーザ ID とパーミッションの管理」>「データベースのパーミッションと権限の概要」を参照してください。

MULTIPLEX ADMIN 権限

MULTIPLEX ADMIN 権限を使用すると、DBA 権限のないユーザがマルチプレックス管理タスクを実行できます。

次のことを行うには、MULTIPLEX ADMIN 権限または DBA 権限が必要です。

- マルチプレックスの名前の変更と、SYS.ISYSIQINFO システム・テーブルへのマルチプレックス名の保存。
- マルチプレックス・サーバの設定の変更。
- マルチプレックス・サーバの作成。
- マルチプレックスからのサーバの削除。

MULTIPLEX ADMIN 権限を他のユーザに付与できるのは、DBA または PERMS ADMIN のみです。

マルチプレックス・プロシージャを使用したセキュリティの調整

セキュリティを完全にするには、基本となるテーブルへのアクセスをすべて禁止し、ユーザまたはユーザのグループには、特定のストアド・プロシージャを実行するパーミッションだけを付与します。この方法であれば、データベースのデータを定義できるユーザを厳密に定義できます。

前提条件

特定の権限を持つユーザが、IQ システム・プロシージャを使用して特定のタスクの管理をできるようにするには次のようにします。

手順

1. 対象の各権限用にグループを作成します。
2. 指定したグループに権限を付与します。

3. グループに、その権限のタスクを実行する IQ プロシージャの EXECUTE パーミッションを付与します。

権限を付与する新しいユーザを作成したら、その権限用に作成したグループに対するメンバシップをこのユーザに付与します。ほとんどの権限はグループ・メンバシップを通じて継承されるため、ユーザは権限と IQ プロシージャの実行パーミッションをグループから継承します。

関連ストアド・プロシージャを実行するためのパーミッションをユーザに付与する

権限を付与する新しいユーザを作成したら、その権限用に作成したグループに対するメンバシップをこのユーザに付与します。ほとんどの権限はグループ・メンバシップを通じて継承されるため、ユーザは権限と IQ プロシージャの実行パーミッションをグループから継承します。

前提条件

記載されている例外の場合を除き、次の手順には DBA 権限または PERMS ADMIN 権限が必要です。

MULTIPLEX ADMIN 権限と、マルチプレックス管理に関連するストアド・プロシージャの実行パーミッションを `user1` という名前のユーザに付与するには、次のようにします。

手順

1. DBA 権限を持つユーザ、または USER ADMIN 権限と PERMS ADMIN 権限を持つユーザとしてデータベースに接続します。
2. `MPXADMIN_GRP` という名前のグループを作成します。次のように記述できます。

   ```
   CREATE USER MPXADMIN_GRP
   GRANT GROUP TO MPXADMIN_GRP
   ```

   または

   ```
   call sp_addgroup('MPXADMIN_GRP')
   ```

3. MULTIPLEX ADMIN 権限を `MPXADMIN_GRP` に付与します。

   ```
   GRANT MULTIPLEX ADMIN TO MPXADMIN_GRP
   ```

4. ユーザ管理用の Sybase IQ ストアド・プロシージャの EXECUTE パーミッションを `MPXADMIN_GRP` に付与します。

   ```
   GRANT EXECUTE on sp_iqmpxgetconnversion
   to MPXADMIN_GRP
   GRANT EXECUTE on sp_iqmpxinfo
   to MPXADMIN_GRP
   ```

```
GRANT EXECUTE on sp_iqmpxincconnpoolinfo
to MPXADMIN_GRP
GRANT EXECUTE on sp_iqmpxincheartbeatinfo
to MPXADMIN_GRP
GRANT EXECUTE on sp_iqmpxvalidate
to MPXADMIN_GRP
GRANT EXECUTE on sp_iqmpxversioninfo
to MPXADMIN_GRP
```

5. グループ `MPXADMIN_GRP` のメンバシップを `user1` に付与します。`user1` は `MULTIPLEX ADMIN` 権限と割り当てられている IQ プロシージャの実行パーミッションを `MPXADMIN_GRP` グループのメンバシップを通じて継承します。

```
GRANT MEMBERSHIP IN GROUP MPXADMIN_GRP TO user1
```

マルチプレックスのログイン管理

ログイン・ポリシーは、ユーザが実行できる機能とデータベースへの接続性を管理します。

Sybase IQ ログイン・ポリシーの概要については、『システム管理ガイド：第 1 巻』の「ユーザ ID とパーミッションの管理」>「ログイン管理」 を参照してください。

コーディネータの障害

現在のコーディネータ・ノードが障害により実行できない場合、または、メンテナンスのために停止する必要がある場合、マルチプレックス全体が読み取り専用の状態になります。この状態では、既存の IQ データのクエリは実行できますが、変更はできません。

読み取り／書き込み機能を再度確立するには、別のサーバをコーディネータとして昇格する必要があります。この操作は、*手動フェールオーバ*と呼ばれます。

参照：

- DROP MULTIPLEX SERVER 文 (121 ページ)
- マルチプレックス・サーバの削除 (Interactive SQL) (25 ページ)

指定済みのフェールオーバ・ノード

マルチプレックスには、現在のコーディネータが実行できない場合にコーディネータの役割を継承する指定済みのフェールオーバ・ノードが必要です。

手動フェールオーバを行う場合は、フェールオーバ・ノードを使用する必要があります。マルチプレックスの作成時に、最初に作成されたセカンダリ・サーバが

指定済みのフェールオーバ・ノードとなりますが、後で別のノードをフェールオーバ・ノードとして指定することもできます。

sp_iqmpxinfo プロシージャを使用すると、指定済みのフェールオーバ・ノードを表示できます。

フェールオーバは、コマンド・ライン、Sybase Central、または Sybase Control Center から実行できます。Web ベースの管理ツールを使用したい場合は、Sybase Control Center を使用して、フェールオーバ・ノードの指定またはフェールオーバの実行を行います。

SCC の『Sybase Control Center for Sybase IQ オンライン・ヘルプ』または http://sybooks.sybase.com/nav/summary.do?prod=10680 を参照してください。

参照：

- sp_iqmpxinfo プロシージャ (145 ページ)
- フェールオーバ・ノードの指定 (Sybase Central) (41 ページ)
- フェールオーバ・ノードの指定 (Interactive SQL) (24 ページ)
- マルチプレックス・トポロジの表示 (85 ページ)
- マルチプレックス・トポロジの印刷 (86 ページ)

フェールオーバの要件

以前のコーディネータ・プロセスが実行していないことを確認してから、フェールオーバを行う必要があります。

最悪のシナリオとして、以前のコーディネータ・コンピュータが実行されているにもかかわらずネットワークに接続されていない場合やハードウェアが休眠状態にある場合が考えられます。このような場合、コーディネータ・コンピュータにログオンできず Sybase Central のようなツールは使用できませんが、突然通常の動作を開始する可能性があります。手動のフェールオーバ・プロセスを実行するときには、それまでコーディネータが実行されていたコンピュータを停止するのが理想的です。

警告！ 以前のコーディネータ・プロセスが有効な状態で手動フェールオーバを開始すると、データベースが破壊される可能性があります。

参照：

- サーバの同期 (コマンド・ライン) (26 ページ)
- コーディネータの置き換え (コマンド・ライン) (46 ページ)
- マルチプレックス・トポロジの表示 (85 ページ)
- マルチプレックス・トポロジの印刷 (86 ページ)

コーディネータの置き換え (Sybase Central)

コーディネータを置き換える前に、コーディネータがほんとうに停止していることを確認します。

1. Sybase Central のツリー・ビューでマルチプレックスのセット・ノードを右クリックします。フェールオーバ・ウィザードは、コーディネータがダウンしている場合で指定済みのフェールオーバ・ノードが実行されている場合のみに表示されます。
2. ドロップダウン・リストの [削除します] (デフォルト)、[リーダとして保存]、または [ライタとして保存] という 3 つのオプションのいずれかを選択して、 現在のコーディネータに対して実行するアクションを指定します。

   サーバを削除することを選択した場合、[サーバ・ファイルの削除] チェック・ボックスが表示されます (デフォルトはオフ)。

   サーバをリーダまたはライタとして維持することを選択した場合、 [包含済み] と [除外済み] (デフォルト) という 2 つのラジオ・ボタンが表示されます。

   [包含済み] または [除外済み] (デフォルト) を選択します。[包含済み] を選択すると、[フェールオーバ後に同期] チェック・ボックスが表示されます。このチェック・ボックスは、デフォルトではオフになっています。
3. [新しいフェールオーバ・ノードを指定] ドロップダウンから新しいフェールオーバ・ノードを選択します。
4. [完了] をクリックしてフェールオーバ・プロセスを開始します。

   2 つのダイアログ・ボックスが表示されます。
5. コーディネータがダウンしていることを確認済みでフェールオーバを開始する準備が整っている場合は [はい] をクリックします。ウィザード画面の一番下に、複数の進行メッセージが表示されます。

コーディネータの置き換え (コマンド・ライン)

コーディネータを置き換える前に、コーディネータがほんとうに停止していることを確認します。

前提条件

置き換えの手順を開始するには、コーディネータ・プロセスが非稼動である必要があります。指定されているコーディネータ・ノードは、マルチプレックスの一部を構成している必要があります。フェールオーバ・ノードにリーダを指定することをおすすめします。リーダには、処理待ちの書き込みトランザクションがないため、フェールオーバの実行がより簡単です。

手順

1. コーディネータ・プロセスが非稼動であることを確認します。

   元のコーディネータの停止中にセカンダリ・ノードで実行中のすべての読み取り／書き込みトランザクションは、ロールバックされます。それまでのコーディネータが専用のサーバ・ハードウェアで実行されていた場合にフェールオーバ・プロセスを実行する場合は、そのコンピュータを停止するのが理想的です。

   - UNIX では、コーディネータ・マシンにログインし、環境変数が設定されていることを確認したうえで次のコマンドを実行します。

     ```
     stop_iq
     ```

     その後、該当する iqsrv15 プロセスを停止します。
   - Windows では、コーディネータ・マシンにログインし、 タスク マネージャーを起動して、`iqsrv15.exe` という名前のプロセスを探します。`iqsrv15.exe` プロセスを停止します。
2. 指定済みのフェールオーバ・ノードを識別するには、実行中の任意のマルチプレックス・サーバに接続し、ストアド・プロシージャ **sp_iqmpxinfo** を実行します。カラム `coordinator_failover` に、指定済みのフェールオーバ・ノードが表示されます。
3. 指定済みのフェールオーバ・ノードに接続し、**COMMIT** を実行してから **BEGIN TRANSACTION** を実行して、このノードが最新の TLV ログの内容で更新されている状態にします。

   Sybase Central (右クリック - [コントロール] - [停止]) または **dbstop** ユーティリティを使用して指定済みのフェールオーバ・ノードを完全に停止します。
4. コマンド・ラインで、次のようにフェールオーバ・スイッチ (**-iqmpx_failover 1**) をサーバ起動ユーティリティで使用して目的とするコーディネータを再起動します。

   ```
   start_iq -STARTDIR/host1/mpx
   @params.cfg -iqmpx_failover 1
   -n mpxnode_w1 -x "tcpip{port=2764}"
   mpxtest.db
   ```

サーバの起動が完了するとフェールオーバ・プロセスが完了し、指定済みのフェールオーバ・ノードが新しいコーディネータ・ノードとなります。フェールオーバが完了してから実行されるトランザクションでは、新しいコーディネータが他のセカンダリ・サーバによって検知され、すべての読み取り／書き込みトランザクション用に接続が実行されます。以前のコーディネータはリーダとなり、通常のセカンダリ・ノードとして起動できます。

以前のコーディネータを起動するには、新しいコーディネータに対して同期する必要があります。「サーバの同期 (コマンド・ライン)」の手順 1 ～ 4 を実行しま

す。ただし、手順 2 (dbbackup) では、**-c** パラメータを使用して指定される接続文字列に新しいコーディネータの接続パラメータを使用する必要があります。

参照：

- フェールオーバの要件 (45 ページ)

コーディネータの置き換え (Sybase Control Center)

マルチプレックス・フェールオーバの管理に Web ベースのツールを使用したい場合は、Sybase Control Center を使用します。

Sybase Control Center では、フェールオーバ・ノードを指定したり、コーディネータ・ノードのフェールオーバを実行したりできます。

SCC の『Sybase Control Center for Sybase IQ オンライン・ヘルプ』または http://sybooks.sybase.com/nav/summary.do?prod=10680 を参照してください。

マルチプレックス・トランザクション

共有オブジェクトを変更するマルチプレックス・トランザクションは、特定のルールに従って動作します。

マルチプレックス・トランザクションの概要

トランザクションには、ローカルとグローバルの 2 つのタイプがあります。

ローカル・トランザクション

*ローカル・トランザクション*とは、共有オブジェクトを変更しないすべてのトランザクションです。

ローカル・トランザクションは、読み取り専用または読み取り／書き込みの可能性がありますが、データの変更はローカル・オブジェクトのみ (SA テーブルまたはテンポラリ・テーブル) に実行されます。ローカル・トランザクションは、任意のマルチプレックス・ノードで実行でき、その影響はそのノードのみで参照できます。

グローバル・トランザクション

*グローバル・トランザクション*とは、共有オブジェクト内のデータまたは永続オブジェクトのスキームを変更するすべてのトランザクションです。グローバル・トランザクションは、コーディネータ・ノードまたはライタ・ノード上のみで実行できます。グローバル・トランザクションの影響は、マルチプレックス内のすべてのノードで参照できます。

すべてのマルチプレックス・トランザクションは、ローカル・トランザクションとして開始されます。最初の読み取り／書き込み操作 (挿入など) が共有 IQ オブジェクトで開始されると、そのトランザクションはグローバル・トランザクションとなります。グローバル・トランザクションがセカンダリ・ライタ・ノードから開始される時点で、コーディネータ・ノードが実行されている必要があります。これは、ライタ・ノードがトランザクションのグローバルな側面をコーディネータ・ノードと通信する必要があるためです。

ライタで開始されたグローバル・トランザクションの読み取り／書き込み操作の処理にはライタ・ノード CPU とローカル・ディスクが使用されますが、グローバル・トランザクション ID の割当て、グローバル・ロック管理、TLV ログの書き込みなど、そのトランザクションのグローバルなトランケーション・ポイントの書き換えの処理はコーディネータ・ノードで実行されます。

グローバル・トランザクション処理時のコーディネータの障害

ライタによって開始されたグローバル・トランザクションの実行中にコーディネータに障害が発生すると、コーディネータ・ノードが応答していないことを示すエラー・メッセージが表示されます。その時点のグローバル・トランザクションの状態によって、次のいずれかの状態が発生します。

- トランザクションの開始時にこのエラーが発生した場合、現在のコマンドのみがロールバックし、トランザクションは継続されます。
- グローバル・トランザクションがすでに開始されている場合でコミットされる前にこのエラーが発生した場合、トランザクションはコミットできず、ロールバックされます。
- グローバル・トランザクションのコミット中にこのエラーが発生した場合、そのユーザの接続が終了されます。

注意： ライタ・ノードで開始されたグローバル・トランザクションがグローバルで永続オブジェクトとローカルで永続オブジェクトの両方を変更する場合 (SA ベースのテーブルや IQ ベースのテーブルなど) に、コミット中にコーディネータに障害が発生すると、グローバル・オブジェクトの変更はコミットされる一方、ローカル・オブジェクトの変更は喪失する場合があります。これは、トランザクションのローカルなコンポーネントとグローバルなコンポーネント両方をコミットするために「最大限の努力」が行われる、同じトランザクションでローカル・テーブルとプロキシ・テーブルの両方を更新するシナリオと一貫した動作です。

DML コマンド

IQ 15.0 では、グローバル・トランザクションは、コーディネータ・サーバと任意のライタ・サーバから実行できます。共有 IQ ストア内のテーブルを変更する DML コマンドは、グローバル DML コマンドと呼ばれます。グローバル DML コマンドを含むコマンドは、すべてグローバル・トランザクションとなります。

グローバル DML コマンド

グローバル DML コマンドは、コーディネータ上で実行されたように動作し、シングル・サーバ上でのロックのルールに従います。

たとえば、任意のサーバ上のトランザクションの 1 つが共有 IQ テーブルを変更した場合、その変更がコミットまたはロールバックされるまでは別のトランザクションによる変更はまったくできなくなる可能性があります。グローバル・トランザクションがコミットされると、ライタ・ノード上で実行されているか、コーディネータ上で実行されているかにかかわらず、そのグローバル・トランザクションのメタデータは TLV ログを使用してマルチプレックス内のすべてのサーバに通信されます。

テーブル・データ・スコープ

DML コマンドをマルチプレックスで実行している場合、テーブル・ローが参照できるかどうかはテーブルのタイプによって異なります。

マルチプレックスには、次の 3 つのタイプのロー参照があります。

- グローバル・スコープ – すべてのサーバ上のすべての接続がローを参照できます。
- サーバ・スコープ – 特定のマルチプレックス・サーバ上のすべての接続がローを参照できます。
- 接続スコープ – 特定のマルチプレックス・サーバ上の単一の接続のみがローを参照できます。

表 5 : マルチプレックス・テーブル・データ・スコープ

テーブル・タイプ	データ・スコープ
IQ ベース	グローバル
IQ テンポラリ	接続
グローバル・テンポラリ・テーブル	接続
SA カタログ (SYSTEM に作成されたテーブル)	サーバ
SA テンポラリ (SYSTEM に作成されたテーブル)	接続

Sybase IQ は、セカンダリ・ノードが同期されている場合に、コーディネータ・ノード上の SA カタログ・テーブル内のデータをセカンダリ・ノードのカタログ・ストアにコピーします。 コピーされない場合は、セカンダリ・サーバ・インスタンスとコーディネータ・サーバ・インスタンスのカタログ・ストア・テーブル内のデータは関連付けられません。同期後は、セカンダリ・サーバ上のローカル SA テーブル・データは、コーディネータのテーブル・データで上書きされるためなくなります。

注意： CIS プロキシ・テーブルはリモート・サーバ上のデータをポイントしているため、データ・スコープが外部となります。このようなテーブルのデータ管理は、リモート・サーバによって行われます。

ジョイン・インデックス内のテーブル

ジョイン・インデックス内のテーブルには、いくつかの制限が適用されます。

15.0 以降のマルチプレックスでは、ジョイン・インデックスに参加しているテーブルのスキーマ変更はできません。ジョイン・インデックスに参加しているテーブル上で実行可能な DML 操作は、シングル・ノード・モードのコーディネータ上のみで実行できます。

参照：

- サーバ起動ユーティリティ (start_iq) (156 ページ)

DDL コマンド

コマンド・タイプとオブジェクト・タイプは、マルチプレックス内のデータ定義言語 (DDL) コマンドのスコープに影響を及ぼします。

DDL コマンドのスコープ

DDL 文は、すべてのノードに送信することも、実行しているノードに対してローカルに送信することもできます。

DDL コマンドには、次の 2 つのスコープがあります。

- ローカル・スコープ – ローカル・サーバ上で実行し、ローカル・カタログ・ストアまたはローカル・テンポラリ・ストアのみに影響します。
- グローバル・スコープ – コーディネータ上で実行し、共有 IQ ストアとグローバル・カタログ・ストアに影響します。コーディネータは、グローバル・スコープを使用してコミット時に文を TLV ログ上に書き込みます。

ローカル DDL コマンド

テンポラリ・オブジェクト (永続オブジェクト ID のないオブジェクト) に影響する DDL コマンドは、ローカルです。

ローカル・コマンドは、次のオブジェクト・タイプに影響します。

- ローカル・テンポラリ・テーブル
- ローカル・プロシージャ
- テンポラリ・オプション

グローバル DDL コマンド

`ISYSOBJECT` テーブル内の永続オブジェクト ID を作成、変更、または削除する DDL コマンドは、グローバルです。

グローバル・コマンドは、次のオブジェクト・タイプに影響します。

- テーブル (SA ベース・テーブル、IQ ベース・テーブル、グローバル・テンポラリ・テーブルを含む)
- ビュー
- マテリアライズド・ビュー (SA テーブルのみ)
- カラム
- インデックス

- プロシージャ
- イベント
- ユーザ
- パブリケーション
- リモート・タイプ
- ログイン・マッピング
- JAR
- Java クラス
- サービス
- データベース・オプション (スコープがローカルに指定されているオプションは例外として除く)

DDL コマンド依存性の例 1

コーディネータ・ノードが実行されていない場合、グローバル DDL コマンドは実行できません。実行を試みると、エラー `SQLCODE: -1004011, SQLSTATE QIA11: Coordinator node not responding` が返されます。

たとえば、テンポラリ関数またはストアド・プロシージャを次のように作成します。

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

続いて、次のようにテンポラリ関数に依存するビューの作成を試みます。

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

永続的なプロシージャではないため、エラー `Procedure 'f1' not found` が返されます。Sybase IQ では、マルチプレックス環境でこのような操作を行うことはできません。

DDL コマンド依存性の例 2

グローバルなオブジェクトの作成時は、そのオブジェクトが、ローカルなオブジェクトに依存しないようにする必要があります。

次の例は、ローカル・オブジェクトに依存するグローバル・オブジェクトを作成します。次のように、セカンダリ・ノードに `lineitem` テンポラリ・テーブルを作成するとします。

```
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) )
```

次に、BEGIN PARALLEL IQ コマンドを使用して、lineitem テンポラリ・テーブルのカラムにインデックスを作成します (インデックスはグローバル・オブジェクトです)。

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

Sybase IQ はエラー Table 'lineitem' not found を返します。これは、BEGIN PARALLEL IQ コマンドがコーディネータ・ノードに送信されるグローバル・コマンドであるのに対し、lineitem テーブルがセカンダリ・ノード上のローカル・テンポラリ・テーブルであるためです。

役割制限

文の中には、次のように特定の役割のノードに使用が制限されているものがあります。

次の制限が、文に適用されます。

- コーディネータ – コーディネータの役割制限のある文はコーディネータ・ノード上のみで実行できます。
- ライタ – ライタの役割制限のある文は、ライタ・ノードまたはコーディネータ上のみで実行できます。
- なし – 役割制限のない文は、マルチプレックス内の任意のノードで実行できます。

表 6 : 役割制限のあるコマンド

コーディネータの役割制限	ライタの役割制限
• IQ メイン・ストア DB 領域に対するすべての **CREATE/ALTER/DROP DBSPACE** コマンド • **BACKUP DATABASE** • **LOCK TABLE** • **sp_iqemptyfile**	IQ メイン・ストア DB 領域内のオブジェクトに影響するすべての DDL コマンド。これには、次に対する **ALTER/DROP** が含まれます。 • テーブル • シングルカラム・インデックスおよびマルチカラム・インデックス • テーブル制約

この表にない文には、役割制限はありません。

ローの保持

ON COMMIT PRESERVE ROWS を使用して作成されたグローバル・テンポラリ・テーブルは、削除を実行中のセカンダリ・ノードの接続にあるそのグローバル・テンポラリ・テーブルのインスタンスにデータがある場合、セカンダリ・ノードから変更または削除できません。

テーブルをトランケートして再度実行を試みるか、そのテーブルをコーディネータ・ノードから変更または削除します。 グローバル・テンポラリ・テーブルが ON COMMIT DELETE ROWS を使用して作成された場合、ローがある場合でもテーブルの変更または削除ができます。

たとえば、次のようにセカンダリ・ノードに接続します。

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

DROP 文は、次のエラーで失敗します。

```
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"
```

マルチプレックス内の DB 領域の更新

マルチプレックスでの DB 領域の更新は、シンプレックス・データベースの場合と多少異なります。

一般的に、共有 DB 領域の更新については、次の違いがあります。

- コーディネータのみが共有 IQ DB 領域を操作できます。
- 共有 DB 領域ファイルを読み取り専用モードにしてから、ファイルを削除する必要があります。 すべてのサーバは、**ALTER READONLY** 文に対応するバージョンに進んでおり、ファイルのすべての予約済みブロックが返されている必要があります。
- メイン DB 領域とカタログ DB 領域に対する **CREATE DBSPACE** コマンド、**ALTER DBSPACE** コマンド、**DROP DBSPACE** コマンドはコーディネータ上でのみ許可されます。
- IQ_SHARED_TEMP の動作については、「マルチプレックス・トランザクション」>「DDL コマンド」>「役割制限」>「マルチプレックス内の DB 領域の更新」>「IQ_SHARED_TEMP の更新」を参照してください。

TLV 送信とバージョン・メンテナンスの遅延によって、空の DB ファイルまたは DB 領域の削除ができるようになるまでに数分間の遅延が発生する場合があります。 ここで `Command not replayed` エラーが表示されます。**DROP DBSPACE** コマンドまたは **ALTER DBSPACE DROP FILE** コマンドを実行するには、**sp_iqdbspace** プロシージャと **sp_iqfile** プロシージャによって報告される `OkToDrop` カラムが“Y”を表示している必要があります。

-iqro を使用して接続しているときには、**ALTER DBSPACE DROP FILE** を入力しないでください。サーバが **-iqro** オプションを指定して起動されている場合、**ALTER DBSPACE DROP FILE** を入力すると読み込み専用データベースに対しての修正は禁止されていますというエラーが返されます。

IQ_SYSTEM_TEMP の更新

要件をよく理解してから、IQ_SYSTEM_TEMP DB 領域を更新してください。

IQ_SYSTEM_TEMP を更新する場合は、次の要件を満たす必要があります。

- テンポラリ DB 領域ファイルの作成、変更、削除は、それらが存在するノードからのみ実行できます。これらのコマンドの構文は、シンプレックス・テンポラリ DB 領域の場合と同じです。
- セカンダリ・サーバを同期してから、**ALTER DBSPACE ALTER FILE RENAME PATH** を実行する必要があります。同期は TLV ログの再生を通じて自動的に実行されます。ただし、この DB 領域上での直前の DB 領域操作 (作成または変更) と **ALTER DBSPACE ALTER FILE RENAME PATH** を実行できるようになるまでの間に最大で 2 分間の遅延が生じる場合があります。セカンダリ・サーバが同期されていないと、"Command not replayed" エラーが表示されることがあります。

DB 領域を更新する前に、DB 領域と DB ファイルの概要について、『システム管理ガイド： 第 1 巻』の「システム管理の概要」>「Sybase IQ 内のデータ記憶領域」を参照してください。

IQ_SYSTEM_MAIN の更新

IQ_SYSTEM_MAIN DB 領域は、使用中のブロックをリストするフリー・リストを含む重要なデータベース構造体を管理します。

セカンダリ・ノードを停止してから、領域を IQ_SYSTEM_MAIN に追加します。DB ファイルが IQ_SYSTEM_MAIN に追加されると、アクティブなセカンダリ・ノードは自動的に停止します。セカンダリ・ノードの IQ メッセージ・ファイルには、次が報告されます。

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

この動作は、IQ_SYSTEM_MAIN DB 領域に対してのみ適用されます。その他の DB 領域操作では、中断はなく、マルチプレックス内のすべてのノードは実行を継続します。「コーディネータ上の IQ_SYSTEM_MAIN への領域の追加」を参照してください。

IQ_SYSTEM_MAIN の更新については、次の制限が適用されます。

- セカンダリ・ノードからアクセスできないパスを持つ共有 IQ メイン DB 領域ファイルがある場合、セカンダリ・ノードは、パスが修正されるまでそのファ

イルまたはそのファイルの内容にアクセスできません。「共有ディスク・アレイ上のファイル」を参照してください。
- **ALTER DBSPACE ALTER FILE RENAME PATH** は、IQ_SYSTEM_MAIN で禁止されています。

IQ_SYSTEM_MAIN に新しいファイルを追加したら、セカンダリ・ノードを同期および再起動します。

参照：

- コーディネータ上の IQ_SYSTEM_MAIN への領域の追加 (58 ページ)
- 複数のサーバの同期 (Sybase Central) (38 ページ)
- サーバの同期 (コマンド・ライン) (26 ページ)

IQ_SYSTEM_MAIN ファイル・パスの変更

IQ_SYSTEM_MAIN 内のすべての DB ファイルのパスは、セカンダリ・ノードから参照できる必要があります。参照できない場合、セカンダリ・ノードは起動せず、`Error opening DBFILE 'filepath'` を報告します。

リンクを作成することで解決できない IQ_SYSTEM_MAIN ファイル・パスを変更するには、コーディネータ上で次の手順に従います。

1. マルチプレックス内のすべてのサーバを停止します。
2. **-iqmpx_sn 1** スイッチを使用して、シングル・ノード・モードでコーディネータを起動します。
3. DB ファイルを読み取り専用にします。

   ```
   ALTER DBSPACE IQ_SYSTEM_MAIN ALTER FILE
   mainfile READONLY
   ```
4. 次のコマンドを実行します。

   ```
   sp_iqemptyfile mainfile
   ```
5. DB ファイルを削除します。

   ```
   ALTER DBSPACE IQ_SYSTEM_MAIN DROP FILE
   mainfile
   ```
6. すべてのセカンダリ・ノードから参照できるパスのある DB ファイルを追加します。次に UNIX の例を示します。

   ```
   ALTER DBSPACE IQ_SYSTEM_MAIN ADD FILE
   mainfile '/dev/rdsk/c4t600A0B80005A7F5D0000024'
   ```

   次に Windows の例を示します。

   ```
   ALTER DBSPACE IQ_SYSTEM_MAIN ADD FILE  mainfile '¥¥¥¥.¥
   ¥PhysicalDrive1'
   ```
7. コーディネータを **-iqmpx_sn** 1 スイッチを使用せずに通常の手順で起動します。

8. セカンダリ・ノードを同期します。「マルチプレックス・サーバの同期」を参照してください。

IQ_SYSTEM_MAIN DB 領域は、使用中のブロックを追跡するフリー・リストを含む重要なデータベース構造体を管理します。IQ_SYSTEM_MAIN に容量を追加する前に、セカンダリ・ノードを停止することをおすすめします。15.0 以降のマルチプレックスでは、IQ_SYSTEM_MAIN に DB 領域が追加された場合、すべての実行中のセカンダリ・ノードは自動的に停止し、セカンダリ・ノードの IQ メッセージ・ファイルに次のようにレポートされます。

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

参照：

- マルチプレックス・サーバの同期 (37 ページ)

コーディネータ上の IQ_SYSTEM_MAIN への領域の追加

予約領域を含む IQ_SYSTEM_MAIN の領域不足がコーディネータに発生すると、データベースの破損を防止するために実行が中止され、コーディネータを通常どおりに起動できなくなる場合があります。

コーディネータが領域の不足により実行を中止した場合、**-iqmpx_sn** 起動スイッチを使用してコーディネータをシングル・ノード・モードで起動し、ファイルを IQ_SYSTEM_MAIN に追加して、より多くの領域を追加します。次のプロシージャにあるように、コーディネータを通常どおり再起動し、すべてのセカンダリ・ノードを同期します。

コーディネータが IQ_SYSTEM_MAIN 内の領域を必要とする場合は、次の手順に従います。

1. マルチプレックス内のすべてのサーバを停止します。
2. **-iqmpx_sn 1** スイッチを使用して、シングル・ノード・モードでコーディネータを起動します。

   ```
   start_iq @params.cfg mpxtest.db -iqmpx_sn 1 -n mpxnode_c
   ```

3. IQ_SYSTEM_MAIN にロー・デバイス上の新しいファイルを使用して領域を追加するには、次のような構文を使用します。

プラットフォーム	アクション
UNIX	`ALTER DBSPACE IQ_SYSTEM_MAIN ADD FILE mymainfile '/dev/rdsk/c4t600A0B80005A7F5D0000024'` ここで、`mymainfile` には、新しい DB ファイルの論理名または選択済みの論理名が入ります。
Windows	`ALTER DBSPACE IQ_SYSTEM_MAIN ADD FILE mymainfile '¥¥.¥PhysicalDrive3'`

4. コーディネータを **-iqmpx_sn 1** スイッチを使用せずに通常の手順で起動します。
5. セカンダリ・ノードを同期します。詳細については、「マルチプレックス・サーバの同期」を参照してください。

参照：

- IQ_SYSTEM_MAIN の更新 (56 ページ)
- マルチプレックス・サーバの同期 (37 ページ)

マルチプレックス・コーディネータ上の IQ_SYSTEM_TEMP の DB ファイルの置き換え

破損した DB ファイルを置き換えるには、ファイルを削除し、サーバを再起動した後に、DB ファイルをコーディネータに追加します。

1. **-iqnotemp** パラメータを使用して、データベースを起動します。

   -iqnotemp が指定されて実行中のデータベース上で実行可能なテンポラリ・ファイル操作は、1 つまたは複数のテンポラリ・ファイル定義を削除することのみです。
2. IQ_SYSTEM_TEMP の最後のファイルを削除します。
   ```
   ALTER DBSPACE IQ_SYSTEM_TEMP DROP FILE filename
   ```
3. データベースを停止します。
4. データベースをシングルノード・モードかつファイルなしで起動し、チェックポイント・ログを空にします。
   ```
   start_iq @params.cfg mpxtest.db -iqmpx_ov 1
   -iqmpx_sn 1 -n mpxnode_c
   ```
5. 「ダミー」DB ファイルをコーディネータに追加します。このファイルを削除し、コーディネータ・モードで再び追加する必要があります。これは、シングルノード・モードで追加されたファイルには、NULL のサーバ ID が付加されるためです。これらのファイルは、ファイルを追加したサーバではなく、コーディネータによって所有されます。IQ_SYSTEM_MAIN にロー・デバイス上の新しいファイルを使用して領域を追加するには、次のような構文を使用します。

プラットフォーム	構文
UNIX	`ALTER DBSPACE IQ_SYSTEM_TEMP ADD FILE tempfile '/dev/rdsk/c4t600A0B80005A7F5D0000024'` ここで、tempfile には、新しい DB ファイルの論理名または選択済みの論理名が入ります。
Windows	`ALTER DBSPACE IQ_SYSTEM_TEMP ADD FILE tempfile '¥¥.¥PhysicalDrive3'`

6. サーバを停止し、再起動します。
7. ダミー・ファイルを削除します。

IQ_SHARED_TEMP の更新

IQ_SHARED_TEMP DB 領域を更新する場合には、いくつかの制限が適用されます。

次のルールが IQ_SHARED_TEMP DB 領域の更新操作に影響します。

- コーディネータのみが共有 IQ DB 領域を操作できます。
- コーディネータをシングルノード・モードで起動し、その後に、ファイルを IQ_SHARED_TEMP から削除します。IQ_SHARED_TEMP で読み取り／書き込みになった最初のファイルを、最後に削除する必要があります。シンプレックスの IQ_SHARED_TEMP のファイルも削除できます。
- IQ_SHARED_TEMP ストアを更新するには、SPACE ADMIN 権限が必要です。
- **ALTER FILE RENAME PATH** は、IQ_SHARED_TEMP DB 領域に対して実行できません。
- **ALTER DBSPACE ALTER FILE READONLY** は、IQ_SHARED_TEMP DB 領域に対して実行できません。

共有ファイル・ステータス・データ

データベースを開いて特定のコマンドを実行する場合、マルチプレックス・ノードは各共有 DB 領域ファイルのファイル・ステータスを保持します。

コーディネータがすべてのノードのステータス・データを保持し、各セカンダリ・ノードがその独自のファイルのステータス・データを保持します。

構成しているすべてのサーバは、新しい IQ_SHARED_TEMP ファイルの有効な読み取り／書き込みステータスを返す必要があります。そうでない場合、文はエラーを返します。 すべてのノードが新しく追加されたファイルにアクセスできる場合、各ノードは有効なステータス・メッセージを返します。 すべてのセカンダリ・サーバは、すべての共有ファイルのファイル・ステータスをコーディネータに返します。Sybase IQ は、すべての共有 DB 領域の DB ファイルの共有ファイ

ル・ステータス情報を保持しますが、使用するのは、IQ_SHARED_TEMP DB 領域をターゲットにする DB 領域更新コマンドにおいてのみです。

セカンダリ・ノードが IQ_SHARED_TEMP の 1 つまたは複数のファイルに対して書き込みアクセスができない場合、そのノードはすべての DQP 操作に参加できません。**sp_iqmpxfilestatus** を使用して、分散ファイルの問題を診断します。

共有 DB 領域への DB ファイルの追加

Interactive SQL または Sybase Central で次の手順を実行します。

前提条件

領域を IQ_SHARED_TEMP に追加する場合は、分散クエリ処理の負荷を考慮します。IQ_SHARED_TEMP に対して DB 領域コマンドを実行するには、SPACE ADMIN 権限が必要です。

手順

1. コーディネータに接続します。
2. ファイルを追加します。ADD FILE 句に、ロー・デバイスへのフル・パスを指定するか、ソフト・リンクを指定します。 次の例は、完全なパスを指定します。

プラットフォーム	構文
UNIX	`ALTER DBSPACE IQ_SHARED_TEMP ADD FILE mydbfilename '/dev/rdsk/c4t600A0B80005A7F5D0000024'`
Windows	`ALTER DBSPACE IQ_SHARED_TEMP ADD FILE mydbfilename '¥¥¥¥.¥¥PhysicalDrive2'`

   次に UNIX システム上でソフト・リンクを指定する例を示します。

```
ALTER DBSPACE IQ_SHARED_TEMP ADD FILE mydbfilename 'store/userdb1'
```

   セカンダリ・サーバの同期または再起動は必要ありません。
3. すべてのセカンダリ・ノードからファイルが参照できることを確認します。

```
sp_iqmpxfilestatus
```

4. IQ_SYSTEM_MAIN DB 領域とユーザ・メイン DB 領域の共有ファイルは、暗黙的に読み取り／書き込み操作が可能です。IQ_SHARED_TEMP の DB ファイルは読み取り専用で作成されます。DB ファイルが共有テンポラリ DB 領域にある場合、新しいファイルを読み取り／書き込みステータスに変更します。

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

次のステップ
1 つまたは複数のノードからファイル・ステータスの問題に関するエラーが返される場合は、sp_iqmpxfilestatus プロシージャを実行してトラブルシューティングし、問題を解決します。解決を延期する必要がある場合、IQ_SHARED_TEMP DB ファイルを強制的に読み取り／書き込みステータスにできます。

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

FORCE READWRITE 句を IQ_SYSTEM_MAIN DB ファイルとユーザ・メイン DB ファイルに対して使用すると、エラーが返されます。

カタログ DB 領域の追加

通常の使用状況では、カタログ DB 領域は必要に応じて自動的にサイズを増大するため、追加が必要になることはありません。ただし、使用しているカタログ DB 領域ファイルが制限済みの記憶領域上にある場合、カタログ・データの格納するために新しいカタログ DB 領域の追加が必要になる場合があります。

1. マルチプレックス内のすべてのサーバを停止します。
2. シングル・ノード・モードでコーディネータを起動します。

   ```
   start_iq @params.cfg mpxtest.db -iqmpx_sn 1
   -n mpxnode_c
   ```

3. DB 領域を作成します。

   ```
   CREATE DBSPACE DspCat2 AS 'sadb2.db'
   CATALOG STORE
   ```

4. コーディネータを **-iqmpx_sn1** スイッチを使用せずに起動します。

   ```
   start_iq @params.cfg mpxtest.db -n mpxnode_c
   ```

5. マルチプレックス内のすべてのセカンダリ・サーバを同期します。「マルチプレックス・サーバの同期」を参照してください。

参照：
- マルチプレックス・サーバの同期 (37 ページ)

カタログ DB 領域の削除

通常の使用状況では、カタログ DB 領域は必要に応じて自動的にサイズを増大するため、削除が必要になることはありません。

1. マルチプレックス内のすべてのサーバを停止します。
2. **-gm** スイッチと **-iqmpx_sn** スイッチを使用して、コーディネータを起動します。

```
start_iq @params.cfg mpxtest.db -gm 1
-iqmpx_sn 1 -n mpxnode_c
```

3. Interactive SQL で、次を入力します。

```
DROP DBSPACE DspCat2
```

4. コーディネータを **-iqmpx_sn** 1 スイッチを使用せずに通常の手順で起動します。
5. マルチプレックス内のすべてのセカンダリ・サーバを同期します。「マルチプレックス・サーバの同期」を参照してください。

参照：

- マルチプレックス・サーバの同期 (37 ページ)

動的衝突

動的衝突は、テーブル、プロシージャ、関数、またはビュー上でスキーマの変更がコミットされたときにセカンダリ・サーバが同じオブジェクトをクエリ中であると発生します。衝突は、共有 IQ オブジェクト・スキーマ更新の TLV 応答によって発生します。

動的衝突は、セカンダリ・ノード上でのみ発生します。対話型の設定では、動的衝突は、セカンダリ・ノードからのユーザ接続の強制切断の原因となります。

DDL 実行中の動的衝突を解決するために、セカンダリ・サーバによってその原因となっている接続が検出および切断されます。切断は、次のようなメッセージを使用して `.iqmsg` ファイルに記録されます。

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

動的衝突を避けるには、マルチプレックスの負荷が低いときに実行されるように、スキーマの変更をスケジュールします。

論理サーバ

論理サーバには IQ マルチプレックス用のリソース・プロビジョニングが用意されています。これにより、複数のコンピューティング・リソース (マルチプレックス・サーバ) のサブセットを 1 つの論理エンティティとしてグループ化できます。

論理サーバの概要

論理サーバを作成することにより、マルチプレックスの 1 つ以上のサーバをグループ化して、1 つ以上のアプリケーションまたは 1 人以上のユーザに示すことができます。

論理サーバに接続してクエリを実行するとき、Sybase IQ は、論理サーバのメンバであるノードのみにクエリの実行を分散します。 アプリケーションのリソース・ニーズの変化に合わせて、論理サーバとの間でメンバ・ノードを動的に追加または削除できます。

Sybase IQ のマルチプレックス・ノードへのアクセスは、論理サーバを介してのみ提供されます。 複数の論理サーバを作成することで、アプリケーションまたはユーザのさまざまなグループを提供できます。論理サーバを作成するときには、1 つまたは複数のマルチプレックス・サーバを明示的に選択し、メンバとして追加します。 マルチプレックス・ノードは、複数の論理サーバのメンバになることができます。 論理サーバは、リーダ・ノードとライタ・ノードの両方を含むことができます。 コーディネータのメンバシップには特別な規則がありますが、コーディネータ・ノードも論理サーバのメンバに含めることができます。

メンバ・サーバのみが、論理サーバに対する負荷を処理できます。負荷分散に適しているのは、並行処理が可能なクエリのみです。他のすべての操作は、初回の接続時に接続されたノードに限定されます。目的の操作に必要な役割 (リーダまたはライタ) を持つノードに接続する必要があります。

組み込みの論理サーバ

OPEN と SERVER は、常に存在する組み込みの論理サーバです。そのメンバシップは暗黙的に定義されており、カタログには格納されません。

OPEN 論理サーバ

どのユーザ定義の論理サーバにも属していないノードは、暗黙的に、OPEN 論理サーバのメンバとなります。OPEN 論理サーバは、いずれのユーザ定義の論理サーバのメンバでもないすべてのマルチプレックス・ノードから構成されます。

マルチプレックス・ノードのメンバシップを直接操作して OPEN 論理サーバに変更することはできません。マルチプレックス・ノードのメンバシップは、ユーザ定義の論理サーバにマルチプレックス・ノードを追加するか、ユーザ定義の論理サーバからマルチプレックス・ノードを削除したときに、暗黙的に変更されます。マルチプレックスの各サーバが 1 つまたは複数の論理サーバのメンバであるとき、OPEN 論理サーバが空である場合があります。

注意： ユーザ定義の論理サーバがない場合、OPEN 論理サーバは、すべてのマルチプレックス・サーバから構成されます。

OPEN 論理サーバは、次の状況をサポートします。

- マルチプレックスの大規模展開で、独自のリソースのプロビジョニングを必要とするアプリケーションが含まれる可能性があります。また、アドホックで使用され、専用リソースを必要としないアプリケーションが含まれる可能性もあります。これらのアドホック・アプリケーションは、ユーザ定義の論理サーバではなく、OPEN 論理サーバでサポートできます。
- リソースがチャージバック方式で使用されるユーティリティ・コンピューティング環境では、すべてのアプリケーションとユーザは自分に割り当てられた論理サーバを含むマルチプレックスにアクセスするので、OPEN 論理サーバを空の状態に保つことができます。

SERVER 論理サーバ

各マルチプレックス・サーバで、SERVER 論理サーバは、そのサーバのみに対する暗黙的なメンバシップを持ちます。これにより、特定の権限を持つユーザは、マルチプレックスの任意のノードに接続して、そのノードがどの論理サーバに含まれるかを知らなくてもサーバ・レベルの管理タスクを実行できます。

SERVER 論理サーバ・コンテキストを含むマルチプレックスに接続するには、次の権限のいずれかを持っている必要があります。

- DBA
- MULTIPLEX ADMIN
- SPACE ADMIN
- OPERATOR

NONE 論理サーバ

NONE 論理サーバは、常に空に定義されます。

有効な論理サーバ設定

障害またはマルチプレックスからの除外により、論理サーバのメンバ・ノードのすべてが常に利用できるわけではありません。

有効な論理サーバ設定は、実際に利用できるすべてのメンバ・ノードで構成される論理サーバの現在の動的な構成を表します。有効な論理サーバ設定は、静的論理サーバ設定とマルチプレックスの動的な状態に基づきます。

論理サーバ・メンバとしてのコーディネータ

現在のコーディネータ・サーバのマルチプレックス・サーバ名を使用して定義された論理サーバ・メンバシップは無効です。

そのようなメンバシップは、マルチプレックス・サーバが現在のコーディネータとして機能しなくなると、再び有効になります。

コーディネータの論理メンバシップ

論理サーバのコーディネータの論理メンバシップは、現在のコーディネータのマルチプレックス・サーバ名ではなく、**FOR LOGICAL COORDINATOR** 句を使用して定義できます。

コーディネータの論理メンバシップは、どのマルチプレックス・ノードがコーディネータの役割を果たしているかに関係なく、コーディネータは常に論理サーバからそのメンバとして使用できることを意味します。

注意： コーディネータの役割は、たとえばフェールオーバ時に、マルチプレックス・サーバ間で移動できます。論理サーバがコーディネータの論理メンバシップを持つ場合、新しいコーディネータ・サーバは、論理サーバの有効な設定の一部になります。

現在のコーディネータ・ノードは、そのコーディネータを論理メンバとして持つ論理サーバからのみ使用できます。マルチプレックス・ノード名を使用した現在のコーディネータ・ノードへのメンバシップを持つ論理サーバからは使用できません。現在のコーディネータは、論理サーバの論理メンバかつ名前付きメンバである場合には、その論理サーバから使用できますが、それは論理メンバシップに基づく場合のみです。名前付きメンバシップは、引き続き無効と見なされます。

コーディネータ・ノード・メンバシップの規則から、コーディネータにかかる負荷を確実に予測できます。コーディネータは既知の一連の論理サーバから利用できますが、このことは、コーディネータがマルチプレックス・ノード間でフェールオーバしても同じです。

論理サーバ・ポリシー

各論理サーバは、論理サーバ・ポリシーに関連付けられます。論理サーバ・ポリシーは、設定可能な論理サーバ・ポリシー・オプションを通じて、関連付けられた論理サーバの一部の側面と動作に影響を及ぼします。

Sybase IQ データベースは、すべての論理サーバに適用される組み込みのルート論理サーバ・ポリシーを含みます。 ルート・ポリシーで実行可能な唯一の変更は、**ALLOW_COORDINATOR_AS_MEMBER** オプションを ON (デフォルト) または OFF に設定することです。 ルート論理サーバ・ポリシーは削除できません。

注意： ユーザ定義の論理サーバ・ポリシーは作成できません。

ALLOW_COORDINATOR_AS_MEMBER オプション

コーディネータがユーザ定義の論理サーバに参加できるかどうかを制御するには、ルート論理サーバ・ポリシーの `ALLOW_COORDINATOR_AS_MEMBER` オプションを使用します。このオプションを OFF に設定すると、コーディネータは、すべてのユーザ定義の論理サーバのメンバとして使用できなくなります。デフォルト設定は ON です。

`ALLOW_COORDINATOR_AS_MEMBER` が OFF に設定されていても、コーディネータの論理メンバシップと現在のコーディネータ・サーバの名前付きメンバシップを定義できますが、これらのメンバシップは有効ではありません。

注意： 現在のコーディネータ・サーバの名前付きメンバシップは、`ALLOW_COORDINATOR_AS_MEMBER` オプションの設定に関係なく、常に無効です。この設定を変更すると、コーディネータの論理メンバシップのみが影響を受けます。たとえば、`ALLOW_COORDINATOR_AS_MEMBER` の現在の設定を OFF から ON に変更すると、現在のコーディネータは、コーディネータの論理メンバシップを持つ論理サーバから使用できるようになります。

論理サーバの管理 (Interactive SQL)

柔軟性のあるスクリプトを使用したい場合や、他の製品よりも Interactive SQL を使い慣れている場合は、Interactive SQL を使用して論理サーバを管理します。

論理サーバの作成 (Interactive SQL)

Interactive SQL を使用して、ユーザ定義の論理サーバを作成します。

1. DBA 権限または MPXADMIN 権限のあるユーザとしてデータベースに接続します。

2. **CREATE LOGICAL SERVER** 文を実行します。

   たとえば、メンバとして 3 つのマルチプレックス・ノードの n1、n2、n3 を含むユーザ定義の論理サーバ ls1 を作成するには、次のように入力します。

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

参照：

- CREATE LOGICAL SERVER 文 (117 ページ)

論理サーバの変更 (Interactive SQL)

Interactive SQL を使用して、ユーザ定義の論理サーバを変更します。

1. DBA 権限または MPXADMIN 権限のあるユーザとしてデータベースに接続します。
2. **ALTER LOGICAL SERVER** 文を実行します。

   たとえば、マルチプレックス・ノードの n4 と n5 を追加して、ユーザ定義の論理サーバ ls1 を変更するには、次のように入力します。

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

参照：

- ALTER LOGICAL SERVER 文 (107 ページ)

ルート論理サーバ・ポリシーの変更 (Interactive SQL)

Interactive SQL を使用して、ユーザ定義の論理サーバのルート論理サーバ・ポリシーを変更します。

1. DBA 権限または MPXADMIN 権限のあるユーザとしてデータベースに接続します。
2. **ALTER LS POLICY** 文を実行します。

   ```
   ALTER LS POLICY root ALLOW_COORDINATOR_AS_MEMBER=ON;
   ```

参照：

- ALTER LS POLICY 文 (113 ページ)

論理サーバへのコメントの追加 (Interactive SQL)

Interactive SQL を使用して、ユーザ定義の論理サーバにコメントを追加します。

1. DBA 権限または MPXADMIN 権限のあるユーザとしてデータベースに接続します。

2. **COMMENT ON LOGICAL SERVER** 文を実行します。

   たとえば、ユーザ定義の論理サーバ `ls1` にコメントを追加するには、次のように入力します。

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

参照：

- COMMENT ON LOGICAL SERVER 文 (116 ページ)

論理サーバの削除 (Interactive SQL)

Interactive SQL を使用して、ユーザ定義の論理サーバを削除します。

1. DBA 権限または MPXADMIN 権限のあるユーザとしてデータベースに接続します。
2. **DROP LOGICAL SERVER** 文を実行します。

   たとえば、ユーザ定義の論理サーバ `ls1` を削除するには、次のように入力します。

   ```
   DROP LOGICAL SERVER ls1
   ```

参照：

- DROP LOGICAL SERVER 文 (120 ページ)

論理サーバ削除の結果

ユーザ定義の論理サーバを削除すると、すべてのノードのメンバシップ定義が削除されます。

論理サーバへの明示的な割り当てを含む各ログイン・ポリシーは、そのログイン・ポリシーから論理サーバ割り当てを削除します。ただし、削除される論理サーバがログイン・ポリシーに割り当てられた唯一のサーバである場合、そのログイン・ポリシーの論理サーバ割り当ては NONE に設定されます。

注意： 論理サーバを削除しても、そのサーバへの既存の接続には影響はありません。論理サーバを削除する場合は、その論理サーバへのアクティブな接続がないことを確認するようおすすめします。

参照：

- 論理サーバの削除 (Sybase Central) (72 ページ)

論理サーバの管理 (Sybase Central)

マルチプレックスに詳しくないか、グラフィカル・インタフェースの使用を希望する場合は、Sybase Central を使用して論理サーバを管理します。

論理サーバの作成 (Sybase Central)

Sybase Central を使用して、ユーザ定義の論理サーバを作成します。

1. Sybase Central で、DBA 権限または MPXADMIN 権限を持つユーザとしてデータベースに接続します。
2. 左側のウィンドウ枠で、[論理サーバ] フォルダを右クリックし、[新規作成] - [論理サーバ] を選択します。
3. 論理サーバの名前を入力します。[次へ] をクリックします。
4. 論理サーバに追加するマルチプレックス・ノードを選択します。[追加] をクリックして、[次へ] をクリックします。
5. 論理サーバについて説明するコメントを入力し、[完了] をクリックします。

論理サーバの変更 (Sybase Central)

Sybase Central を使用して、ユーザ定義の論理サーバを変更します。

1. Sybase Central で、DBA 権限または MPXADMIN 権限を持つユーザとしてデータベースに接続します。
2. [論理サーバ] フォルダのリストで、論理サーバ名を右クリックし、[プロパティ] を選択します。
3. [一般] タブで、任意のコメントを入力します。
4. [サーバ] タブで、論理サーバに追加する使用可能なマルチプレックス・ノードを選択します。[追加] をクリックします。
5. (オプション) 選択したマルチプレックス・ノードを削除するには、[削除] をクリックします。
6. [適用] をクリックします。
7. [OK] をクリックします。

ルート論理サーバ・ポリシーの変更 (Sybase Central)

Sybase Central を使用して、ユーザ定義の論理サーバのルート論理サーバ・ポリシーを変更します。

1. Sybase Central で、DBA 権限または MPXADMIN 権限を持つユーザとしてデータベースに接続します。
2. 左側のウィンドウ枠で、[論理サーバ・ポリシー] フォルダをクリックします。
3. 右側のウィンドウ枠で、「ルート」論理サーバ・ポリシーを右クリックし、[プロパティ] を選択します。
4. ALLOW_COORDINATOR_AS_MEMBER の値を選択します。
5. [適用] をクリックします。
6. [OK] をクリックします。

論理サーバへのコメントの追加 (Sybase Central)

Sybase Central を使用して、ユーザ定義の論理サーバにコメントを追加します。

1. Sybase Central で、DBA 権限または MPXADMIN 権限を持つユーザとしてデータベースに接続します。
2. [論理サーバ] フォルダのリストで、論理サーバ名を右クリックし、[プロパティ] を選択します。
3. [一般] タブで、論理サーバについて説明するコメントを入力します。
4. [適用] をクリックします。
5. [OK] をクリックします。

論理サーバの削除 (Sybase Central)

Sybase Central を使用して、ユーザ定義の論理サーバを削除します。

1. Sybase Central で、DBA 権限または MPXADMIN 権限を持つユーザとしてデータベースに接続します。
2. [論理サーバ] フォルダのリストで、論理サーバ名を右クリックし、[削除] を選択します。
3. 削除するかどうかを確認する画面が表示された場合は、[はい] をクリックします。

参照：

• 論理サーバ削除の結果 (70 ページ)

論理サーバの管理 (Sybase Control Center)

論理サーバの管理に Web ベースのツールを使用したい場合は、Sybase Control Center を使用します。

Sybase Control Center では、論理サーバの追加または削除、論理サーバのプロパティまたはポリシーの変更、論理サーバ・ノードのメンバシップの設定、または論理サーバ割り当ての変更を行うことができます。

SCC の『Sybase Control Center for Sybase IQ オンライン・ヘルプ』または http://sybooks.sybase.com/nav/summary.do?prod=10680 を参照してください。

論理サーバへのアクセス

マルチプレックス・サーバは、論理サーバを使用した場合のみアクセスできます。ログイン・ポリシーは、論理サーバへのユーザ・アクセスを制御します。

1 つまたは複数の論理サーバをログイン・ポリシーに割り当てることができます。ログイン・ポリシーを使用するすべてのユーザは、割り当てられた論理サーバの有効なメンバであるマルチプレックス・サーバのみにアクセスできます。ログイン・ポリシーには、次のいずれかを割り当てることができます。

- 1 つまたは複数のユーザ定義の論理サーバか OPEN 論理サーバまたはその両方
- SERVER — すべてのマルチプレックス・サーバにアクセスできます。SERVER 論理サーバのセマンティックに従います。
- NONE — どのマルチプレックス・サーバへのアクセスも拒否されます。

注意： SERVER または NONE は、他の論理サーバと組み合わせて指定できません。

ノード・メンバシップの重複

複数の論理サーバが単一のログイン・ポリシーに割り当てられる場合、コーディネータの論理メンバシップを含めてそれらの論理サーバ間でのノード・メンバシップの重複は許可されません。

次に対して変更が行われる場合に、自動メンバシップ重複確認が強制的に実行されます。

- 論理サーバのメンバシップ — 1 つまたは複数のメンバ・ノードが論理サーバに追加されると、この論理サーバへのアクセスを許可する各ログイン・ポリシーに割り当てられている他の論理サーバのメンバシップと新しいメンバシップが

重複しないように、それらのログイン・ポリシーが検証されます。このような重複が検出されると、論理サーバのメンバシップの変更は失敗します。

- ログイン・ポリシーの論理サーバ割り当て — 1 つまたは複数のユーザ定義の論理サーバがログイン・ポリシーに追加されると、ログイン・ポリシーの新しい論理サーバ割り当てによって割り当て済みの論理サーバ間でメンバシップ重複が発生しないように、検証が実行されます。このような重複が検出されると、ログイン・ポリシーの論理サーバ割り当ての変更は失敗します。

注意： 論理サーバの定義では、同じログイン・ポリシーに割り当てられないかぎり、メンバシップの重複が許可されます。

論理サーバ割り当ての継承

ログイン・ポリシーは、独自の論理サーバ割り当てを持っていない場合に、論理サーバ割り当てをルート・ログイン・ポリシーから継承します。デフォルトでは、ルート・ログイン・ポリシーの論理サーバ割り当てには、OPEN 論理サーバのみが含まれます。これにより、デフォルト設定が使用されている場合、すべてのマルチプレックス・サーバがアクセス可能になります。

ログイン・ポリシーの有効な論理サーバ割り当ては、ログイン・ポリシーに対して明示的に行われた論理サーバ割り当てからか、またはルート・ログイン・ポリシーの論理サーバ割り当ての継承から行われます。

論理サーバ割り当てを DEFAULT に設定することにより、既存の論理サーバ割り当てをユーザ定義のログイン・ポリシーから完全に削除できます。ルート・ログイン・ポリシーで論理サーバ割り当てを DEFAULT に設定すると、OPEN 論理サーバのデフォルト割り当てが復元されます。

注意： NONE の論理サーバ割り当ては、論理サーバ割り当てがないことではありません。

接続の論理サーバ・コンテキスト

ユーザ接続を確立すると、接続の論理サーバ・コンテキストが、ユーザのログイン・ポリシーと現在のノードによって自動的に決定されます。

- ユーザのログイン・ポリシーの有効な論理サーバ割り当てが 1 つまたは複数の論理サーバである場合、接続の論理サーバ・コンテキストは、指定された論理サーバのうちの 1 つへの現在のノードの明確なメンバシップに基づきます。
- 現在のノードが、ユーザのログイン・ポリシーに割り当てられた論理サーバのどのメンバでもない場合、接続は失敗します。
- ユーザのログイン・ポリシーの有効な論理サーバ割り当てが SERVER 論理サーバであり、かつユーザに十分な権限がない場合、接続は失敗します。
- ユーザのログイン・ポリシーの論理サーバ割り当てが NONE の場合にも、接続は失敗します。

封鎖の防止

ログイン・ポリシーの論理サーバ・アクセス設定が原因となる封鎖状態を防止するために、ログイン・ポリシーでノードにアクセスできないようにされていても、DBA 接続は失敗しません。

ログイン・ポリシーの論理サーバ・アクセス設定に基づいて接続が拒否される場合、DBA 接続は、SERVER 論理サーバ・コンテキストを使用すると受け入れられます。

論理サーバを作成したときの論理サーバ割り当ての設定 (Sybase Central)

Sybase Central を使用して、論理サーバ割り当てを作成します。

1. Sybase Central で、DBA 権限または USER ADMIN 権限を持つユーザとしてデータベースに接続します。
2. 左側のウィンドウ枠で、[論理ポリシー] フォルダを右クリックし、[新規作成] - [ログイン・ポリシー] を選択します。
3. 適切なポリシー・オプションを設定し、[次へ] をクリックします。
4. 論理サーバ割り当てタイプを選択し、論理サーバをログイン・ポリシーに追加します。[追加] をクリックして、[次へ] をクリックします。
5. 適切な論理サーバ・オプションの上書き値を選択し、[次へ] をクリックします。
6. ログイン・ポリシーについて説明するコメントを入力し、[完了] をクリックします。

論理サーバを変更したときの論理サーバ割り当ての設定 (Sybase Central)

Sybase Central を使用して、論理サーバ割り当てを変更します。

1. Sybase Central で、DBA 権限または USER ADMIN 権限を持つユーザとしてデータベースに接続します。
2. 左側のウィンドウ枠で、[ログイン・ポリシー] をクリックします。
3. 右側のウィンドウ枠で、ログイン・ポリシーを右クリックし、[プロパティ] を選択します。
4. [一般] タブで、該当するポリシー・オプションを変更します。
5. [論理サーバ割り当て] タブで、論理サーバ割り当てタイプを変更し、論理サーバをログイン・ポリシーに追加するか、ログイン・ポリシーから削除します。[追加] または [削除] をクリックして、[次へ] をクリックします。

6. [論理サーバ・オプションの上書き] タブで、該当する論理サーバ・オプションの上書き値を変更します。
7. [適用] をクリックします。
8. [OK] をクリックします。

論理サーバを変更したときの論理サーバ割り当ての設定 (Interactive SQL)

Interactive SQL を使用して、論理サーバ割り当てを変更します。

1. DBA 権限または USER ADMIN 権限を持つユーザとしてデータベースに接続します。
2. **ALTER LOGIN POLICY** 文を実行します。**LOGICAL SERVER** 句は、特定のログイン・ポリシー・オプションの値を論理サーバ単位で設定します。

   たとえば、論理サーバ `ls1` をログイン・ポリシー `lp1` に割り当てることによって論理サーバ割り当てを変更するには、次のように入力します。

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

参照：

- ALTER LOGIN POLICY 文 (108 ページ)

ログイン・ポリシー・オプション

ログイン・ポリシー・オプションの **max_connections** と **dqp_enabled** の論理サーバごとのレベルの設定を指定できます。

locked ログイン・ポリシーの論理サーバは上書きできません。

注意： ログイン・ポリシー・オプションの **MULTIPLEX SERVER** 上書き句を指定して、**ALTER LOGIN POLICY** コマンドを使用することはできません。

locked オプション

ログイン・ポリシーに割り当てられた論理サーバは、**locked** オプションが OFF の場合のみアクセスできます。

locked を ON に設定すると、すべてのログイン・ポリシー・ユーザがロックアウトされます。ログイン・ポリシー・ユーザに対してアクセスを一時的に拒否するには、この設定を使用します。

max_connections オプション

max_connections 設定は、論理サーバの各メンバ・ノードで強制的に適用され、ベース・レベルまたは論理サーバごとのレベルで設定できます。

注意： あるユーザの接続時に論理サーバ・メンバ・ノードの最大接続数の制限に達すると、それ以降のそのノードでのユーザ接続は失敗します。この状況は、他のメンバ・ノードがユーザ接続数の制限に達していなくても発生します。

dqp_enabled ログイン・ポリシー・オプション

dqp_enabled ログイン・ポリシー・オプションは、そのログイン・ポリシーに割り当てられたすべてのデータベース・ユーザに対して DQP を有効または無効にします。

dqp_enabled ログイン・ポリシー・オプションに対して、論理サーバの特定の上書きを指定することもできます。このオプションで可能な設定は、ON と OFF です。

注意： デフォルトでは、**dqp_enabled** は、ルート・ログイン・ポリシーで ON に設定されます。

分散クエリ処理オプション

クエリの実行は、論理サーバの使用可能で適格なすべてのメンバ・ノードに分散できます。論理サーバに接続しているすべてのユーザが同時に負荷分散を必要とするとはかぎりません。

dqp_enabled ログイン・ポリシー・オプションかテンポラリ・データベース・オプション **dqp_enabled** またはその両方を設定することによって、一連のユーザに対して、または個々の接続レベルで、DQP オプションを有効／無効にできます。

参照：

- 分散クエリ処理 (81 ページ)

dqp_enabled テンポラリ・データベース・オプション

テンポラリ・データベース・オプション **dqp_enabled** を使用すると、接続レベルで DQP を有効または無効にできます。

テンポラリ・データベース・オプション **dqp_enabled** を OFF に設定することによって、現在の接続における DQP を無効にすることができます。このオプションを ON (デフォルト値) に設定することによって現在の接続における DQP を有効にすることができますが、それが可能なのは現在の接続の論理サーバに対するユー

ザのログイン・ポリシーによって、そのユーザに対して DQP が有効になっている場合のみです。

ユーザのログイン・ポリシーに基づいて DQP が無効にされている場合、**dqp_enabled** を ON に設定すると、次のエラーが返ります。

```
Invalid setting for option 'dqp_enabled'
```

注意： ユーザのログイン・ポリシー・オプションへのすべての変更は、新しい接続のみに影響します。既存の接続のログイン・ポリシー・オプション設定は、接続が最初に確立した時点の設定に基づきます。

論理サーバ設定

論理サーバをサポートするのは、マルチプレックスのみです。

組み込みの論理サーバと論理サーバ・ポリシーに関する情報は、シンプレックス環境のカタログに残りますが、使用できません。

論理サーバと新しいシンプレックス・データベース

新しいシンプレックス・データベースの作成時に、カタログは、組み込みの論理サーバの OPEN、SERVER、NONE に関する情報を使用して作成されます。

ルート論理サーバ・ポリシーも自動的に作成されます。OPEN 論理サーバがルート・ログイン・ポリシーに割り当てられます。

注意： 新しい論理サーバを作成するための DDL コマンドを実行すると、エラーが返されます。

マルチプレックス・データベースのシンプレックス・データベースへの変換

マルチプレックス・データベースをシンプレックス・データベースに変換する前に、ユーザ定義の論理サーバをすべて削除します。

ユーザ定義の論理サーバをすべて削除するには、**WITH DROP LOGICAL SERVER** 句を指定して **DROP MULTIPLEX SERVER** を使用します。

カタログは、次の情報を保持します。

- 組み込みの論理サーバ
- 「ルート」論理サーバ・ポリシー
- 組み込みの論理サーバに対するログイン・ポリシーの割り当て
- ログイン・ポリシー・オプションの論理サーバ・レベルの上書き。これが存在するのは、組み込みの論理サーバの場合のみです。

これらの情報は、シンプレックス・データベースをマルチプレックスに戻す場合に再使用されます。

シンプレックスでの接続

シンプレックス・データベースへの接続は、論理サーバ割り当てのログイン・ポリシー設定に影響されません。接続には、論理サーバ・コンテキストはありません。

ログイン・ポリシーの **locked** オプションは、接続がサーバに受け付けられる前に、引き続き適用されます。

分散クエリ処理

分散クエリ処理 (DQP： Distributed Query Processing) では、作業をクラスタ内の複数のノード全体に分散することによって IQ マルチプレックス設定のパフォーマンスが向上します。

分散クエリ処理を有効にするのに、設定オプションを設定する必要はありません。**dqp_enabled** ログイン・ポリシー・オプションまたは **dqp_enabled** テンポラリ・データベース・オプションをオフにして DQP を無効にしないかぎり、次の場合のクエリの審査時に DQP が自動的に行われます。

- サーバ間で MIPC 接続が確立されているマルチプレックスの一部として、サーバが構成されている。サーバはこれらの接続を起動時に自動的に確立します。確認するには、**sp_iqmpxinfo** を実行し、`mipc_public_state` カラムで *active* ステータスを探します。
  マルチプレックスにプライベート接続を設定しており、DQP がプライベート相互接続で実行することを期待している場合は、**sp_iqmpxinfo** を実行し、`mipc_private_state` カラムで *active* ステータスを探します。
- 現在の接続の論理サーバで、少なくとも他に 1 つのメンバ・ノードが利用できる。

  注意： マルチプレックス・サーバ・ノードを論理サーバに分割することにより、分散クエリ処理に参加するマルチプレックス・ノードを制御できます。

- 共有テンポラリ DB 領域で書き込み可能ファイルが利用できる。詳細については、『システム管理ガイド：第 1 巻』>「データベース・オブジェクトの管理」>「データベースの定義」>「SQL でのデータベース作成」>「データベース・ファイルの配置」および『システム管理ガイド：第 1 巻』>「データベース・オブジェクトの管理」>「Sybase IQ データベースの設定」>「領域の割り付け」を参照してください。

論理サーバに接続しているすべてのユーザが、負荷を常に分散させる必要があるわけではありません。分散クエリ処理オプションを使用して、一連のユーザに対して、または個々の接続レベルで DQP を有効または無効にします。**dqp_enabled** ログイン・ポリシー・オプションを使用して、一連のユーザに対して DQP を有効／無効にします。**dqp_enabled** テンポラリ・データベース・オプションを使用して、個々の接続レベルで DQP を有効／無効にします。

参照：

- 分散クエリ処理オプション (77 ページ)

リーダ・ノードとワーカ・ノード

分散クエリ処理において、リーダ・ノードは作業単位をワーカ・ノードに渡し、作業は、リーダ・ノードとワーカ・ノードの両方で実行しているスレッドによって実行されます。

リーダ・ノードは、クエリの発信元であるクラスタ内のどのノードでもかまいません。ワーカ・ノードは、分散クエリ処理の作業を受け入れることができるクラスタ内の任意のノードです。これらのノードを、マルチプレックスのコーディネータ・ノード、ライタ・ノード、リーダ・ノードと混同しないでください。

分散クエリ処理スレッドの使用状況は、**sp_iqcontext** システム・ストアド・プロシージャを使用して詳細に表示できます。

クエリが発行されると作業単位が分散されますが、現在の接続の論理サーバのメンバであるノードのみに分散されます。現在の接続の論理サーバのメンバではないマルチプレックス・ノードは、そのクエリの分散クエリ処理に参加しません。リーダ・ノードは、リーダ・ノードと同じ論理サーバ内から、分散クエリを処理するワーカ・ノードを自動的に選択します。論理サーバからマルチプレックス・ノードを除外すると、その論理サーバに関してそれらのノードで分散クエリ処理は行われません。

クエリ部分

クエリは、個別の部分で分散されます。

次のクエリ部分は、分散されることはありません。

- 読み取り／書き込みクエリ部分 (DDL、INSERT、LOAD、UPDATE、DELETE など)
- テンポラリ・テーブルを参照するクエリ部分
- システム DB 領域のテーブルを参照するクエリ部分
- プロキシ・テーブルを参照するクエリ部分
- NEWID など非決定的関数を参照するクエリ部分

分散クエリのパフォーマンス

一般的には、利用可能なノードとリソースが多いほど、クエリのパフォーマンスが高くなる可能性があります。

分散クエリ処理では、論理サーバのすべてのノードの利用可能なメモリと CPU のリソースが使用されます。

分散クエリ処理で得られるメリットの程度は、クエリのタイプ、クエリのサイズ、論理サーバ内のノードの現在の負荷によって異なります。

注意： サーバ名、ホスト名、ポートなどのマルチプレックス・サーバのプロパティを変更した場合、マルチプレックス・サーバを再起動した後に、DQP クエリに参加できるまでに少なくとも 2 分間待機する必要があります。サーバが再起動してから 2 分以内に DQP クエリが実行されると、そのサーバは作業に参加しない可能性があります。

同じクエリを 2 回実行すると、異なる作業分散となることがあります。クラスタ内の負荷レベルが変化するため、異なる負荷分散が行われます。分散クエリのパフォーマンスは、任意のタイミングでの論理サーバの全体的な負荷によって決定されます。同様に、処理時間の長いクエリを 1 回実行すると、ワーカ・ノード間で負荷バランスが変化するため、クエリの実行中に作業分散が変化します。

注意： **-iqmc** スイッチと **-iqtc** スイッチを使用することにより、マルチプレックス内の各ノードで異なるキャッシュ・サイズを指定できますが、これはパフォーマンスに悪影響を与えることがあります。たとえば、1 つのワーカ・ノードでリーダ・ノードよりもかなり小さなキャッシュが設定されていると、リーダ・ノードのハッシュ・ジョインが、並行処理を禁止するページング・モードで実行されることがあります。

分散クエリのパフォーマンスを最大化するために高速なプライベート相互接続が推奨されますが、必須ではありません。『インストールおよび設定ガイド』の「インストール前の準備」>「インストールの計画」>「分散クエリ処理または高可用性のための計画」 を参照してください。

注意： DQP のパフォーマンスを確認するのに、**NOEXEC** オプションは使用しないでください。**NOEXEC** は、DQP のトラブルシューティングに役立ちません。

Sybase Central のパフォーマンスと統計のモニタ

トポロジ・ビューは、マルチプレックス環境全体をグラフィカルに表現し、既存のリスト・ビューを補完します。これにより、マルチプレックス環境の状態を視覚的にモニタリングすることが簡単になります。

トポロジ・ビュー

Sybase Central 用の Sybase IQ プラグインにあるマルチプレックス・コンテナには、グラフィカルな詳細ウィンドウ枠が用意されています。

ビューは、ノードとリンクで構成されています。ノードはマルチプレックス・サーバを、リンクは 2 つのマルチプレックス・サーバ間の接続をそれぞれ表します。リンクは、コーディネータ・ノード (1 つだけ存在する) とセカンダリ・ノードの間にのみ存在します。リンクが 2 つのセカンダリ・ノードの間に存在することはありません。

ビュー内でノードを移動して、レイアウトを並べ替えることができます。次にトポロジ・ビューを表示すると、最後に使ったグラフィカル・レイアウトに最も近い形でレイアウトがリストアされます (マルチプレックス設定が変更された場合、レイアウトは部分的にのみリストアできます)。

トポロジ・ビューを使用して、フェールオーバ・ノードを指定し、フェールオーバを実行できます。「指定済みのフェールオーバ・ノード」と「フェールオーバの要件」を参照してください。

参照：

- マルチプレックス・トポロジの表示 (85 ページ)
- マルチプレックス・トポロジの印刷 (86 ページ)

マルチプレックス・トポロジの表示

Sybase Central のトポロジ・ビューには、マルチプレックス内にあるサーバの種類と現在のステータスが表示されます。
Sybase Central のツリー・ビューで、マルチプレックスの名前を選択し、[トポロジ] タブをクリックします。

トポロジ・ビューは印刷および保存もできます。

参照：

- トポロジ・ビュー (85 ページ)
- 指定済みのフェールオーバ・ノード (44 ページ)
- フェールオーバの要件 (45 ページ)

マルチプレックス・トポロジの印刷

Sybase Central のトポロジ・ビューでパフォーマンス・グラフを印刷できます。

1. Sybase Central のツリー・ビューでマルチプレックス名を選択し、[トポロジ] タブに切り替えます。
2. トポロジ・ビューで右クリックして、[グラフの印刷] を選択します。
3. 印刷オプションを確認し、印刷するプリンタを指定してページを印刷します。

参照：

- トポロジ・ビュー (85 ページ)
- 指定済みのフェールオーバ・ノード (44 ページ)
- フェールオーバの要件 (45 ページ)

グラフの保存

Sybase Central でパフォーマンス・モニタ・グラフを保存できます。

1. トポロジ・ビューで、右クリックして、[名前を付けてグラフを保存] を選択します。
2. JPEG ファイルの名前を入力します。
3. [保存] をクリックします。

マルチプレックス・ステータス・モニタの更新頻度の変更

トポロジ・ウィンドウ枠の一番上には、最終更新日時のタイムスタンプ、サーバ・ステータスの概要、ステータス変更イベント・メッセージ (ステータス変更イベントが発生した場合) が記録されたステータス・メッセージが表示されます。

1. Sybase Central は、マルチプレックスのステータスをモニタし、サーバが削除または除外された場合にトポロジ・ビューとステータス・メッセージを自動的に再表示します。更新間隔を指定することもできます。

2. トポロジ・ビューで右クリックし、[モニタ・レートの変更] を選択します。
3. ステータス・チェック間の秒数を指定します。
4. [OK] をクリックします。

ノード情報の表示

トポロジ・ビューでは、グラフィカル表示に加え、サーバに関するテキストによる情報も表示されます。

1. 各アイコンの横のラベルにはサーバ名とモードが表示されます。
2. サーバのアイコンにマウスを合わせると、ホスト、ポート、状態、役割、およびステータスがヒントに表示されます。

ノード間通信の状態

トポロジ・ビューでは、各セカンダリ・ノードからコーディネータへの通信リンクがさまざまな線で表示されます。

このリンクのラベル、スタイル、色によって、ノード間通信 (INC) の状態がわかります。

- アクティブ – 緑色の実線
- 応答なし – オレンジ色の破線
- タイムアウト – 赤色の点線
- 除外済み – 「除外済み」のラベル付きのリンク

各リンクを右クリックすると、ポップアップ・メニューが表示され、ハートビートと接続プールの状態および統計を示すプロパティ・シートが表示されます。「ハートビート接続」と「プール済み接続」を参照してください。

参照：

- ハートビート接続 (9 ページ)
- プール済み接続 (9 ページ)

サーバ・モード

ノードのラベルには、サーバ名とモードが表示されます。

- コーディネータ — 作業を割り当て、マルチプレックスの設定データを管理するリーダ／ライタ・サーバ。マルチプレックスごとに 1 つのコーディネータのみが存在します。
- リーダ — 読み取り専用のセカンダリ・サーバ。

- ライタ — 読み取り／書き込み用のセカンダリ・サーバ。

サーバの状態

サーバ・アイコンには、サーバの状態が次のように示されます。

- 実行中 – 現在接続されている場合、緑色。接続されていない場合、右上隅にアダプタ (電源プラグ) のある灰色の表示。
- 停止中 – 下向き矢印のある赤色の表示。

サーバ・ステータス

サーバ・アイコンには、サーバのステータスが次のように示されます。

- 包含済み。
- 除外済み – 「除外済み」ラベル。

パフォーマンス・モニタへのアクセス

[パフォーマンス・モニタ] には、1 つまたは複数の参加ノードの統計のコレクションが表示されます。統計は、リアル・タイムな動的チャートで表示されます。

パフォーマンス・モニタには、次の 2 つのレベルでアクセスできます。

- マルチプレックス・レベル – 複数のサーバについて、1 つの統計のみをモニタできます。
- サーバ・レベル – 単一のサーバまたは単一のマルチプレックス・サーバについて、同時に最大 10 個の統計をモニタできます。

Sybase Central のツリー・ビューで、サーバまたはマルチプレックスを選択し、[パフォーマンス・モニタ] タブをクリックします。パフォーマンス・モニタ・グラフの種類や内容は変更できます。

データ収集レートの設定

[収集レートの設定] ダイアログは、サーバ・レベルのパフォーマンスのモニタリングでのみ使用できます。このダイアログでは、統計を収集する頻度を変更できます。

パフォーマンス・モニタが表示されると、マルチスレッド・データ収集エンジンは、モニタ対象の統計を異なる 2 つの頻度で収集します。コストの低い統計の収集頻度は、コストの高い統計の収集頻度より多くなります。このデフォルトの収集レートを変更するには、各自でレートを設定します。

1. Sybase Central のツリー・ビューで、サーバを選択してから [パフォーマンス・モニタ] タブをクリックします。
2. チャート領域で右クリックして、[収集レートの設定] を選択します。
3. 統計収集の頻度を秒単位で選択します。
4. サーバ・パフォーマンスへの影響を低減するには、このレート値を大きくして、統計収集が頻繁に行われないようにします。
5. [OK] をクリックします。

[収集レートの設定] ダイアログ

[収集レートの設定] ダイアログの動作は、サーバ・レベルのモニタリングとマルチプレックス・レベルのモニタリングで異なります。

- このダイアログをサーバ・レベルのパフォーマンス・モニタから起動すると (サーバは単一サーバでもマルチプレックス・サーバでも可)、同時に最大 10 の統計をモニタリングできます。
- このダイアログをマルチプレックス・レベルのパフォーマンス・モニタから起動すると、同時にモニタリングできる統計は 1 つだけになります。

1 つの統計のモニタリング

一度に 1 つの統計のみをモニタリングする必要がある場合、マルチプレックス・レベルのパフォーマンス・モニタから [収集レートの設定] ダイアログを起動します。

1. マルチプレックス・サーバをクリックし、[パフォーマンス・モニタ] タブをクリックします。
2. チャート領域で右クリックして、[統計の変更] を選択します。
3. モニタリングする統計を選択します。表示可能なすべての統計情報は、[パフォーマンス・モニタへのアクセス] に示されています。
4. 変更内容を保存するには、[OK] をクリックします。

モニタリング対象のサーバの変更

[モニタするサーバの変更] ダイアログは、マルチプレックス・レベルのパフォーマンスのモニタリングでのみ使用できます。

1. マルチプレックスをクリックし、[パフォーマンス・モニタ] タブに切り替えます。
2. パフォーマンス・モニタのチャートで右クリックし、[サーバを変更] を選択します。

3. モニタするサーバをそれぞれ選択します。モニタの対象外とするサーバは、選択を解除します。
4. 変更内容を保存するには、[OK] をクリックします。

チャートの保存

Sybase Central では、パフォーマンス・モニタのチャートは JPEG イメージ・ファイルとして保存できます。

1. パフォーマンス・モニタのチャートで右クリックして、[名前を付けてチャートを保存] を選択します。
2. JPEG ファイルの名前を入力します。
3. [保存] をクリックします。

チャートの印刷

パフォーマンス・モニタのチャートを印刷できます。

1. パフォーマンス・モニタのチャートで右クリックして、[チャートの印刷] を選択します。
2. 印刷オプションを確認し、印刷するプリンタにページを送信します。

チャート・ビューの切り替え

チャート・ビューは、時系列、2 次元棒グラフ、3 次元棒グラフから選択できます。

1. パフォーマンス・モニタのチャートで右クリックして、[Chart View] を選択します。
2. [時系列グラフ]、[棒グラフ 2 D 縦]、または [棒グラフ 3 D 縦] を選択します。

チャートのカスタマイズ

チャートをカスタマイズして、チャートの設定とチャートの更新レート (モニタ GUI のハートビート・レート) を変更します。

チャートの設定とチャートの更新レート (モニタ GUI のハートビート・レート) を変更するには、次の手順を実行します。

1. パフォーマンス・モニタのチャート領域で右クリックして、[チャートのカスタマイズ] を選択します。

   次のコンポーネントを変更します。

- [期間] ー チャートが時系列チャートである場合にのみ表示されます。データを追跡し、チャートの統計を生成する期間を指定します。最小値は 1 分です。最大値は 240 分 (4 時間) です。
- [チャート更新レート] – チャートのデータを更新する頻度を秒単位で指定します。チャート更新レートは、[パフォーマンス・モニタ] パネルの下部にも表示されます。
- [実際の値と正規化された値] ー 実際の値 (デフォルト) は実際のデータを反映します。正規化された値は、チャート・データを固定範囲にスケールします。このオプションは表示のみを目的としたもので、異なる範囲の統計を 1 つのチャートでより適切に表示できます。
- [凡例] ー 凡例の表示または非表示を選択できます。

2. 変更内容を保存するには、[OK] をクリックします。

統計のカテゴリー

統計は、次のカテゴリーにグループ化されています。

- CPU 使用率統計
- メモリ使用状況統計
- キャッシュ統計
- スレッド統計
- 接続統計
- 要求統計
- トランザクション統計
- ストア I/O 統計
- DB 領域使用状況
- ネットワーク統計

CPU 使用率統計

CPU 使用率統計には、システムによる使用率、ユーザによる使用率、CPU の全使用率が表示されます。

表 7 : CPU 使用率

名前	説明	デフォルトでのモニタリング
CPU 使用率	IQ プロセスの CPU 使用率のパーセンテージ (この値には、システムによる使用とユーザによる使用の両方が含まれる)。	可

名前	説明	デフォルトでのモニタリング
CPU システム使用率	IQ プロセスの CPU システム使用率のパーセンテージ。	不可
CPU ユーザ使用率	IQ プロセスの CPU ユーザ使用率のパーセンテージ。	不可

メモリ使用状況統計

メモリ使用状況統計には、現在の割り付け量と最大割り付け量が、いずれも MB 単位で示されます。

表 8 : メモリ使用量

名前	説明	デフォルトでのモニタリング
割り付けられたメモリ	IQ サーバによって割り付けられているメモリ容量 (MB 単位)。	可
割り付けられる最大メモリ	IQ サーバによって割り付けられる最大のメモリ容量 (MB 単位)。	不可

キャッシュ統計

キャッシュ統計には、メイン、テンポラリ、カタログの 1 秒あたりのヒット数とリード数、使用中のキャッシュ、固定されたキャッシュ、ダーティなキャッシュのそれぞれの割合 (パーセンテージ)、固定されたページの数が表示されます。

表 9 : キャッシュ統計

名前	説明	デフォルトでのモニタリング
カタログ・キャッシュ・ヒット数	1 秒あたりのカタログ・キャッシュ・ヒット数。	不可
テンポラリ・キャッシュ・ヒット数	1 秒あたりのテンポラリ・キャッシュ・ヒット数。	不可
メイン・キャッシュ・ヒット数	1 秒あたりのメイン・キャッシュ・ヒット数。	不可

名前	説明	デフォルトでのモニタリング
カタログ・キャッシュ・リード数	1 秒あたりのカタログ・キャッシュ・リード数。	可
テンポラリ・キャッシュ・リード数	1 秒あたりのテンポラリ・キャッシュ・リード数。	不可
メイン・キャッシュ・リード数	1 秒あたりのメイン・キャッシュ・リード数。	不可
現在のカタログ・キャッシュ・サイズ	現在のカタログ・キャッシュ・サイズ (MB 単位)。	不可
現在のテンポラリ・キャッシュ・サイズ	現在のテンポラリ・キャッシュ・サイズ (MB 単位)。	不可
現在のメイン・キャッシュ・サイズ	現在のメイン・キャッシュ・サイズ (MB 単位)。	不可
カタログ・キャッシュ使用率 (パーセンテージ)	使用中のカタログ・キャッシュの割合 (パーセンテージ)。	不可
テンポラリ・キャッシュ使用率 (パーセンテージ)	使用中のテンポラリ・キャッシュの割合 (パーセンテージ)。	不可
メイン・キャッシュ使用率 (パーセンテージ)	使用中のメイン・キャッシュの割合 (パーセンテージ)。	不可
固定されたカタログ・キャッシュ	固定されたカタログ・キャッシュ・ページの数。	不可
固定されたテンポラリ・キャッシュ	固定されたテンポラリ・キャッシュ・ページの数。	不可
固定されたメイン・キャッシュ	固定されたメイン・キャッシュ・ページの数。	不可
カタログ・キャッシュの固定率 (パーセンテージ)	固定されたカタログ・キャッシュの比率 (パーセンテージ)。	不可
テンポラリ・キャッシュの固定率 (パーセンテージ)	固定されたテンポラリ・キャッシュの比率 (パーセンテージ)。	不可
メイン・キャッシュの固定率 (パーセンテージ)	固定されたメイン・キャッシュの比率 (パーセンテージ)。	不可

名前	説明	デフォルトでのモニタリング
ダーティなカタログ・キャッシュ・ページの割合 (パーセンテージ)	ダーティなカタログ・キャッシュ・ページの比率 (パーセンテージ)。	不可
ダーティなテンポラリ・キャッシュ・ページの割合 (パーセンテージ)	ダーティなテンポラリ・キャッシュ・ページの比率 (パーセンテージ)。	不可
ダーティなメイン・キャッシュ・ページの割合 (パーセンテージ)	ダーティなメイン・キャッシュ・ページの比率 (パーセンテージ)。	不可

スレッド統計

スレッド統計には、使用中の SA スレッド数と IQ スレッド数、使用可能な IQ スレッド数が表示されます。

表 10 : スレッド統計

名前	説明	デフォルトでのモニタリング
使用中の IQ スレッド数	IQ サーバによって使用されているスレッド数。	不可
使用可能な IQ スレッド数	IQ サーバが使用できるスレッドの数。	不可
使用中の SA スレッド数	SQL Anywhere エンジンによって使用されているスレッド数。	不可

接続統計

接続統計には、ユーザ接続の数、1 分あたりのユーザ接続とユーザ切断の数、INC 受信接続と INC 送信接続の数が表示されます。

表 11 : 接続統計

名前	説明	デフォルトでのモニタリング
接続数合計	ユーザ接続および INC 接続を含めた接続総数。	可

名前	説明	デフォルトでのモニタリング
ユーザ接続数	ユーザ接続の数。	不可
INC 受信接続数	INC 受信接続の数。	不可
INC 送信接続数	INC 送信接続の数。	不可
1 分あたりのユーザ接続数	1 分あたりのユーザ接続の数。	不可
1 分あたりのユーザ切断数	1 分あたりのユーザ切断の数。	不可

要求統計

要求統計には、要求の数、スケジュールされていない要求の数、待機オペレーションの数、アクティブ・オペレーションの数が表示されます。

表 12 : 要求統計

名前	説明	デフォルトでのモニタリング
要求	新しい要求処理または既存の要求の処理の継続のためにサーバに入った 1 秒あたりの回数。	不可
スケジュールされていない要求	現在、使用可能なサーバ・スレッドをキュー内で待機している要求の数。	不可
IQ 待機オペレーション	リソース・ガバナを待機している IQ オペレーションの数。	不可
IQ アクティブ・オペレーション	アクティブな IQ オペレーションの数。	不可

トランザクション統計

トランザクション統計には、アクティブなユーザ・トランザクションとアクティブな INC トランン ザクションの数と、アクティブな LOAD TABLE 文の数が表示されます。

表 13 : トランザクション統計

名前	説明	デフォルトでのモニタリング
トランザクション数合計	ユーザ・トランザクションおよび INC トランザクションを含めたアクティブなトランザクションの総数。	不可
ユーザ・トランザクション数	アクティブなユーザ・トランザクションの数。	不可
INC トランザクション数	アクティブな INC トランザクションの数。	不可
アクティブな LOAD TABLE 文	アクティブな LOAD TABLE 文の数。	不可

ストア I/O 統計

ストア I/O 統計には、カタログ・ストア、メイン・ストア、テンポラリ・ストアとの間で読み書きされた、1 秒あたりのキロバイト数が表示されます。

表 14 : ストア I/O 統計

名前	説明	デフォルトでのモニタリング
カタログ・ストア・ディスク・リード数	カタログ・ストアから読み取られた 1 秒あたりのキロバイト数。	不可
テンポラリ・ストア・ディスク・リード数	テンポラリ・ストアから読み取られた 1 秒あたりのキロバイト数。	不可
メイン・ストア・ディスク・リード数	メイン・ストアから読み取られた 1 秒あたりのキロバイト数。	不可
カタログ・ストア・ディスク・ライト数	カタログ・ストアに書き込まれた 1 秒あたりのキロバイト数。	不可

名前	説明	デフォルトでのモニタリング
テンポラリ・ストア・ディスク・ライト数	テンポラリ・ストアに書き込まれた 1 秒あたりのキロバイト数。	不可
メイン・ストア・ディスク・ライト数	メイン・ストアに書き込まれた 1 秒あたりのキロバイト数。	不可

DB 領域使用状況

DB 領域統計には、ファイルごとの使用可能な領域と使用中の領域の量、DB 領域ごとの使用可能な領域と使用中の領域の量が表示されます。

表 15 : DB 領域使用状況

名前	説明	デフォルトでのモニタリング
DB 領域ごとの空き容量 (パーセンテージ)	各 DB 領域で使用可能な空き領域をパーセンテージで示します。DB 領域ごとにこの統計があります。	不可
使用中の各 DB 領域サイズ	使用中の DB 領域のサイズ。DB 領域ごとにこの統計があります。	不可
各 DB 領域のファイルごとの空き容量 (パーセンテージ)	各 DB 領域ファイルで使用可能な空き領域をパーセンテージで示します。DB 領域のファイルごとにこの統計があります。	不可
使用中の各 DB 領域ファイル・サイズ	使用中の DB 領域ファイル・サイズ。DB 領域のファイルごとにこの統計があります。	不可

<u>ネットワーク統計</u>
ネットワーク統計には、クライアント／サーバ間の通信で送受信された 1 秒あたりのバイト数、使用可能な通信バッファと全通信バッファの数が表示されます。

表 16 : ネットワーク統計

名前	説明	デフォルトでのモニタリング
受信バイト数	クライアント／サーバ間の通信で受信された 1 秒あたりのバイト数。	可
圧縮されていない状態で受信されたバイト数	クライアント／サーバ間の通信で受信された 1 秒あたりのバイト数 (圧縮が無効の場合)。	不可
送信されたバイト数	クライアント／サーバ間の通信で送信された 1 秒あたりのバイト数。	可
圧縮されていない状態で送信されたバイト数	クライアント／サーバ間の通信で送信された 1 秒あたりのバイト数 (圧縮が無効の場合)。	不可
未使用の通信バッファ数	使用可能なネットワーク通信バッファ数。	不可
通信バッファ数合計	ネットワーク通信バッファ数の合計。	不可

マルチプレックスでのバックアップ操作とリストア操作

BACKUP SQL コマンドと **RESTORE** SQL コマンドをコーディネータ・ノードのみで実行します。

データベースの一貫性のチェック

バックアップ前にデータベースの一貫性をチェックします。

sp_iqcheckdb プロシージャは、指定されたパラメータに応じて、さまざまな機能を実行できます。

データベースの一貫性の検査と修復の 4 つのモードについては、『リファレンス：ビルディング・ブロック、テーブル、およびプロシージャ』の「システム・プロシージャ」>「sp_iqcheckdb プロシージャ」を参照してください。

sp_iqcheckdb は、任意のマルチプレックス・ノードで実行できますが、セカンダリ・サーバではプロシージャがフリー・リストにアクセスできないため、フリー・リストのチェックは行われません。

sp_iqcheckdb dropleaks を実行するには、マルチプレックス・コーディネータをシングルノード・モード (**-iqmpx_sn 1**) で起動する必要があります。マルチプレックスのセカンダリ・ノードでは dropleaks モードを実行できません。

バックアップ要件

BACKUP SQL コマンドをコーディネータ・ノードのみで実行します。

- 推奨方法に従ってロー・デバイスの名前にシンボリック・リンクを使用している場合は、システム・バックアップ・ユーティリティがシンボリック・リンクに従ってデバイスをバックアップしていることを確認してください。
- 仮想バックアップを使用している場合、バックアップで指定されているすべてのメイン・ストア DB ファイルを、使用するシステム・バックアップの仕様に追加します。システム・バックアップ・リストの作成には、ストアド・プロシージャの **sp_iqfile** を使用します。
- バックアップを追跡し、リストア・アクションを計画するには、ストアド・プロシージャの **sp_iqbackupsummary**、**sp_iqbackupdetails**、**sp_iqrestoreaction**、システム・ビューの SYSIQBACKUPHISTORY と

SYSIQBACKUPHISTORYDETAIL、**db_backupheader** ユーティリティを使用します。

IQ ストアとカタログ・ストアのバックアップ

次のガイドラインに従って、マルチプレックス・データベースをバックアップしてください。

1. マルチプレックス・データベース上で IQ ストアおよびカタログ・ストアをバックアップするには、DBA 権限またはバックアップ権限があるアカウントを使用してコーディネータにログインします。
2. SQL BACKUP コマンドを発行します。完全な構文については、『リファレンス：文とオプション』の「SQL 文」>「BACKUP 文」を参照してください。
3. 『システム管理ガイド：第 1 巻』の「データのバックアップ、リカバリ、アーカイブ」>「バックアップの種類」の説明に従って、IQ ストアをバックアップします。

リストア前の作業

リストアを行う前に、リストア操作が必要かどうかを Sybase 製品の保守契約を結んでいるサポート・センタに確認します。

セカンダリ・サーバ上のデータベースが開けない場合は、「マルチプレックス・サーバの同期」の手順に従ってサーバを同期します。セカンダリ・ノードに問題がある場合に、コーディネータ・ノードをリストアする必要はありません。

読み取り／書き込みリストア操作は、ユーティリティ・データベースを実行中のサーバに対してのみ実行できます。リストア操作をセカンダリ・サーバに対して実行することはできません。

完全な構文については、『リファレンス：文とオプション』の「SQL 文」>「RESTORE 文」を参照してください。

リストア操作中は、読み取り専用ファイルのバックアップをリストアする場合のみデータベースを実行できます。読み取り専用の DB 領域内にファイルをリストアする場合には、その DB 領域がオフラインになっている必要があります。

何も問題がない場合でも、リストア操作を実行することで異なるシステム上にマルチプレックスを再作成できます。

参照：

- 同じ場所にデータベースをリストアする場合の IQ ストア・データのリストア (103 ページ)

• 異なる場所にデータベースをリストアする場合の IQ ストア・データのリストア (101 ページ)

異なる場所にデータベースをリストアする場合の IQ ストア・データのリストア

リストア操作は、データをリストアする場所によって異なります。

1. 各サーバのデータベース・ホーム・ディレクトリが残っているかどうかを確認します。残っていない場合は、ディレクトリを作成するか、ファイル・システムのバックアップからリストアします。
2. 新しい場所へのリストアが初めてでない場合には、リストア先の場所で実行されているすべてのマルチプレックス・サーバ (コーディネータ・サーバおよびセカンダリ・サーバ) を停止します。Sybase Central でマルチプレックスを右クリックし、サーバ停止ウィザードを使用します。このウィザードを使用すると、1 つ以上のサーバを停止できます。バックアップを行った元の場所のマルチプレックスは引き続き実行できます。

注意： ODBC 設定で自動起動が有効になっている場合、サーバと同じマシン上のユーザが、サーバを自動的に起動するように設定している可能性があります。データベースのリストア中に、サーバが自動的に起動しないようにしてください。

3. データベースが正常に停止されていることを確認します。

プラットフォーム	対処法
UNIX	`% ps -ef | grep iqsrv15` マルチプレックスの名前が付いたアクティブな **iqsrv15** プロセスがある場合は、そのプロセスを停止します。
Windows	タスク マネージャーの [プロセス] タブで `iqsrv15.exe` を探すか、システム・トレイの IQ サーバ・アイコンを探して、右クリックで表示されるメニューから [プロセスの終了] を選択します。

4. マルチプレックスのデバッグと再設定に必要なファイルを移動します。
   - `.iqmsg` ファイルのファイル・システム・コピーを作成します。メッセージ・ログのアーカイブが設定されている場合は、『システム管理ガイド：第 1 巻』の「データのバックアップ、リカバリ、アーカイブ」>「システ

ム・レベルのバックアップ」>「適切なファイルのバックアップ」を参照してください。

5. 元のコーディネータを停止し、そのコーディネータ・サーバの名前を使用して、コーディネータ・サーバのディレクトリからユーティリティ・データベースを起動します。

```
% start_iq -n coordinator_svr -c 32MB
-x tcpip(port=1234)
```

6. ユーティリティ・データベース (`utility_db`) に接続します。

```
% dbisql -c "eng=coordinator_svr;uid=DBA;pwd=SQL;
dbn=utility_db" -host myhost -port 1234
```

7. データベースの新しいロケーション・パスとすべての DB 領域を使用して、**RESTORE** コマンドを実行します。特定の DB 領域ファイルを別のパスにリストアする場合は、`RENAME` 句を指定します。『リファレンス：文とオプション』の「SQL 文」>「RESTORE 文」を参照してください。

8. リストアされたデータベースの起動は、`utility_db` サーバに再接続し、リストアされたデータベース・ファイル名を指定するか、サーバを停止し、リストアされたデータベースを使用してサーバを再起動することによって行います。サーバを再起動する場合は、シングル・ノードと上書きフラグ (**-iqmpx_sn 1 -iqmpx_ov 1**) を使用します。

9. **DROP MULTIPLEX SERVER** 文を使用して、すべてのセカンダリ・ノードを削除します。たとえば、`DROP MULTIPLEX SERVER node_w3_skm` のように記述します。

   最後のセカンダリ・ノードを削除すると、コーディネータは自動的に停止します。これは、シンプレックスへの変換を意味します。

10. コーディネータをシングル・ノードまたは上書きスイッチなしで再起動します。

11. データベース・ファイルの拡張子 (.DB) を含めて適切なパスを使用して、すべてのセカンダリ・ノードを再作成します。

```
CREATE MULTIPLEX SERVER node_r2_skm DATABASE
'/sunx3005/mpx_simdb.db'
HOST 'localhost' PORT 8998
ROLE READER STATUS INCLUDED
```

   最初のセカンダリ・ノードを作成すると、サーバは自動的に停止します。これは、マルチプレックスへの変換を意味します。

12. コーディネータを再起動すると、マルチプレックス環境が無効であることを示す警告がサーバ・ログに表示されます。この警告は、IQ_SYSTEM_TEMP DB 領域にファイルがまったくない場合に表示されます。これは、手順 12 で作成したすべてのセカンダリ・ノードに該当します。現時点ではこの警告を無視します。

13.「マルチプレックス・サーバの同期」の手順に従って、セカンダリ・サーバを同期します。
14. セカンダリ・サーバを起動します。
15. 各セカンダリ・サーバに接続し、IQ_SYSTEM_TEMP にファイルを追加します。
16. コーディネータ上で **sp_iqmpxvalidate** を実行します。`no error detected` が返されるはずです。

マルチプレックスのまったく同じコピーを別の場所にリストアするときに、サーバのすべてのテンポラリ・ファイルのコピーがその新しい場所にある場合は、手順 9 ～ 16 の代わりに次の手順を実行します。

ALTER MULTIPLEX SERVER を使用して、各サーバのサーバ名、ホスト、ポート、データベース・パスを変更します。

参照：

- リストア前の作業 (100 ページ)
- マルチプレックス・サーバの同期 (37 ページ)

同じ場所にデータベースをリストアする場合の IQ ストア・データのリストア

リストア操作は、データをリストアする場所によって異なります。

1. 各サーバのデータベース・ホーム・ディレクトリが残っているかどうかを確認します。残っていない場合は、ディレクトリを作成するか、ファイル・システムのバックアップからリストアします。
2. マルチプレックス内のすべてのサーバ (コーディネータ・サーバおよびセカンダリ・サーバ) を停止します。Sybase Central でマルチプレックスのアイコンを右クリックし、サーバ停止ウィザードを使用します。ウィザードでは停止するサーバを選択できます。

   注意： ODBC 設定で自動起動が有効になっている場合、サーバと同じマシン上のユーザが、サーバを自動的に起動するように設定している可能性があります。データベースのリストア中に、サーバが自動的に起動しないようにしてください。

3. データベースが正常に停止されていることを確認します。

プラットフォーム	対処法
UNIX	`% ps -ef | grep iqsrv15` マルチプレックスの名前が付いたアクティブな **iqsrv15** プロセスがある場合は、そのプロセスを停止します。
Windows	タスク マネージャーの [プロセス] タブで `iqsrv15.exe` を探すか、システム・トレイの IQ サーバ・アイコンを探して、右クリックで表示されるメニューから [プロセスの終了] を選択します。

4. マルチプレックスのデバッグと再設定に必要なファイルを移動します。
   - `.iqmsg` ファイルのファイル・システム・コピーを作成します。メッセージ・ログのアーカイブが設定されている場合は、『システム管理ガイド：第 1 巻』の「データのバックアップ、リカバリ、アーカイブ」>「システム・レベルのバックアップ」>「適切なファイルのバックアップ」を参照してください。
   - 各サーバ上で、そのサーバ用に IQ_SYSTEM_TEMP に追加されたすべてのファイルを保存します。これらのファイルは、オペレーティング・システム・ファイルを使用した場合は `dbname.iqtmp` の形式になります。また、ロー・デバイスの場合もあります。IQ テンポラリ・ストアが破損している場合は、テンポラリ・ストア DB 領域を削除および再作成できるよう、**-iqnotemp** スイッチを使用してサーバを起動します。詳細については、『リリース・ノート』を参照してください。
     コーディネータからデータベースまたは次のファイルを削除します。

     ```
     <database_home>/<dbname>.db
     ```

     ```
     <database_home>/<dbname>.log
     ```

     クエリ・サーバが破損している場合は、**RESTORE** の実行後にクエリ・サーバを削除および再作成します。その後、「異なる場所にデータベースをリストアする場合の IQ ストア・データのリストア」にある手順に従います。
5. コーディネータ・サーバのディレクトリからユーティリティ・データベースを起動します。登録されているセカンダリ・サーバの名前以外の任意の有効な識別子を、サーバ名として使用します。コーディネータの名前を使用する場合、リストア後にコーディネータの名前を変更します。

   ```
   % start_iq -n utility_startup_svr -c 32MB
   -x tcpip(port=1234)
   ```
6. ユーティリティ・データベース (`utility_db`) に接続します。

   ```
   % dbisql -c "eng=utility_startup_svr;uid=DBA;pwd=SQL;
   dbn=utility_db"
   ```

7. **RESTORE** コマンドを実行します。特定の DB 領域ファイルを別のパスにリストアする場合は、**RENAME** 句を指定します。詳細については、『リファレンス：文とオプション』の「SQL 文」>「RESTORE 文」を参照してください。
8. ユーティリティ・データベースを停止します。
9. テンポラリ DB 領域が引き続きロー・デバイスに残っているか、または正しい長さのファイルとして残っているかを確認します。『システム管理ガイド：第 1 巻』の「データのバックアップ、リカバリ、アーカイブ」>「システム・レベルのバックアップ」>「適切なファイルのバックアップ」を参照してください。IQ テンポラリ・ストアを使用しないサーバの起動については、使用しているプラットフォームの『リリース・ノート』を参照してください。
10. コーディネータ・サーバを起動し、同じ場所にリストアする場合にはセカンダリ・サーバを同期します。 詳細については、「マルチプレックス・サーバの同期」を参照してください。
11. セカンダリ・サーバを起動します。

参照：

- リストア前の作業 (100 ページ)
- マルチプレックス・サーバの同期 (37 ページ)

マルチプレックス環境での選択的なリストア操作

データベースのリストアは、完全なリストアまたは選択的なリストア (読み取り／書き込み DB 領域のみのリストア、読み取り専用 DB 領域のセットのリストア、または読み取り専用ファイルのリストア) のいずれでも可能です。

『システム管理ガイド：第 1 巻』の「データのバックアップ、リカバリ、アーカイブ」と『リファレンス：文とオプション』の「RESTORE 文」を参照してください。

コーディネータ用の読み取り専用バックアップのリストア

次の手順を使用すると、コーディネータの名前を使用するためにユーティリティ・データベースの名前を変更することなく、コーディネータをリストアできます。これは、マルチプレックス・コーディネータに対して読み取り専用の選択的リストアを行うための唯一のサポートされる方法です。

次のプロシージャを使用して、読み取り／書き込みアーカイブから読み取り専用 DB 領域を、またはその逆に誤ってリストアしたために発生した問題を修正します。

1. セカンダリ・ノード名以外の任意のサーバ名を使用してユーティリティ・サーバを起動します。
2. utility_db に接続し、読み取り／書き込みデータベースについて **RESTORE** 文を実行します。**RENAME** 句を使用して、DB ファイルを対応するロケーションに移動します。
3. ユーティリティ・サーバの接続を切断し、停止します。
4. リストアされたデータベースを起動します。データベースを別のロケーションに移動している場合は、**-iqmpx_sn 1** フラグと **-iqmpx_ov 1** フラグを指定してサーバを起動します。
5. 別の RO バックアップのみでバックアップした RO DB 領域について、**ALTER DBSPACE<dbspace name>** OFFLINE を実行します。
6. サーバの接続を切断し、停止します。
7. セカンダリ・ノード名以外の任意のサーバ名を使用してユーティリティ・データベースを起動します。
8. ユーティリティ・サーバに接続し、RO DB 領域についてのリストア・コマンドを実行します。

マルチプレックス・リファレンス

一部の SQL 文、データベース・ユーティリティ・オプション、システム・オブジェクトには、マルチプレックス機能をサポートする特別な構文があります。

参照：

- 管理シェル・スクリプト (36 ページ)
- サーバの管理 (Sybase Central) (35 ページ)

SQL 文

多くの SQL 文には、マルチプレックス機能をサポートする構文があります。

ALTER LOGICAL SERVER 文

データベース内の既存のユーザ定義による論理サーバの設定を変更します。

構文

```
ALTER LOGICAL SERVER  logical-server-name
    { alter_clause }
```

パラメータ

- **alter_clause** ： –

```
{ADD MEMBERSHIP   '(' { ls-member, ... } ')'
| DROP MEMBERSHIP '(' { ls-member, ... } ')'
}
```

- **ls-member** ： –

```
FOR LOGICAL COORDINATOR
| mpx-server-name
```

例

- **例 1** – 次の例は、マルチプレックス・ノードの n1 と n2 を論理サーバ ls1 に追加して、ユーザ定義の論理サーバを変更します。

```
ALTER LOGICAL SERVER ls1 ADD MEMBERSHIP (n1, n2)
```

- **例 2** – 次の例は、コーディネータの論理メンバシップを追加し、現在のコーディネータ・ノード n1 の名前付きメンバシップを ls1 から削除します。

```
ALTER LOGICAL SERVER ls1 ADD MEMBERSHIP (FOR LOGICAL COORDINATOR)
ALTER LOGICAL SERVER ls1 DROP MEMBERSHIP (n1)
```

使用法

マルチプレックスのみに該当します。

logical-server-name は、既存のユーザ定義の論理サーバ名を参照します。言い換えると、組み込みまたは予約済みの論理サーバ名ではありません。SYS.ISYSIQLSMEMBER システム・テーブルには、論理サーバ・メンバシップの定義が格納されています。

論理サーバに追加された、または論理サーバから削除されたメンバ・ノードが論理サーバ接続の受け付けを開始または停止するのは、**ALTER LOGICAL SERVER** に対応する TLV ログがそのノードで再生された後のみです。論理サーバの既存の接続は、ノードが論理サーバから削除されてもそのノードで稼動し続けますが、これらの接続の分散処理は停止します。

次の場合には、エラーが返ります。

- **ADD MEMBERSHIP** 句で指定された *ls-member* が、すでに論理サーバのメンバである。
- **DROP MEMBERSHIP** 句で指定された *ls-member* が論理サーバの既存のメンバではない。
- 論理サーバのメンバシップ変更により、メンバシップの重複チェックが失敗する。

パーミッション

DBA 権限または MPX ADMIN 権限を持っている必要があります。

参照：

- 論理サーバの変更 (Interactive SQL) (69 ページ)

ALTER LOGIN POLICY 文

この文は、2 つの機能を実行します。データベース内の既存のログイン・ポリシーのオプション値を変更する機能と、論理サーバ・アクセスを設定する機能です。

構文

ALTER LS POLICY *policy-name option-value-list*

パラメータ

- **alter-clause**： –

```
{ { ADD | DROP | SET } LOGICAL SERVER
                     ls-assignment-list
       | policy-option-name = policy-option-value [ LOGICAL SERVER
                     ls-override-list ]
}
```

- **ls-assignment-list**： –

```
{ { ls-name, ...} | SERVER | NONE | DEFAULT }
```

- **ls-override-list**： –

```
{ ls-name, ...}
```

- **ls-name**： –

```
{ OPEN | user-defined-ls-name }
```

- **policy-option-value**： –

```
{ UNLIMITED | DEFAULT | value }
```

例

- 例 **1** – 「論理サーバへのアクセス許可設定」と「ログイン・ポリシー・オプションの設定」を参照してください。

使用法

マルチプレックスのみに該当します。

「論理サーバへのアクセス許可設定」と「ログイン・ポリシー・オプションの設定」を参照してください。

パーミッション

DBA 権限または USER ADMIN 権限を持っている必要があります。

参照：

- 論理サーバを変更したときの論理サーバ割り当ての設定 (Interactive SQL) (76 ページ)
- DQP_ENABLED オプション (122 ページ)

論理サーバへのアクセス許可設定

ALTER LOGIN POLICY を使用して、論理サーバへのアクセス許可を設定できます。

例 1

ルート・ログイン・ポリシーが論理サーバの `ls4` と `ls5` へのアクセスを許可し、ログイン・ポリシー `lp1` が論理サーバの割り当てなしに存在するとします。次の

文は、ログイン・ポリシー lp1 を論理サーバの ls4 と ls5 に実質的に割り当てます。

論理サーバ ls1 をログイン・ポリシー lp1 に割り当てます。

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

例 2

次の文は、論理サーバの ls2 と ls3 にログイン・ポリシー lp1 からアクセスすることを許可します。

```
ALTER LOGIN POLICY lp1 ADD LOGICAL SERVER ls2, ls3
```

例 3

ログイン・ポリシー lp1 を変更して、ls3 と ls4 にのみにアクセスを許可します。

```
ALTER LOGIN POLICY lp1 ADD LOGICAL SERVER ls4
```

```
ALTER LOGIN POLICY lp1 DROP LOGICAL SERVER ls1, ls2
```

または

```
ALTER LOGIN POLICY lp1 SET LOGICAL SERVER ls3, ls4
```

例 4

ログイン・ポリシー lp1 を変更して、すべての論理サーバへのアクセスを拒否します。

```
ALTER LOGIN POLICY lp1 SET LOGICAL SERVER NONE
```

例 5

ログイン・ポリシー lp1 の現在の論理サーバ割り当てを削除し、 ルート・ログイン・ポリシーの論理サーバ割り当てを継承させます。

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

使用法

マルチプレックスのみに該当します。

ADD 句、**DROP** 句、または **SET** 句を使用すると、ログイン・ポリシーの論理サーバ割り当てを設定できます。

- **ADD** – 新しい論理サーバ割り当てをログイン・ポリシーに追加します。
- **DROP** – ログイン・ポリシーから既存の論理サーバ割り当てを削除します。
- **SET** – ログイン・ポリシーのすべての論理サーバ割り当てを新しい一連の論理サーバに置き換えます。

ADD 句、**DROP** 句、または **SET** 句のいずれか 1 つのみを使用します。**SERVER**、**NONE**、**DEFAULT** は、**SET** 句でのみ使用します。**ls-assignment list** または **ls-override list** ごとに特定の論理サーバ名を 1 回のみ指定します。

次の場合には、エラーが返ります。

- **ADD** 句で指定された論理サーバが、すでにログイン・ポリシーに割り当てられている。
- **DROP** 句で指定された論理サーバが、ログイン・ポリシーに割り当てられていない。
- 論理サーバ割り当ての変更により、割り当てられている論理サーバ間でメンバシップの重複が発生する。

`SYS.ISYSIQLOGINPOLICYLSINFO` には、論理サーバ割り当ての情報が格納されています。ログイン・ポリシーのログイン・ポリシー・オプションの各論理サーバ上書きについては、対応するローが `ISYSIQLOGINPOLICYLSINFO` に存在します。

ログイン・ポリシー・オプションの設定

ALTER LOGIN POLICY を使用して、ログイン・ポリシー・オプションを設定できます。

例

次の例は、2 つの論理サーバのログイン・ポリシー設定を上書きします。論理サーバ `ls1` で分散クエリ処理を有効にし、論理サーバ `ls2` で最大接続数を増やします。

```
ALTER LOGIN POLICY lp1 dqp_enabled=ON LOGICAL SERVER ls1;
```

```
ALTER LOGIN POLICY lp2 max_connections=20 LOGICAL SERVER ls2;
```

使用法

マルチプレックスのみに該当します。

論理サーバ・レベルの上書きは、特定のログイン・ポリシー・オプションが、異なる論理サーバに対して異なる設定を持つことを意味します。`SYS.ISYSIQLSLOGINPOLICYOPTION` には、論理サーバ上書きのログイン・ポリシー・オプション値が格納されています。ログイン・ポリシーのログイン・ポリシー・オプションの各論理サーバ上書きについては、対応するローが `ISYSIQLSLOGINPOLICYOPTION` に存在します。

注意： 論理サーバ上書き設定は、ログイン・ポリシー・オプションの **max_connections** と **dqp_enabled** にのみ指定できます (**dqp_enabled** オプションは、マルチプレックス・サーバにのみ影響します。「マルチプレックス・リファレンス」>「データベース・オプション」>「dqp_enabled オプション」を参照してくだ

さい)。ルート・ログイン・ポリシーに論理サーバの上書きを指定することはできません。

表 17 : ログイン・ポリシー・オプション

オプション	説明	値	ROOT ポリシーの初期値	適用対象
dqp_enabled	ON の場合、ログイン・ポリシーに割り当てられているユーザに対して分散クエリ処理が有効になります。	ON、OFF	ON	マルチプレックス・サーバのみ。DBA 権限を持つユーザを含むすべてのユーザ。
locked	このオプションの値が ON の場合、ユーザが新しい接続を確立するのを禁止します。	ON、OFF	OFF	DBA 権限を持っていないユーザのみ。
max_connections	ユーザに許可された同時接続の最大数。	0 – 2147483647	Unlimited	DBA 権限を持っていないユーザのみ。
max_days_since_login	同一ユーザによる前回のログイン時から次回のログイン時までの最大許容日数。	0 – 2147483647	Unlimited	DBA 権限を持っていないユーザのみ。
max_failed_login_attempts	前回の正常なログイン時以降に行った、ユーザ・アカウントがロックアウトされる原因となるログイン試行失敗回数の最大値。	0 – 2147483647	Unlimited	DBA 権限を持っていないユーザのみ。
max_non_dba_connections	DBA 権限を持っていないユーザが実行できる同時接続の最大数。このオプションは、ルート・ログイン・ポリシーでのみサポートされています。	0 – 2147483647	Unlimited	DBA 権限を持っていないユーザのみ。ルート・ログイン・ポリシーのみをサポート。

オプション	説明	値	ROOT ポリシーの初期値	適用対象
`password_expiry_on_next_login`	このオプションの値が ON に設定されている場合は、ユーザのパスワードは次回ログイン時に有効期限が切れます。	ON、OFF	OFF	DBA 権限を持つすべてのユーザ。
`password_grace_time`	パスワードの有効期限が切れるまでの日数 (この間は、ログインすることはできますが、デフォルトの post_login プロシージャによって警告が発行されます)。	0 – 2147483647	0	DBA 権限を持つすべてのユーザ。
`password_life_time`	パスワードの変更が必要になるまでの最大日数。	0 – 2147483647	Unlimited	DBA 権限を持つすべてのユーザ。

注意： Sybase IQ では、**MULTIPLEX SERVER** 上書き句がサポートされなくなりました。ログイン・ポリシー・オプションに対して、**MULTIPLEX SERVER** 上書き句を指定して **ALTER LOGIN POLICY** 文を使用すると、エラーが返ります。

ALTER LS POLICY 文

データベース内の既存のルート論理サーバ・ポリシーの一部またはすべてのオプション値を変更します。

構文

ALTER LS POLICY *policy-name option-value-list*

パラメータ

- **option-value-list：** –

  {*option-name*=*value*}

例

- **例 1** – 次の例では、論理サーバ・ポリシーを変更します。

```
ALTER LS POLICY root
ALLOW_COORDINATOR_AS_MEMBER=ON;
```

注意： ALLOW_COORDINATOR_AS_MEMBER は唯一の論理サーバ・ポリシー・オプションで、ルートは唯一の論理サーバ・ポリシーです。論理サーバ・ポリシーは作成できません。

使用法

マルチプレックスのみに該当します。

パーミッション

DBA 権限または MPXADMIN 権限を持っている必要があります。

参照：
- ルート論理サーバ・ポリシーの変更 (Interactive SQL) (69 ページ)

ALTER MULTIPLEX RENAME 文

マルチプレックスの名前を変更し、マルチプレックス名を SYS.ISYSIQINFO システム・テーブルに格納します。

構文

```
ALTER MULTIPLEX RENAME  multiplex-name
```

使用法

マルチプレックスのみに該当します。

作成されたマルチプレックスには、コーディネータの名前が付けられます。マルチプレックス名は、マルチプレックス・フォルダ内でマルチプレックスを識別するために Sybase Central のみで使用されます。この文は自動的にコミットされます。

パーミッション

DBA 権限または MULTIPLEX ADMIN 権限を持っている必要があります。

ALTER MULTIPLEX SERVER 文

特定のサーバの名前、カタログ・ファイル・パス、役割、またはステータスを変更します。

構文

構文 1：

```
ALTER MULTIPLEX SERVER server-name server-option
```

構文 2：

```
ALTER MULTIPLEX SERVER PRIVATE NULL
```

パラメータ

- **server-option： –**

```
{ RENAME new-server-name
  | DATABASE 'dbfile'
  | ROLE { WRITER | READER | COORDINATOR }
  | STATUS { INCLUDED | EXCLUDED }
  | ASSIGN AS FAILOVER SERVER
  | host-port-list }
```

- **host-port-list： –** { **HOST** ' *hostname* ' **PORT** *port number* ...} { **PRIVATE HOST** ' *hostname* ' **PORT** *port number* ...}

注意： サーバを除外する前に、そのサーバを停止することをおすすめします。除外されるサーバは、停止されていない場合、除外された後に自動的に停止します。また、除外されたサーバをマルチプレックスに再び参加させるには、**ALTER MULTIPLEX SERVER** *server-name* **STATUS INCLUDED** を実行し、同期を行う必要があります。

例

- **例** – 次の例は、セカンダリ・サーバ `mpx_writer1` を除外します。

```
ALTER MULTIPLEX SERVER mpx_writer1 STATUS EXCLUDED
```

使用法

マルチプレックスのみに該当します。

マルチプレックス・サーバを次のように変更します。

RENAME – 特定のサーバの名前を変更します。サーバは自動的に停止します。次の再起動時に新しい名前が必要となります。

DATABASE – 特定のサーバのカタログ・ファイル・パスを変更します。サーバは自動的に停止し、次回の起動から新しいカタログ・パスを使用します。カタログ・ファイル自体の移動は、ユーザの責任で行う必要があります (Sybase Central に表示されない場合があります)。

ROLE – 特定のサーバの役割を変更します。ユーザは、コーディネータの役割を変更したり、コーディネータでないサーバをコーディネータに変更したりすること

はできません。ライタ・ノードの役割がリーダに変更された場合、サーバは停止します。

STATUS – 特定のサーバのステータスを変更します。フェールオーバ・ノードは、そのノードが除外される最後のノードである場合を除き、除外することはできません。除外後に、サーバは自動的に停止します。ノードの追加後は、そのノードを同期し、再起動する必要があります。

ASSIGN – 特定のサーバを新しいフェールオーバ・サーバとして指定します。除外済みのステータスにあるノードは使用できません。ASSIGN AS FAILOVER 句は、他の **ALTER MULTIPLEX SERVER** 句とともに使用できないスタンドアロン句です。

コーディネータが実行中である必要がありますが、**ALTER MULTIPLEX SERVER** コマンドはマルチプレックス内のいずれのサーバからでも実行できます(すべての DDL 文をコーディネータ上で実行することをおすすめします)。役割をリーダからライタに変更する以外のすべての場合に、名前付きサーバは自動的に停止します。

注意： サーバを除外する前に、そのサーバを停止することをおすすめします。除外されるサーバは、停止されていない場合、除外された後に自動的に停止します。また、除外されたサーバをマルチプレックスに再び参加させるには、`ALTER MULTIPLEX SERVER` *`server-name`* `STATUS INCLUDED` を実行し、同期を行う必要があります。

パーミッション

DBA 権限または MULTIPLEX ADMIN 権限を持っている必要があります。

参照：

- マルチプレックス・サーバの名前の変更 (Interactive SQL) (23 ページ)

COMMENT ON LOGICAL SERVER 文

ユーザ定義の論理サーバにコメントを追加します。

構文

```
COMMENT ON LOGICAL SERVER logical-server-name IS 'comment'
```

例

- **例** – 次の例は、ユーザ定義の論理サーバ *ls1* についてのコメントを作成します。

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

使用法

マルチプレックスのみに該当します。

パーミッション

DBA 権限または MPX ADMIN 権限を持っている必要があります。

参照：

- 論理サーバへのコメントの追加 (Interactive SQL) (69 ページ)

CREATE LOGICAL SERVER 文

ユーザ定義の論理サーバを作成します。

構文

```
CREATE LOGICAL SERVER  logical-server-name [MEMBERSHIP
    '(' { ls-member, ...} ')']
```

パラメータ

- **ls-member**： **– FOR LOGICAL COORDINATOR** | *mpx-server-name*

例

- **例 1 –** 次の例は、メンバとして 3 つのマルチプレックス・ノードを含む、ユーザ定義の論理サーバ *ls1* を作成します。

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

使用法

マルチプレックスのみに該当します。

カタログには、論理サーバとそのメンバシップの定義が格納されています。コーディネータとの論理的なメンバシップを定義するには、MEMBERSHIP 句に FOR LOGICAL COORDINATOR を指定します。

論理サーバの作成時にメンバを指定しないと、論理サーバは空で作成されます。

注意： OPEN 論理サーバや SERVER 論理サーバなどに対する暗黙的な論理サーバ・メンバシップ定義が格納されることはありません。

SYS.ISYSLOGICALSERVER システム・テーブルには、論理サーバについての情報が格納されています。

SYS.ISYSLOGICALMEMBER システム・テーブルには、論理サーバ・メンバシップ定義についての情報が格納されています。

logical-server-name には、次を除く任意のユーザ定義の識別子を指定できます。

- OPEN
- SERVER
- NONE
- DEFAULT
- COORDINATOR
- ALL

ルート論理サーバ・ポリシーの ALLOW_COORDINATOR_AS_MEMBER オプションを ON から OFF に変更しても、カタログに格納されているメンバシップ情報に影響はありません。論理サーバ設定の有効性に影響するのみです。

ALLOW_COORDINATOR_AS_MEMBER が OFF に設定されている場合でも、マルチプレックス・サーバ名を指定するか、FOR LOGICAL COORDINATOR 句を使用して、現在のコーディネータへの論理サーバのメンバシップを定義できます。メンバシップ定義はカタログに格納されますが、マルチプレックス・サーバがコーディネータとして動作している間は非アクティブです。

パーミッション

DBA 権限または MULTIPLEX ADMIN 権限を持っている必要があります。

参照：

- 論理サーバの作成 (Interactive SQL) (68 ページ)

CREATE MULTIPLEX SERVER 文

マルチプレックス・サーバを作成します。

構文

変数宣言：

```
CREATE MULTIPLEX SERVER server-name DATABASE 'dbfile
' host-port list [ ROLE {  READER |  WRITER  } ] [  STATUS | {
INCLUDED |  EXCLUDED } ]
```

パラメータ

- **host-port-list**： – {[**PRIVATE**] **HOST** ' *hostname* ' **PORT***port number* }

例

- **例 1 –**「マルチプレックス・サーバの追加 (Interactive SQL)」を参照してください。

使用法

マルチプレックスのみに該当します。

サーバ・パスに UNIX のソフト・リンク (シンボリック・リンク) を使用することを計画している場合、ソフト・リンクを作成してから **CREATE MULTIPLEX SERVER** を実行します。新しいサーバを起動する場合、データベース・ファイル・パスは、サーバを作成したときに指定したデータベース・ファイル・パスに一致している必要があります。

サーバ起動オプション **-n** の規則に従って、マルチプレックス・サーバの名前 (*server-name*) を選択します。『ユーティリティ・ガイド』の「第 1 章 データベース・サーバの実行」>「データベース・サーバの起動」を参照してください。

最初のマルチプレックス・サーバを作成すると、コーディネータ・ノードとセカンダリ・ノードのローが SYS.ISYSIQMPXSERVER に追加されます。トランザクション・ログは、この操作を 2 つの異なる **CREATE MULTIPLEX SERVER** コマンドとして記録します。1 つはコーディネータ・ノード用で、他の 1 つはセカンダリ・ノード用です。

SYS.ISYSIQMPXSERVER システム・テーブルには、HOST ホスト名 PORT ポート名のペアが host:port[;host:port…] の形式で connection_info 文字列に格納されています。

注意： マルチプレックス・サーバを実行中のコンピュータに、異なるネットワーク・アドレスにマップされている複数の冗長ネットワーク・カードがある場合は、複数の host:port ペアを使用してください。

DATABASE、host-port list、ROLE、STATUS の各句は、どのような順序で指定してもかまいません。デフォルトの ROLE は READER です。デフォルトの STATUS は INCLUDED です。

host-port-list のキーワード PRIVATE は、特定の HOST PORT ペアがプライベート相互接続用であることを指定します。MIPC 用の個別のプライベート相互接続を使用することにより、可用性が高く、パフォーマンスに優れたネットワーク設定が実現できます。Sybase IQSybase IQ は、プライベート・ポートを自動的に開きます。サーバを起動するために使用される host-port-list にリストする必要はありません。 すべてのパブリック・ポートとプライベート・ポートは、競合を避けるために、ユニークなポート番号を必要とします。

サーバの追加時にはコーディネータが実行中である必要がありますが、**CREATE MULTIPLEX SERVER** コマンドはマルチプレックス内のいずれのサーバからでも実行できます。

この文は自動的にコミットされます。

パーミッション

DBA 権限または MULTIPLEX ADMIN 権限を持っている必要があります。

参照：

- マルチプレックスの管理 (Interactive SQL) (21 ページ)
- マルチプレックス・サーバの追加 (Interactive SQL) (22 ページ)

DROP LOGICAL SERVER 文

ユーザ定義の論理サーバを削除します。

構文

```
DROP LOGICAL SERVER logical-server-name
```

例

- **例 1 –** 次の例は、ユーザ定義の論理サーバ *ls1* を削除します。

  ```
  DROP LOGICAL SERVER ls1
  ```

使用法

マルチプレックスのみに該当します。

Sybase IQ は、論理サーバの削除時に、次のカタログ変更を内部で実行します。

- 論理サーバのすべてのメンバシップ定義を削除します。
- 削除対象の論理サーバへの明示的な割り当てを含む各ログイン・ポリシーから論理サーバの割り当てを削除します。 削除対象の論理サーバがログイン・ポリシーに割り当てられている唯一の論理サーバの場合、Sybase IQ はそのログイン・ポリシーの論理サーバ割り当てを NONE に設定します。
- 論理サーバ・エントリを ISYSIQ.LOGICALSERVER から削除します。

パーミッション

DBA 権限または MULTIPLEX ADMIN 権限を持っている必要があります。

参照：

- 論理サーバの削除 (Interactive SQL) (70 ページ)

DROP MULTIPLEX SERVER 文

マルチプレックスからサーバを削除します。

構文

```
DROP MULTIPLEX SERVER {server-
name} [drop_mpx_server_clause]
```

パラメータ

- **drop_mpx_server_clause： –** { **WITH DROP MEMBERSHIP** | **WITH DROP LOGICAL SERVER** }

例

- **例 1 –**

```
DROP MULTIPLEX SERVER writer1
```

使用法

マルチプレックスのみに該当します。

削除する前に各マルチプレックス・サーバを停止することをおすすめします。この文は自動的にコミットされます。

削除されるサーバが、停止されていない場合は、この文の実行後に自動的に停止します。

最後のセカンダリ・サーバを削除すると、マルチプレックスはシンプレックスに変換されます。マルチプレックス内の最後のセカンダリ・サーバの削除後に、コーディネータは自動的に停止します。必要に応じて再起動する必要があります。マルチプレックスがシンプレックスに変換されるときに適用される一連のルールの詳細については、セクション 3.3.1 を参照してください。

WITH DROP MEMBERSHIP 句 – **DROP MULTIPLEX SERVER** は、削除するマルチプレックス・サーバに、1 つまたは複数の論理サーバ・メンバシップが存在すると、エラーで失敗します。**WITH DROP MEMBERSHIP** 句を指定すると、論理サーバはそのすべてのメンバシップとともに削除されます。

句 **WITH DROP LOGICAL SERVER** – 最後のセカンダリ・サーバを削除する場合、1 つまたは複数のユーザ定義の論理サーバが存在すると、**DROP MULTIPLEX SERVER** コマンドは失敗します。**WITH DROP LOGICAL SERVER** 句を指定すると、最後のセカンダリ・サーバはすべてのユーザ定義の論理サーバとともに削除されます。

注意： **WITH DROP LOGICAL SERVER** 句は、最後のセカンダリ・サーバを削除する場合のみ有効です。他の場合は、エラーが報告されます。

パーミッション

DBA 権限または MULTIPLEX ADMIN 権限を持っている必要があります。

参照：

- マルチプレックス・サーバを削除するための要件 (25 ページ)
- コーディネータの障害 (44 ページ)
- マルチプレックス・サーバの削除 (Interactive SQL) (25 ページ)

データベース・オプション

マルチプレックスでは、データベース・オプションを使用して、接続、分散クエリ処理、マルチプレックスのノード間通信を制御します。

DQP_ENABLED オプション

テンポラリ・データベース・オプション **dqp_enabled** を使用すると、接続レベルで DQP を有効または無効にすることができます。

指定できる値

ON、OFF

デフォルト値

ON

スコープ

個々の接続に一時的に設定できます。このオプションはすぐに有効になり、現在の接続のみに影響します。

説明

テンポラリ・データベース・オプション **dqp_enabled** を OFF に設定することによって、現在の接続における DQP を無効にすることができます。このオプションを ON (デフォルト値) に設定することによって現在の接続における DQP を有効にすることができますが、それが可能なのは現在の接続の論理サーバに対するユーザのログイン・ポリシーによって、そのユーザに対して DQP が有効になっている場合のみです。

ユーザのログイン・ポリシーに基づいて DQP が無効にされている場合、**dqp_enabled** を ON に設定すると、次のエラーが返ります。

```
Invalid setting for option 'dqp_enabled'
```

注意： ユーザのログイン・ポリシー・オプションへのすべての変更は、新しい接続のみに影響します。既存の接続のログイン・ポリシー・オプション設定は、接続が最初に確立した時点の設定に基づきます。

参照：

- ALTER LOGIN POLICY 文 (108 ページ)

MPX_AUTOEXCLUDE_TIMEOUT オプション

コーディネータ・ノード上でのセカンダリ・ノードの自動除外のタイムアウト。この値を 0 にすると、ノードは自動的に除外されません。このオプションは、指定済みのフェールオーバ・ノードには適用されません。

指定できる値

0 ～ 10080 分 (1 週間)。値は、`MPX_HEARTBEAT_FREQUENCY` の設定でちょうど割り切れる値である必要があります。たとえば、`MPX_HEARTBEAT_FREQUENCY` 設定が 120 (2 分) の場合、`MPX_AUTOEXCLUDE_TIMEOUT` は 2 で割り切れる値に設定する必要があります。

デフォルト値

60 分

スコープ

PUBLIC グループのみに設定できます。このオプションを設定するには、DBA パーミッションが必要です。設定はすぐに有効になり、サーバの再起動後も保持されます。

MPX_HEARTBEAT_FREQUENCY オプション

ハートビート・スレッドがウェイクアップし、セカンダリ・ノードの接続プールをクリーンアップするまでの間隔。

指定できる値

2 秒 ～ 3600 秒

デフォルト値

60 秒

スコープ

PUBLIC グループのみに設定できます。このオプションを設定するには、DBA パーミッションが必要です。設定はすぐに有効になり、サーバの再起動後も保持されます。

MPX_IDLE_CONNECTION_TIMEOUT オプション

セカンダリ・ノード上の接続プールにある未使用の接続がクローズされるまでの待機時間。

指定できる値
0 秒以上無制限

デフォルト値
600 秒

スコープ
PUBLIC グループのみに設定できます。このオプションを設定するには、DBA パーミッションが必要です。設定はすぐに有効になり、サーバの再起動後も保持されます。

MPX_MAX_CONNECTION_POOL_SIZE オプション

セカンダリ・ノード上の接続プールで許容される接続の最大数。

指定できる値
1 〜 1000

デフォルト値
10

スコープ
PUBLIC グループのみに設定できます。このオプションを設定するには、DBA パーミッションが必要です。設定はすぐに有効になり、サーバの再起動後も保持されます。

説明
INC 接続は、セカンダリ・ノードとコーディネータ・ノードのサーバ間接続です。INC 接続は、DDL 操作か読み取り／書き込み操作を行うセカンダリ・サーバ上の各ユーザ接続と関連付けられます。コマンドがコミットまたはロールバックされるまで、接続はアクティブです。その後接続はプールに戻されます。これらのトランザクションがすぐに終了する場合、`MPX_MAX_CONNECTION_POOL_SIZE` のデフォルトの設定で、DDL 操作か読み取り／書き込み操作を実行している多数のユーザ接続に十分に対応できます。多数の同時接続が DDL 操作または読み取り／書き込み操作を実行しているか、トランザクションに時間がかかる場合、`MPX_MAX_CONNECTION_POOL_SIZE` の値を大きくします。 たとえば、多数のユーザ接続がコミットしないで同時にロードを行う場合、この値を大きくします。

`MPX_MAX_CONNECTION_POOL_SIZE` を超えると、`"SQL Anywhere Error -1004000: The number of connections in the connection pool have exceeded the upper limit"` が返されます。

必要なプールのサイズを見積もるには、`-gm` サーバ・オプションを設定することを検討します。`-gm` の設定は、セカンダリ・サーバに接続できるユーザ数を示します。INC 接続は含まれていませんが、後にこの数値に加算されます。アプリケーションの稼働条件を使用して、ユーザごとに発生する可能性のある読み取り／書き込み操作か DDL 操作の数を調べ、それに応じてプールのサイズを大きくします。

各接続 (INC またはユーザ) は、`-gn` 設定とコアの数に応じてメモリ・オーバヘッドを実行します。メモリの負荷とスレッド接続は、Sybase IQ サーバの応答時間に影響を与える可能性があります。詳細については、『パフォーマンス＆チューニング・ガイド』の「システム・リソースの管理」>「メモリ使用の最適化」>「バッファ・キャッシュ・サイズの決定」>「メモリ・オーバヘッド」 を参照してください。

参照：

- プール済み接続 (9 ページ)
- パラレル・ロード用の複数のライター (6 ページ)

MPX_MAX_UNUSED_POOL_SIZE オプション

セカンダリ・ノード上の接続プールでの未使用接続の最大数。

指定できる値

0 ～最大プール・サイズ

デフォルト値

0

スコープ

PUBLIC グループのみに設定できます。このオプションを設定するには、DBA パーミッションが必要です。設定はすぐに有効になり、サーバの再起動後も保持されます。

参照：

- プール済み接続 (9 ページ)
- パラレル・ロード用の複数のライター (6 ページ)

MPX_WORK_UNIT_TIMEOUT オプション

マルチプレックス DQP リーダが、完了していない分散作業を他の DQP ワーカ・ノードに再割り当てするまでの時間 (秒数)。

指定できる値
0 ～ 3600 秒。

DQP 作業単位は、通常わずか数秒の長さに分けられます。 ワーカ・ノードがオフラインになるか異常に高い負荷をかけられた場合、以前にそのワーカ・ノードに割り当てられた DQP 作業は、指定されたタイムアウト後に他のノードに再割り当てされます。

デフォルト値
60 秒

通常、このオプションをデフォルト値から変更する必要はありません。ただし、まれに、クエリの中間結果が非常に大きいために個々の作業単位がタイムアウトする場合には、このオプションを大きくします。

ネットワークやサーバの信頼性が低いために分散作業が失われ、タイムアウト間隔が長くなりすぎる場合は、このオプションを小さくします。このオプションを下げすぎると、不必要にタイムアウトが早くなります。

スコープ
個々の接続に一時的に設定するか、サーバの再起動後も保持されるように設定できます。このオプションを設定するには、DBA パーミッションが必要です。設定は、すぐに有効になります。

システム・テーブル

一部のシステム・テーブルは、マルチプレックス機能をサポートしています。

ISYSIQINFO システム・テーブル

このテーブルは、**CREATE DATABASE** を使用して Sybase IQ データベースを作成したときに定義された、データベースの特性を示します。このテーブルに含まれるローの数は常に 1 つのみです。

カラム名	カラム・タイプ	説明
last_full_backup	TIMESTAMP	前回のバックアップの完了時刻。

カラム名	カラム・タイプ	説明
last_incr_backup	TIMESTAMP	前回のインクリメンタル・バックアップの完了時刻。
create_time	TIMESTAMP NOT NULL	作成された日時。
update_time	TIMESTAMP NOT NULL	前回の更新日時。
file_format_version	UNSIGNED INT NOT NULL	このデータベースのファイルのファイル・フォーマット番号。
cat_format_version	UNSIGNED INT NOT NULL	このデータベースのカタログ・フォーマット番号。
sp_format_version	UNSIGNED INT NOT NULL	このデータベースのストアド・プロシージャ・フォーマット番号。
block_size	UNSIGNED INT NOT NULL	データベースに定義したブロック・サイズ。
chunk_size	UNSIGNED INT NOT NULL	データベースに定義したブロック・サイズとページ・サイズを基に決められた、各部分のブロック数。
file_format_date	CHAR(10) NOT NULL	ファイル・フォーマット番号の最終更新日付。
dbsig	BINARY(136) NOT NULL	カタログが内部で使用。
multiplex_name	CHAR(128) NULL	カタログが内部で使用。
last_multiplex_mode	TINYINT NULL	カタログ読み取り／書き込みを最後にオープンしたサーバのモード。次の値のいずれかになります。 • 0 – シングル・ノード • 1 – リーダ • 2 – コーディネータ • 3 – ライタ

制約 プライマリ・キー (create_time)

ISYSIQLOGICALSERVER システム・テーブル

ISYSIQLOGICALSERVER には、論理サーバの情報、論理サーバと関連付けられた論理サーバ・ポリシーとの対応情報が格納されています。

参照：

- SYSIQLOGICALSERVER システム・ビュー (129 ページ)

ISYSIQLOGINPOLICYLSINFO システム・テーブル

ISYSIQLOGINPOLICYLSINFO には、ログイン・ポリシーの論理サーバ割り当ての情報が格納されています。

参照：

- SYSIQLOGINPOLICYLSINFO システム・ビュー (130 ページ)

ISYSIQLSLOGINPOLICYOPTION システム・テーブル

ISYSIQLSLOGINPOLICYOPTION には、論理サーバ・レベルの設定を含むログイン・ポリシー・オプション値が格納されています。

参照：

- SYSIQLSPOLICYOPTION システム・ビュー (132 ページ)

ISYSIQLSMEMBER システム・テーブル

ISYSIQLSMEMBER には、論理サーバのメンバシップ情報が格納されています。

参照：

- SYSIQLSMEMBER システム・ビュー (130 ページ)

ISYSIQLSPOLICY システム・テーブル

ISYSIQLSPOLICY には、論理サーバ・ポリシーが格納されています。

参照：

- SYSIQLSPOLICY システム・ビュー (134 ページ)

ISYSIQLSPOLICYOPTION システム・テーブル

ISYSIQLSPOLICYOPTION には、論理サーバ・ポリシー・オプションが格納されています。

ISYSIQMPXSERVER システム・テーブル

ISYSIQMPXSERVER には、特定のマルチプレックス・ノードのメンバシップ・プロパティとバージョン・ステータス・データが格納されています。

参照：

- SYSIQMPXSERVER システム・ビュー (133 ページ)

システム・ビュー

一部のシステム・ビューは、マルチプレックス機能をサポートしています。

SYSIQLOGICALSERVER システム・ビュー

システム・テーブル ISYSIQLOGICALSERVER を読みやすい形式で表示します。

ISYSIQLOGICALSERVER システム・テーブルには、論理サーバの情報と、関連付けられた論理サーバ・ポリシーの情報が格納されています。

カラム名	カラム・タイプ	説明
ls_id	UNSIGNED BIGINT NOT NULL	論理サーバの ID 番号。
ls_object_id	UNSIGNED BIGINT NOT NULL	論理サーバのオブジェクト ID 番号。
ls_policy_id	UNSIGNED BIGINT NOT NULL	論理サーバ・ポリシーの ID 番号。
ls_name	CHAR(128) NOT NULL UNIQUE	論理サーバ名。

基本となるシステム・テーブルでの制約

- プライマリ・キー (ls_id)
- object_id 外部キー (ISYSOBJECT)
- ls_policy_id 外部キー (ISYSIQLSPOLICY)

参照：

- ISYSIQLOGICALSERVER システム・テーブル (128 ページ)

SYSIQLOGINPOLICYLSINFO システム・ビュー

テーブル ISYSIQLOGINPOLICYLSINFO を読みやすい形式で表示します。

システム・テーブル ISYSIQLOGNPOLICYLSINFO には、ログイン・ポリシーの論理サーバ割り当ての情報が格納されています。

カラム名	カラム・タイプ	説明
login_policy_id	UNSIGNED BIGINT NOT NULL	ログイン・ポリシーの ID 番号。
ls_id	UNSIGNED BIGINT NOT NULL	論理サーバの ID 番号。

基本となるシステム・テーブルでの制約

- プライマリ・キー (login_policy_id, ls_id)
- login_policy_id 外部キー (ISYSLOGINPOLICY)
- ls_id 外部キー (ISYSIQLOGICALSERVER)

参照：

- ISYSIQLOGINPOLICYLSINFO システム・テーブル (128 ページ)

SYSIQLSMEMBER システム・ビュー

テーブル ISYSIQLSMEMBER を読みやすい形式で表示します。

`ISYSIQLSMEMBER` には、論理サーバの情報と対応するマルチプレックス・サーバの情報が格納されています。

カラム名	カラム・タイプ	説明
ls_id	UNSIGNED BIGINT NOT NULL	論理サーバの ID 番号。
logical_membership_type	TINYINT NOT NULL	論理メンバシップのタイプ。
mpx_server_id	UNSIGNED INT NOT NULL	マルチプレックス・サーバの ID 番号。
membership_info	UNSIGNED INT NOT NULL	メンバシップ情報。

基本となるシステム・テーブルでの制約

- プライマリ・キー (ls_id、logical_membership_id、mpx_server_id)
- ls_id 外部キー (ISYSIQLOGICALSERVER)

論理サーバのメンバシップがマルチプレックス・サーバ名を使用して定義されている場合、logical_membership_type の値は 0 で、mpx_server_id はマルチプレックス・サーバのサーバ ID です。

コーディネータの論理メンバシップの場合、mpx_server_id は 0 で logical_membership_type は 1 です。

参照：

- ISYSIQLSMEMBER システム・テーブル (128 ページ)

SYSIQLSMEMBERS 統合システム・ビュー

ISYSIQLSMEMBERS システム・テーブルからの論理サーバのメンバシップ情報を読みやすい形式で表示します。

カラム名	カラム・タイプ	説明
ls_id	UNSIGNED BIGINT NOT NULL	論理サーバの ID 番号。
ls_name	CHAR(128) NOT NULL	論理サーバ名。
server_id	UNSIGNED INT NOT NULL	サーバ名を使用して定義されているメンバシップの場合はメンバのマルチプレックス・サーバ ID。コーディネータの論理メンバシップの場合は 0。
server_name	CHAR(128) NOT NULL	サーバ名を使用して定義されているメンバシップの場合はメンバのマルチプレックス・サーバ名。コーディネータの論理メンバシップの場合は 'LOGICAL COORDINATOR'。
membership_type	TINYINT NOT NULL	サーバ名を使用して定義されているメンバシップの場合は 0、コーディネータの論理メンバシップの場合は 1。

SYSIQLSLOGINPOLICIES システム・ビュー

ログイン・ポリシーからの論理サーバ割り当てについての情報を読みやすい形式で表示します。

このシステム・ビューは、SYSIQLOGICALSERVER、ISYSIQLOGINPOLICYLSINFO、ISYSLOGINPOLICY からの情報を統合して表示します。

カラム名	カラム・タイプ	説明
ls_id	UNSIGNED BIGINT NOT NULL	論理サーバの ID 番号。

カラム名	カラム・タイプ	説明
ls_name	CHAR(128)	論理サーバ名。
login_policy_id	UNSIGNED BIGINT NOT NULL	ログイン・ポリシーの ID 番号。
login_policy_name	char(128)	ログイン・ポリシーの名前。

SYSIQLSLOGINPOLICYOPTION システム・ビュー

テーブル ISYSIQLSLOGINPOLICYOPTION を読みやすい形式で表示します。

ISYSIQLSLOGINPOLICYOPTION テーブルには、ログイン・ポリシー・オプション値の論理サーバ・レベルの設定が格納されています。

カラム名	カラム・タイプ	説明
login_policy_id	UNSIGNED BIGINT NOT NULL	ログイン・ポリシーの ID 番号。
ls_id	UNSIGNED BIGINT NOT NULL	論理サーバの ID 番号。
login_option_name	CHAR(128) NOT NULL	ログイン・ポリシー・オプションの名前。
login_option_value	LONG VARCHAR NOT NULL	ログイン・ポリシー・オプションの値。

基本となるシステム・テーブルでの制約

- プライマリ・キー (login_policy_id、ls_id、login_option_name)
- login_policy_id 外部キー (ISYSLOGINPOLICY)
- ls_id 外部キー (ISYSIQLOGICALSERVER)

SYSIQLSPOLICYOPTION システム・ビュー

テーブル ISYSIQLSPOLICYOPTION を読みやすい形式で表示します。

ISYSIQLSPOLICYOPTION テーブルには、論理サーバ・ポリシー・オプションが格納されています。

カラム名	カラム・タイプ	説明
ls_policy_id	UNSIGNED BIGINT NOT NULL	ログイン・ポリシーの ID 番号。
ls_policy_option_name	CHAR(128) NOT NULL	論理サーバ・ポリシー・オプション名。

カラム名	カラム・タイプ	説明
ls_policy_option_value	LONG VARCHAR NOT NULL	論理サーバ・ポリシー・オプション値。

基本となるシステム・テーブルでの制約

- プライマリ・キー (ls_policy_id、ls_policy_option_name)
- ls_policy_id 外部キー (ISYSIQLSPOLICY)

参照：

- ISYSIQLSLOGINPOLICYOPTION システム・テーブル (128 ページ)

SYSIQMPXSERVER システム・ビュー

テーブル `ISYSIQMPXSERVER` を読みやすい形式で表示します。

ISYSIQMPXSERVER システム・テーブルには、特定のマルチプレックス・ノードのメンバシップ・プロパティおよびバージョン・ステータス・データが格納されています。

カラム名	カラム・タイプ	説明
server_id	UNSIGNED INT NOT NULL	サーバの ID 番号。
server_name	CHAR(128) NOT NULL	サーバ名。大文字と小文字の区別があるユニークな名前である必要があります。
role	TINYINT NOT NULL	コーディネータ、リーダ、またはライタ。
status	TINYINT NOT NULL	除外済みまたは対象。
current_version	UNSIGNED BIGINT NULL	サーバの現在のバージョン ID。
active_version	LONG BINARY NULL	サーバ上でアクティブなバージョンのリスト (エンコード済み)。
connection_info	LONG VARCHAR NULL	セミコロンで区切られた、パブリック・ドメイン接続用のホスト名とポートのペアを含む文字列。
db_path	LONG VARCHAR NOT NULL	サーバのデータベース・ファイルへのフル・パス。
private_connection_info	LONG VARCHAR NULL	セミコロンで区切られた、プライベート・ネットワーク接続用のホスト名とポートのペアを含む文字列。

基本となるシステム・テーブルでの制約

- プライマリ・キー (server_id)

参照：

- ISYSIQMPXSERVER システム・テーブル (129 ページ)

SYSIQLSPOLICY システム・ビュー

テーブル ISYSIQLSPOLICY を読みやすい形式で表示します。

システム・テーブル ISYSIQLSPOLICY には、論理サーバ・ポリシーが格納されています。

カラム名	カラム・タイプ	説明
ls_policy_Id	UNSIGNED BIGINT NOT NULL	論理サーバ・ポリシーの ID 番号。
ls_policy_name	CHAR(128) NOT NULL UNIQUE	論理サーバ・ポリシー名。

基本となるシステム・テーブルでの制約

- プライマリ・キー (ls_policy_id)
- object_id 外部キー (ISYSOBJECT)

参照：

- ISYSIQLSPOLICY システム・テーブル (128 ページ)

システム・プロシージャ

一部のシステム・プロシージャには、マルチプレックス機能をサポートするための特別な構文または使用に関する注意事項があります。

構文の記載がない場合、マルチプレックス・サーバでも、『リファレンス：ビルディング・ブロック、テーブル、およびプロシージャ』に記述されているシンプレックス・サーバと共通の構文を使用することを示します。

sp_iqcheckdb プロシージャ

現在のデータベースの妥当性を確認します。オプションで、DB 領域またはデータベースの割り付けの問題を解決します。マルチプレックス・コーディネータ・ノードの dropleaks モードは、リーク・ブロック、重複ブロック、または追加ブロックをマルチプレックス全体で検出します。

パーミッション

DBA 権限が必要です。DBA 権限を持たないユーザがこのストアド・プロシージャを実行するには、`EXECUTE` パーミッションが付与される必要があります。

sp_iqconnection プロシージャ

接続およびバージョンについての情報を表示します。この情報には、テンポラリ DB 領域を使用しているユーザ、バージョンを有効にしているユーザ、各接続が Sybase IQ 内で行っている作業、接続ステータス、データベース・バージョン・ステータスなどが含まれます。

構文

```
sp_iqconnection [ connhandle ]
```

使用法

入力パラメータ *connhandle* は、Number 接続プロパティに等しい、接続の ID 番号です。**connection_property** システム関数は、次のように接続 ID を返します。

```
SELECT connection_property ( 'Number' )
```

有効な *connhandle* の入力パラメータで呼び出されると、**sp_iqconnection** はその接続に対応する 1 つのローのみを返します。

パーミッション

DBA 権限が必要です。DBA 権限を持たないユーザがこのストアド・プロシージャを実行するには、EXECUTE パーミッションが付与される必要があります。

説明

sp_iqconnection は、アクティブな各接続に対して 1 つのローを返します。 カラムの ConnHandle、Name、Userid、LastReqTime、ReqType、CommLink、NodeAddr、および LastIdle は、接続プロパティの Number、Name、Userid、LastReqTime、ReqType、CommLink、NodeAddr、および LastIdle にそれぞれ対応しており、システム関数 **sa_conn_info** と同じ値を返します。追加のカラムは、Sybase IQ エンジンの Sybase IQ 側から接続データを返します。 ローは、ConnCreateTime の順で並べられます。

MPXServerName カラムには、次の表に示すようにマルチプレックスのノード間通信 (INC) に関連する情報が格納されています。

表 18 : MPXServerName カラム値

実行されているサーバ	MPXServerName カラムの内容
シンプレックス・サーバ	NULL (すべての接続はローカル接続またはユーザ接続です)。

実行されているサーバ	**MPXServerName** カラムの内容
マルチプレックス・コーディネータ	• ローカル／ユーザ接続の場合は NULL。 • 各 INC 接続 (オンデマンド接続または専用ハートビート接続のいずれか) のセカンダリ・ノードのサーバ名 (接続元) の値を含む。
マルチプレックス・セカンダリ	• ローカル／ユーザ接続の場合は NULL。 • コーディネータのサーバ名 (接続元) の値を含む。

Java アプリケーションでは、RemotePWD フィールドに、TDS クライアントから Sybase IQ 固有の接続プロパティを指定します。次の例は、IQ 固有の接続パラメータの指定方法を示します。**myconnection** には IQ 接続名が入ります。

```
p.put("RemotePWD",",,CON=myconnection");
```

詳細については、『SQL Anywhere サーバー – プログラミング』を参照してください。

表 19 : sp_iqconnection カラム

カラム名	説明
ConnHandle	接続の ID 番号。
Name	サーバの名前を指定します。
Userid	接続のユーザ ID。
LastReqTime	指定された接続に対する直前の要求が開始された時刻。
ReqType	最後の要求のタイプを示す文字列。
IQCmdType	Sybase IQ 側で現在実行されているコマンド (ある場合)。 コマンドの種類には、エンジンの実装レベルで定義されたコマンドが反映されます。 これらのコマンドは、トランザクション・コマンド、IQ ストア内のデータを対象とした DDL および DML コマンド、内部 IQ カーソル・コマンド、特殊な制御コマンド (**OPEN** と **CLOSE DB**、**BACKUP**、**RESTORE** など) で構成されます。
LastIQCmdTime	この接続の Sybase IQ エンジンの IQ 側で最後の IQ コマンドが開始または完了した時刻。
IQCursors	この接続の IQ ストアでオープンしているカーソルの数。

カラム名	説明
LowestIQCursorState	IQ カーソルの状態 (あれば)。接続に複数のカーソルがある場合、すべてのカーソルの中で最小のカーソル状態、つまり完了までの時間が最も長いものが表示されます。カーソル状態は内部の Sybase IQ 実装の詳細を反映するもので、将来的に変更される可能性があります。 このバージョンのカーソル状態は、NONE、INITIALIZED、PARSED、DESCRIBED、COSTED、PREPARED、EXECUTED、FETCHING、END_OF_DATA、CLOSED、COMPLETED です。 名前からもわかるように、カーソル状態は操作の最後に変更されます。 たとえば、状態 PREPARED は、カーソルが実行中であることを示します。
IQthreads	現在、接続に割り当てられている Sybase IQ スレッドの数。 割り当て済みのスレッドには、アイドルなスレッドも含まれます。このカラムから、どの接続がリソースを最も多く使用しているかを判断できます。
TxnID	接続の現在のトランザクションのトランザクション ID。この ID は、BeginTxn、CmtTxn、および PostCmtTxn メッセージによって `.iqmsg` ファイルに表示されるトランザクション ID、また、データベースが開かれたときにログ記録される Txn ID Seq と同じです。
ConnCreateTime	接続が作成された時刻。
TempTableSpaceKB	この接続が IQ テンポラリ・テーブルに格納されているデータに使用している IQ テンポラリ・ストアの領域 (KB 単位)。
TempWorkSpaceKB	この接続が、ソート、ハッシュ、テンポラリ・ビットマップなどの作業領域として使用している IQ テンポラリ・ストアの領域 (キロバイト)。ビットマップや、Sybase IQ テンポラリ・テーブルのインデックスの一部分であるその他のオブジェクトによって使用されている領域は、TempTableSpaceKB に反映されます。
IQConnID	`.iqmsg` ファイル内のすべてのメッセージの一部として表示される 10 桁の接続 ID。これは、サーバ・セッション内でユニークな、単純増加する整数です。
satoiq_count	Sybase IQ エンジンの SQL Anywhere 側から IQ 側への超過の数を表示するのに使用される内部カウンタ。これは、接続のアクティビティを確認するのに役立つことがあります。結果セットはローのバッファに返され、satoiq_count や iqtosa_count をローごとに 1 回増分することはありません。
iqtosa_count	Sybase IQ エンジンの IQ 側から SQL Anywhere 側への超過の数を表示するのに使用される内部カウンタ。これは、接続のアクティビティを確認するのに役立つことがあります。

カラム名	説明
CommLink	接続用の通信リンク。これは、Sybase IQ がサポートするネットワーク・プロトコルであり、同一マシン接続用の「ローカル」なプロトコルです。
NodeAddr	クライアント／サーバ接続のクライアント用ノード。
LastIdle	要求間のチックの数。
MPXServerName	INC 通信の場合、varchar(128) 値には、INC 通信を開始したマルチプレックス・サーバの名前が含まれます。INC 通信でない場合、NULL となります。
LSName	接続の論理サーバ名。論理サーバのコンテキストが未知または適用不可の場合、NULL となります。

例

sp_iqconnection の出力例を次に示します。

```
ConnHandle      Name    Userid                 LastReqTime   ReqType
=== =================== ======  ==========================   =======
 1  'SQL_DBC_100525210' 'DBA'   '2011-03-28  09:29:24.466'  'OPEN'


           IQCmdType    LastIQCmdTime      IQCursors  LowestIQCursorState
====================    ==============     =========  ===================
'IQUTILITYOPENCURSOR'  2011-03-28 09:29:24.0       0             'NONE'


IQthreads   TxnID          ConnCreateTime TempTableSpaceKB TempWorkSpaceKB
=========  =======  =====================  =============== ===============
        0  3352568  2011-03-28 09:29:20.0                0           0


IQconnID satoiq_count iqtosa_count CommLink NodeAdd LastIdle MPXServerName LSName
======== ============ ============ ======== ======= ======== ============  ============
      34           43          2  'local'      ''      244  (NULL)        Finance_LS
```

sp_iqdbsize プロシージャ

マルチプレックス・データベース上で実行する場合、デフォルトのパラメータは **main** です。これにより、共有 IQ ストアのサイズが返されます。

パーミッション

DBA 権限が必要です。DBA 権限を持たないユーザがこのストアド・プロシージャを実行するには、EXECUTE パーミッションが付与される必要があります。

sp_iqdbspace プロシージャ

sp_iqdbspace プロシージャがマルチプレックス構成内のセカンダリ・ノードに対して実行された場合にのみ、IQ DB 領域の Usage カラムに NA (適用外) が表示されます。

パーミッション
DBA 権限が必要です。DBA 権限を持たないユーザがこのストアド・プロシージャを実行するには、EXECUTE パーミッションが付与される必要があります。

sp_iqfile プロシージャ

sp_iqfile プロシージャがマルチプレックス構成内のセカンダリ・ノードに対して実行された場合にのみ、IQ DB 領域のファイルの Usage カラムに NA (適用外) が表示されます。

パーミッション
DBA 権限が必要です。DBA 権限を持たないユーザがこのストアド・プロシージャを実行するには、EXECUTE パーミッションが付与される必要があります。

sp_iqindexinfo プロシージャ

マルチプレックス・データベースでは、**sp_iqindexinfo** は、デフォルトで、セカンダリ・サーバ上の共有 IQ ストアに関する情報を表示します。個別のテーブルまたはインデックスが指定された場合、表示するストアは自動的に選択されます。

パーミッション
DBA 権限が必要です。DBA 権限を持たないユーザがこのストアド・プロシージャを実行するには、EXECUTE パーミッションが付与される必要があります。

sp_iqmpxincconnpoolinfo プロシージャ

コーディネータ・ノードで実行した場合、各ノードの INC 接続プール・ステータスが表示されます。セカンダリ・ノードで実行した場合、現在のノードのみの INC 接続プール・ステータスが表示されます。

構文

```
sp_iqmpxincconnpoolinfo
```

使用法
このプロシージャをコーディネータ・ノードで実行した場合にセカンダリ・ノードが応答しない場合やタイムアウトした場合は、そのノードのローは結果セット

には含まれません。これは、そのノードが実行されていない場合にこのデータへのアクセスができないためです。

パーミッション
DBA 権限が必要です。DBA 権限を持たないユーザがこのストアド・プロシージャを実行するには、`EXECUTE` パーミッションが付与される必要があります。

説明
sp_iqmpxincconnpoolinfo プロシージャは、次の出力を返します。

表 20 : sp_iqiqmpxincconnpoolinfo のカラム

カラム名	データ型	説明
server_id	unsigned int	サーバの識別子
server_name	char(128)	サーバの名前
current_pool_size	unsigned int	接続プールの現在のサイズ
idle_connection_count	unsigned int	プール内でアイドル中の接続数
connections_in_use	unsigned int	使用中の接続の数

例
次に **sp_iqmpxincconnpoolinfo** の出力例を示します。

```
server_id,server_name,current_pool_size,
idle_connection_count,connections_in_use

2,'r2_dbsrv90210',0,0,0

3,'w3_dbsrv90210',0,0,0
```

sp_iqmpxcheckdqpconfig プロシージャ

sp_iqmpxcheckdqpconfig は、現在の接続の DQP 設定をチェックする診断ツールです。DQP に失敗する場合は、**sp_iqmpxcheckdqpconfig** を実行して、DQP 設定に問題があるためにクエリ分散が失敗していないかどうかを確認します。

構文

```
sp_iqmpxcheckdqpconfig
```

パーミッション
このプロシージャの実行に、特別な権限は必要ありません。

説明

表 21 : カラムの説明

カラム名	説明
DiagMsgID	診断メッセージの一意の ID
説明	見つかった DQP 設定の問題を説明する診断メッセージ

表 22 : 診断メッセージ

DiagMsgID	説明
0	`DQP` 設定に問題はない
1	データベースがシンプレックスである
2	マルチプレックスがシングルノード設定モードで実行されている
3	ログイン・ポリシー・オプションの `dqp_enabled` が `OFF` に設定されている
4	`dqp_enabled` 接続オプションが一時的に `OFF` に設定されている
5	論理サーバ・コンテキストがメンバ・ノードを `1` つしかサポートしていない
6	論理サーバで指定されたメンバシップが現在無効のため、コーディネータが `DQP` に参加していない
7	`ALLOW_COORDINATOR_AS_MEMBER` オプションが `OFF` に設定されているため、論理サーバで指定されたメンバシップが現在無効になっており、コーディネータが `DQP` に参加していない
8	`IQ_SHARED_TEMP DB` 領域に `dbfile` が存在しない
9	`IQ_SHARED_TEMP DB` 領域内のすべての `dbfile` が読み取り専用になっている
10	`IQ_SHARED_TEMP DB` 領域が動的にオフラインになっている

例

sp_iqmpxcheckdqpconfig プロシージャからの出力例を示します。

```
diagmsgid           description
3                   Login policy option dqp_enabled is set to OFF
5                   Logical server context has only one member node
6                   Coordinator does not participate in DQP since its
                    named membership in the logical server is
                    currently ineffective
7                   Coordinator does not participate in DQP since
                    its logical membership in the logical server
                    is currently ineffective because
                    ALLOW_COORDINATOR_AS_MEMBER option in Root
                    Logical server policy set to OFF
8                   There is no dbfile in IQ_SHARED_TEMP dbspace
```

sp_iqmpxdumptlvlog プロシージャ

テーブル・バージョン・ログの内容を読みやすい形式で返します。

構文

```
sp_iqmpxdumptlvlog
[main], [asc | desc]
```

パーミッション

DBA 権限が必要です。DBA 権限を持たないユーザがこのストアド・プロシージャを実行するには、EXECUTE パーミッションが付与されている必要があります。

説明

sp_iqmpxdumptlvlog は、コーディネータが DML および DDL コマンドをセカンダリ・ノードに送信するのに用いるキューの内容を返します。

引数 **asc** または **desc** でローの順序を指定します。これらの引数には **main** の引数が必要です。デフォルトのオプションは次のとおりです。

```
'main', 'asc'.
```

例

次に **sp_iqmpxdumptlvlog** の出力を示します。

```
RowID     Contents
--------------------------------------------------------------
1         Txn CatId:196 CmtId:196 TxnId:195 Last Rec:1
          UpdateTime: 2011-08-08 15:41:43.621
2         Txn CatId:243 CmtId:243 TxnId:242 Last Rec:5
          UpdateTime: 2011-08-08 15:42:25.070
3         DDL: Type=34, CatID=0, IdxID=0,
          Object=IQ_SYSTEM_TEMP, Owner=mpx4022_w1
4         CONN: CatID=0, ConnUser=
5         SQL: ALTER DBSPACE "IQ_SYSTEM_TEMP" ADD FILE
          "w1_temp1" '/dev/raw/raw25' FILE ID 16391 PREFIX 65536
          FINISH 0 FIRST BLOCK
1         BLOCK COUNT 3276792 RESERVE 0 MULTIPLEX SERVER
          "mpx4022_w1" COMMITID 242 CREATETIME
```

```
          '2011-08-08 15:42:24.860'
6         Txn CatId:283 CmtId:283 TxnId:282 Last Rec:7
          UpdateTime: 2011-08-08 15:42:50.827
7         RFRB TxnID: 242 CmtID:243 ServerID 0 BlkmapID:
          0d00000000000000d2000a000000000002000000000000000000
          0000000000000000000008003501010000000c38000000000000
          010000000000000000000000RFID:0100050100000000001300000
          00000000001000000000001000RBID:010005010000000013000
```

sp_iqmpxfilestatus プロシージャ

コーディネータ・ノードで実行した場合は、コーディネータのファイル・ステータスと、含まれている各セカンダリ・ノード上のすべての共有 DB 領域ファイルのファイル・ステータスが表示されます。セカンダリ・ノードで実行した場合は、現在のノードのファイル・ステータスのみが表示されます。

構文

```
sp_iqmpxfilestatus
```

パーミッション

DBA 権限、MULTIPLEX ADMIN 権限、または SPACE ADMIN 権限を持っている必要があります。これらの権限を持たないユーザがこのストアド・プロシージャを実行するには、EXECUTE パーミッションが付与される必要があります。

説明

sp_iqmpxfilestatus の戻り値。

表 23 : sp_iqmpxfilestatus のカラム

カラム名	データ型	説明
ServerID	unsigned int	SYSIQMPXINFO からの、マルチプレックス・サーバの識別子
DBSpaceName	char(128)	領域の予約元の DB 領域。
FileName	char(128)	DB 領域ファイルの論理ファイル名。
FileStatus	char(2)	DB 領域ファイルのステータス。 • VALID – ファイル・パスとパーミッションは正しい。 • INVALID_PATH – パス名はアクセス不能。 • INVALID_PERM – ファイル・パーミッションが正しくない。

例

次に **sp_iqmpxfilestatus** のサンプル出力の例を示します。

```
server_id,server_name,DBSpace_name,FileName,FileStatus
1,'mpx2422_m','IQ_SYSTEM_MAIN','IQ_SYSTEM_MAIN','VALID'
1,'mpx2422_m','mpx_main1','mpx_main1','VALID'
1,'mpx2422_m','IQ_SHARED_TEMP','sharedfile_dba','VALID'
1,'mpx2422_m','IQ_SHARED_TEMP','sharedfile_dba1','VALID'
2,'mpx2422_w1','IQ_SYSTEM_MAIN','IQ_SYSTEM_MAIN','VALID'
2,'mpx2422_w1','mpx_main1','mpx_main1','VALID'
2,'mpx2422_w1','IQ_SHARED_TEMP','sharedfile_dba','VALID'
2,'mpx2422_w1','IQ_SHARED_TEMP','sharedfile_dba1','VALID'
3,'mpx2422_r1','IQ_SYSTEM_MAIN','IQ_SYSTEM_MAIN','VALID'
3,'mpx2422_r1','mpx_main1','mpx_main1','VALID'
3,'mpx2422_r1','IQ_SHARED_TEMP','sharedfile_dba','VALID'
3,'mpx2422_r1','IQ_SHARED_TEMP','sharedfile_dba1','VALID'
```

sp_iqmpxincheartbeatinfo プロシージャ

コーディネータ・ノードで実行した場合、各ノードの INC ハートビート・ステータスが表示されます。セカンダリ・ノードで実行した場合、現在のノードのみの INC ハートビート・ステータスが表示されます。

構文

```
sp_iqmpxincheartbeatinfo
```

パーミッション

DBA 権限が必要です。DBA 権限を持たないユーザがこのストアド・プロシージャを実行するには、EXECUTE パーミッションが付与される必要があります。

説明

sp_iqmpxincheartbeatinfo プロシージャは、次の出力を返します。

表 24 : sp_iqmpxincheartbeatinfo のカラム

カラム名	データ型	説明	値
server_id	unsigned int	サーバの識別子	
server_name	char(128)	サーバの名前	
last_positive_hb	TIMESTAMP	正常に実行された最後のハートビート ping の日付／時刻	DD:MM:YYYY:HH:MM:SS
time_not_responding	時刻	正常に実行された最後のハートビート ping からの経過時間	HH:MM:SS
time_until_timeout	時刻	ノードが応答していない場合に、ノードのオフラインが宣言されるまでの残り時間	

例

次に **sp_iqmpxincheartbeatinfo** の出力例を示します。

```
server_id,server_name,last_positive_hb,
time_not_responding,time_until_timeout
2,'r2_dbsrv90210',2008-11-17
15:48:42.0,00:00:00,00:00:00
3,'w3_dbsrv90210',2008-11-17
15:48:42.0,00:00:00,00:00:00
```

経過時間が 24 時間を超えると、Sybase IQ は次のような **sp_iqmpxincheartbeatinfo** 出力を返します。

```
server_id,server_name,last_positive_hb,
time_not_responding,time_until_timeout
2,'r2_mpx_cr_srv',Jan 14 2010 11:57AM,11:59PM,11:59PM
3,'w4_mpx_cr_srv',Jan 14 2010
11:57AM,11:59PM,11:59PM
(2 rows affected)
(return status = 0)
```

`time_not_responding` カラムと `time_until_timeout` カラムの 11:59PM という値は、時間が 24 時間の制限を超えたことを意味します。

sp_iqmpxinfo プロシージャ

マルチプレックス内のすべてのノードのローを返します。任意のマルチプレックス・ノードから実行できます。

注意： RESOURCE 権限を持つユーザは、DBA 権限または PERMS ADMIN 権限を持つユーザによって `EXECUTE` パーミッションが付与されないかぎり、このストアド・プロシージャを実行できません。

構文

```
sp_iqmpxinfo
```

パーミッション

DBA 権限が必要です。DBA 権限を持たないユーザがこのストアド・プロシージャを実行するには、`EXECUTE` パーミッションが付与される必要があります。

説明

sp_iqmpxinfo プロシージャは、次の出力を返します。

表 25 : sp_iqmpxinfo のカラム

カラム名	データ型	説明
server_id	unsigned int	表示する情報を特定するためのサーバ用の識別子。

カラム名	データ型	説明
server_name	char(128)	サーバの名前
connection_info	long varchar	マルチプレックス・サーバ間の TCP/IP 接続に使用される接続文字列のホスト／ポート部分を含むフォーマット済みの文字列。
db_path	long varchar	データベースへのフル・パス
role	char(16)	'coordinator' | 'writer' | 'reader'
status	char(8)	'included' | 'excluded'
mpx_mode	char(16)	'single' | 'coordinator' | 'writer' | 'reader' | 'unknown'
inc_state	char(16)	'active' | 'not responding' | 'timed out'
coordinator_failover	char(128)	フェールオーバ・サーバの名前
current_version	unsigned bigint	10 進数形式のバージョン ID
active_versions	long_varchar	10 進数形式のバージョン ID のカンマ区切りリスト
private_connection_info	long varchar	マルチプレックス・サーバ間のプライベート TCP/IP 接続に使用される接続文字列のホスト／ポート部分を含むフォーマット済みの文字列。
mipc_priv_state	char(16)	'active' – このノードへの MIPC 接続は、プライベート相互接続上でアクティブです。| 'not responding' – このノードへの MIPC 接続は、プライベート相互接続上で応答しません。
mipc_public_state	char(16)	'active' – このノードへの MIPC 接続は、パブリック相互接続上でアクティブです。| 'not responding' – このノードへの MIPC 接続は、パブリック相互接続上で応答しません。

例

sp_iqmpxinfo の出力例を次に示します。

```
server_id,server_name,connection_info,db_path,role,
status,mpx_mode,inc_state,coordinator_failover,
current_version,active_versions,private_connection_
info,mipc_priv_state,mipc_public_state

1,'my_mpx1','host=(fe80::214:4fff:fe45:be26%2):1362
0,(fd77:55d:59d9:329:214:4fff:fe45:be2
6%2):13620,10.18.41.196:13620','/system3/users
/devices/s16900269/iqmpx1/mpx1.db',
```

```
'coordinator','included','coordinator','N/A',
'my_mpx2',0,,,'active','active'

2,'IQ_mpx2','host=system3:13625',
'/system3/users/devices/s16900269
/iqmpx_2/wk0001.db','writer','included',
'writer','active','IQ_mpx20', 'not responding','active'

3,'IQ_mpx3,'host=system3:13630/system3/users/devi
ces/s16900269/iqmpx_3/mpx1.db','reader','included',
'unknown',timed out',
'IQ_mpx20','not responding',
'not responding'
```

参照：

- サーバ・ステータスのチェック (Sybase Central) (28 ページ)
- 指定済みのフェールオーバ・ノード (44 ページ)
- フェールオーバ・ノードの指定 (Sybase Central) (41 ページ)
- フェールオーバ・ノードの指定 (Interactive SQL) (24 ページ)
- サーバ・ステータスのチェック (Interactive SQL) (22 ページ)

sp_iqmpxvalidate プロシージャ

マルチプレックス設定の矛盾をチェックします。

構文

```
call dbo.sp_iqmpxvalidate( 'show_msgs' )
```

パーミッション

DBA 権限が必要です。DBA 権限を持たないユーザがこのストアド・プロシージャを実行するには、EXECUTE パーミッションが付与される必要があります。

説明

テーブル SYS.SYSIQDBFILE、他のマルチプレックス・イベントとストアド・プロシージャで複数のチェックを実行します。どのサーバでも実行可能です。呼び出し元に、次の値で重大度を示す結果を返します。

値	説明
0	エラーは検出されていません
1	予想外の動的な状態
2	致命的でない設定エラー (マルチプレックス操作の障害など)
3	致命的な設定の問題 (1 つまたは複数のサーバが起動しないおそれがあるなど)

また、対話的に呼び出した場合は、呼び出し時のパラメータが 'Y' でなかった場合を除いて、エラーが検出されたときにそのテーブルを返します。

各エラーは、その重大度を表します。エラーがなかった場合、プロシージャは `No errors detected` を返します。

sp_iqmpxversioninfo プロシージャ

このサーバの現在のバージョン情報を表示します。この情報には、サーバ・タイプ (ライタ・サーバ、クエリ・サーバ、シングルノード・モード) と同期ステータスが含まれます。

構文

```
sp_iqmpxversioninfo( )
```

パーミッション

DBA 権限が必要です。DBA 権限を持たないユーザがこのストアド・プロシージャを実行するには、`EXECUTE` パーミッションが付与される必要があります。

説明

表 26 : 返される sp_iqmpxversioninfo のカラム

カラム	データ型	説明
CatalogID	unsigned bigint	このサーバのカタログ・バージョン。
VersionID	unsigned bigint	このサーバで利用可能な最新バージョン。
OAVID	unsigned bigint	このサーバ上の最も古いバージョン。
ServerType	char(1)	サーバの種類。"C" (コーディネータ)、"W" (ライタ・サーバ)、"Q" (クエリ・サーバ)。
CatalogSync	char(1)	カタログ同期。"T" (同期)、"F" (非同期)。
WCatalogID	unsigned bigint	ライタ・サーバ上のカタログ・バージョン。
WVersionID	unsigned bigint	ライタ・サーバ上で利用可能な最新バージョン。

sp_iqsharedtempdistrib プロシージャ

共有テンポラリ領域の使用分散について、現在の状況を示します。コーディネータ・ノードで実行すると、すべてのノードの共有テンポラリ領域の分散状況が表示されます。セカンダリ・ノードで実行すると、そのノードの共有テンポラリ領域の使用状況が表示されます。

共有テンポラリ領域は、マルチプレックスを構成する各ノードがオン・デマンドで使用できるように予約されています。領域はアロケーション・ユニットを単位

として各ノードに予約されます。ノードは、動的な領域需要に基づいて、複数のアロケーション・ユニットを予約できます。アロケーション・ユニットは、必要に応じてより多くの領域を使用できるようにノードに貸し出され、不要になるとグローバル・プールに戻されます。領域の使用状況が減少し、かつ貸し出し時間が終了するか、またはサーバが停止すると、アロケーション・ユニットの有効期間は終了します。

構文

```
sp_iqsharedtempdistrib()
```

パーミッション

DBA 権限が必要です。DBA 権限を持たないユーザがこのストアド・プロシージャを実行するには、`EXECUTE` パーミッションが付与される必要があります。

説明

カラム	データ型	説明
ServerID	unsigned bigint	SYSIQMPXINFO からの、マルチプレックス・サーバのサーバ ID。
DBSpaceName	char(128)	領域の予約元の DB 領域の名前。
UnitType	char(10)	アロケーション・ユニットのタイプ。有効な値は次のとおりです。 • アクティブ – 現在予約されており、ノードによって使用されています。 • 有効期間切れ – ノード用に予約されていますが、グローバル領域プールへの返還処理中です。 • 隔離中 – ノード用に予約されていますが、ノード障害により隔離されています。
VersionID	unsigned bigint	ユニットのバージョン ID。アクティブなユニットでは、ユニットがノード用に予約された時点のバージョンです。有効期間切れのユニットでは、ユニットが有効期間切れとなった時点のバージョンです。隔離中のユニットでは、ユニットが隔離された時点のバージョンです。
NBlocks	unsigned bigint	ユニット内の未使用ブロックの数。

sp_iqspaceinfo プロシージャ

マルチプレックス・データベース上で実行する場合、デフォルトのパラメータは **main** です。これにより、共有 IQ ストアのサイズが返されます。

パラメータが指定されていない場合、テーブルなどのユーザ作成オブジェクトが 1 つもなければ、このプロシージャは結果を返しません。

パーミッション

DBA 権限が必要です。DBA 権限を持たないユーザがこのストアド・プロシージャを実行するには、EXECUTE パーミッションが付与される必要があります。

sp_iqspaceused プロシージャ

マルチプレックス・データベースで実行すると、このプロシージャは、プロシージャを実行しているサーバに適用されます。また、IQ_SHARED_TEMP で使用される領域を返します。

完全な構文と例については、『リファレンス：ビルディング・ブロック、テーブル、およびプロシージャ』を参照してください。

パーミッション

DBA 権限が必要です。DBA 権限を持たないユーザがこのストアド・プロシージャを実行するには、EXECUTE パーミッションが付与される必要があります。

sp_iqstatus プロシージャ

マルチプレックス内のセカンダリ・サーバ上では、このプロシージャは、共有 IQ ストアと IQ テンポラリ・ストアに関する情報もリストします。

sp_iqstatus でマルチプレックス・サーバのメイン・ブロック使用率が高いことが示された場合は、**sp_iqversionuse** を実行して、使用されているバージョンと、バージョンをリリースすることでリカバリできる容量を確認します。

パーミッション

DBA 権限が必要です。DBA 権限を持たないユーザがこのストアド・プロシージャを実行するには、EXECUTE パーミッションが付与される必要があります。

参照：

- sp_iqversionuse プロシージャ (154 ページ)

sp_iqtransaction プロシージャ

トランザクションとバージョンに関する情報を表示します。

構文

```
sp_iqtransaction
```

パーミッション

DBA 権限が必要です。DBA 権限を持たないユーザがこのストアド・プロシージャを実行するには、EXECUTE パーミッションが付与される必要があります。

説明

sp_iqtransaction は、Sybase IQ トランザクション・マネージャ内の各トランザクション制御ブロックのローを返します。 Name、Userid、ConnHandle の各カラムは、**Name**、**Userid**、**Number** の各接続プロパティにそれぞれ対応しています。ローは TxnID の順に並べられます。

sp_iqtransaction の出力には、実行中のトランザクションを持たない接続に対応するローは含まれません。すべての接続を表示するには、**sp_iqconnection** を使用します。

注意： **sp_iqtransaction** を使用して、他のユーザによるテーブルへの書き込みをブロックしているユーザを確認することもできますが、この場合は **sp_iqlocks** を使用することをおすすめします。

表 27 : sp_iqtransaction のカラム

カラム名	説明
Name	サーバの名前を指定します。
Userid	接続のユーザ ID。
TxnID	このトランザクション制御ブロックのトランザクション ID。トランザクション ID は、begin transaction の間に割り当てられます。これは、BeginTxn、CmtTxn、および PostCmtTxn メッセージによって .iqmsg ファイルに表示されるトランザクション ID、また、データベースが開かれたときにログ記録される Txn ID Seq と同じです。
CmtID	トランザクションがコミットしたときにトランザクション・マネージャによって割り当てられる ID。アクティブなトランザクションでは 0 です。

カラム名	説明
VersionID	シンプレックス・データベースでは、VersionID は TxnID と同一です。 マルチプレックス・コーディネータでは、VersionID はアクティブなトランザクションの TxnID、コミット済みのトランザクションの CmtID と同一です。 マルチプレックスのセカンダリ・サーバでは、VersionID は、マルチプレックス・コーディネータでデータベース・バージョンを作成したトランザクションの CmtID です。これは、マルチプレックス・データベース内のすべてのノードに対してデータベース・バージョンを一意に識別するために、Sybase IQ のメモリ内カタログと IQ トランザクション・マネージャで内部的に使用されます。
State	トランザクション制御ブロックの状態。 この変数は内部の Sybase IQ 実装の詳細を反映するもので、将来的に変更される可能性があります。 本稿の執筆時点では、トランザクションの状態は NONE、ACTIVE、ROLLING_BACK、ROLLED_BACK、COMMITTING、COMMITTED、APPLIED です。
ConnHandle	接続の ID 番号。
IQConnID	`.iqmsg` ファイル内のすべてのメッセージの一部として表示される 10 桁の接続 ID。これは、サーバ・セッション内でユニークな、単純増加する整数です。
MainTableKBCr	このトランザクションによって作成された IQ ストアの領域 (KB 単位)。
MainTableKBDr	このトランザクションによって削除済みの IQ ストアの領域のうち、他のデータベース・バージョン、またはこのトランザクションの他のセーブポイントでこの領域が表示されているためにストア上のディスクに残っている領域の容量 (KB 単位)。
TempTableKBCr	このトランザクションが IQ テンポラリ・テーブルのデータの格納用に作成した IQ テンポラリ・ストアの領域 (KB 単位)。
TempTableKBDr	このトランザクションによって削除された IQ テンポラリ・テーブルの領域のうち、IQ カーソルで表示されているため、またはこのトランザクションの他のセーブポイントが所有しているためにテンポラリ IQ ストアのディスク上に残っている領域の容量 (KB 単位)。

カラム名	説明
TempWorkSpaceKB	ステータスが ACTIVE であるトランザクションでは、このトランザクションが使用中のワークスペースのスナップショットです (ソート、ハッシュ、テンポラリ・ビットマップなど)。この数字は、**sp_iqtransaction** を実行するタイミングによって変わります。たとえば、クエリ・エンジンがテンポラリ・キャッシュに 60MB を作成した場合、クエリ処理が継続中でもその大部分をすぐに解放することがあります。したがって、クエリが完了した後に **sp_iqtransaction** を実行すると、このカラムの数字がずっと小さくなっていることがあります。トランザクションがアクティブでなくなると、このカラムはゼロになります。 ACTIVE なトランザクションでは、このカラムは **sp_iqconnection** の TempWorkSpaceKB カラムと同一です。
TxnCreateTime	トランザクションの開始時刻。すべての Sybase IQ トランザクションは、アクティブな接続が確立されるか、または前のトランザクションがコミットまたはロールバックしたときに暗黙的に開始されます。
CursorCount	このトランザクション制御ブロックを参照している、オープンしている Sybase IQ カーソルの数。トランザクションが ACTIVE である場合、トランザクション内で作成されたオープンしているカーソルの数を示します。トランザクションのステータスが COMMITTED の場合は、このトランザクション制御ブロックが所有するデータベース・バージョンを参照する HOLD カーソルの数を示します。
SpCount	トランザクション制御ブロック内に存在する、セーブポイント構造の数。セーブポイントは、暗示的に作成および解放されます。したがって、この番号はトランザクション内でセーブポイントを作成したユーザの数を示すものではありません。
SpNumber	トランザクションの、アクティブなセーブポイントの数です。これは実装の詳細を反映しているため、セーブポイントを作成したユーザの数は反映されていない場合があります。
MPXServerName	この値は、アクティブなトランザクションが、ノード間通信 (INC) 接続からのトランザクションであるかどうかを示します。INC 接続からのトランザクションの場合、この値は、そのトランザクションが開始されたマルチプレックス・サーバの名前となります。INC 接続からのトランザクションでない場合、NULL となります。トランザクションがアクティブでない場合は、常に NULL となります。
GlobalTxnID	この値は、現在のトランザクションに関連付けられているグローバル・トランザクション ID を示します。関連付けられているグローバル・トランザクションがない場合、ゼロとなります。

例

sp_iqtransaction の出力の例を次に示します。

```
Name,Userid,TxnID,CmtID,VersionID,State,ConnHandle,IQConnID,
MainTableKBCr,MainTableKBDr,TempTableKBCr,TempTableKBDr,
TempWorkSpaceKB,TxnCreateTime,CursorCount,SpCount,SpNumber,
MPXServerName,GlobalTxnID

'IQ_MPX_SERVER_H','dbo',49878,49881,49881,'COMMITTED',9,23198,152,
152,0,0,0,'2008-11-18 13:15:00.015',0,0,0,,0

'IQ_MPX_SERVER_H','dbo',49880,49882,49882,'COMMITTED',13,23207,152,
152,0,0,0,'2008-11-18 13:15:00.016',0,0,0,,0

'IQ_MPX_SERVER_H','dbo',49884,49885,49885,'COMMITTED',11,23202,152,
152,0,0,0,'2008-11-18 13:15:00.038',0,0,0,,0

'IQ_MPX_SERVER_H','dbo',49909,49910,49910,'COMMITTED',15,23212,152,
152,0,0,0,'2008-11-18 13:16:00.016',0,0,0,,0

'SQL_DBC_49450e8','DBA',50024,0,50024,'ACTIVE',17,23267,0,0,0,
0,0,'2008-11-18 13:28:23.358',0,2,261,,0

'Sybase Central 1','DBA',50545,0,50545,'ACTIVE',44,23443,0,0,0,
0,0,'2008-11-18 14:04:53.256',0,1,0,,0
```

sp_iqversionuse プロシージャ

使用されているバージョンと、使用しているユーザを表示します。

このプロシージャは、あるバージョンの各ユーザのローを生成します。まずコーディネータ上で **sp_iqversionuse** を実行し、解放するバージョンと、そのバージョンを使用しなくなった場合に解放される領域の大きさ (KB 単位) を調べます。接続 ID は、コーディネータに接続しているユーザの IQConn カラムに表示されます。セカンダリ・サーバで使用されているバージョンは、セカンダリ・サーバ名として接続 ID 0 とともに表示されます。

マルチプレックス・セカンダリ・サーバ上で **sp_iqversionuse** を実行して、セカンダリ・サーバへの個別の接続を調べます。他のサーバのユーザはセカンダリ・サーバ上で表示されません。

パーミッション

DBA 権限が必要です。DBA 権限を持たないユーザがこのストアド・プロシージャを実行するには、`EXECUTE` パーミッションが付与される必要があります。

参照：

- sp_iqstatus プロシージャ (150 ページ)

例

sp_iqversionuse の出力は、コーディネータ・サーバとセカンダリ・サーバで異なります。

この例では、最も古いバージョン 42648 がコーディネータ (*mpxw*) 上で接続 108 により使用されています。接続 108 のトランザクションがコミットまたはロールバックされると、7.9MB の領域が解放されます。コーディネータからの出力によれば、バージョン 42686 はセカンダリ・サーバ (*mpxq*) により使用されています。セカンダリ・サーバ出力では、実際の接続は接続 31 です。バージョン 42686 を解放して返される実際の領域の大きさは、最初に 42648 を解放するかどうかによって異なります。

コーディネータ上でのバージョン 42715 および 42728 に対する WasReported は 0 です。これは、まだレプリケートされていない新しいバージョンが存在するためです。バージョン 42728 はセカンダリ・サーバ出力に含まれていないため、セカンダリ・サーバによってまだ使用されていません。

sp_iqversionuse がコーディネータ *mpxw* 上で実行された場合、次の出力が返されます。

```
call dbo.sp_iqversionuse
```

VersionID	Server	IQConn	WasReported	MinKBRelease	MaxKBRelease
42648	'mpxw'	108	1	7920	7920
42686	'mpxq'	0	1	7920	304
42702	'mpxq'	0	1	0	688
42715	'mpxq'	0	0	0	688
42728	'mpxq'	0	0	0	688

sp_iqversionuse がセカンダリ・サーバ (*mpxq*) 上で実行された場合、次の出力が返されます。

```
call dbo.sp_iqversionuse
```

VersionID	Server	IQConn	WasReported	MinKBRelease	MaxKBRelease
42686	'mpxq'	31	1	0	0
42715	'mpxq'	00	1	0	0

起動ユーティリティとデータベース管理ユーティリティ

一部のコマンド・ライン・ユーティリティには、マルチプレックスの構文または制約があります。

マルチプレックス・サーバとシンプレックス・サーバに共通する構文については、『ユーティリティ・ガイド』を参照してください。

バックアップ・ユーティリティ (dbbackup)

dbbackup ユーティリティは、データベース名を 70 文字にトランケートし、トランケートされた名前でターゲット・ファイルを作成します。Sybase IQ は、セカンダリ・サーバを同期するときに **dbbackup** を使用します。 この **dbbackup** の制限のため、データベース名は 70 文字未満にする必要があります。

サーバ起動ユーティリティ (start_iq)

サーバ起動をカスタマイズするには、コマンド・ラインで start_iq を実行します。

Sybase IQ マルチプレックス・サーバを起動時に設定するには、**start_iq** コマンドでサーバ起動スイッチを使用します。

表 28 : サーバ起動スイッチ

起動スイッチ	値	説明
-iqmpx_failover	1	マルチプレックス・コーディネータ・フェールオーバを初期化し、指定されたフェールオーバ・セカンダリ・ノードを新しいコーディネータとして確立します。このオプションを使用してコーディネータを起動しても影響はありません。
-iqmpx_ov	1	現在のノードのマルチプレックス設定の上書きを実行します。ノードの場所またはその他のプロパティが変更された場合、起動時にノードのプロパティを変更するために使用されます。
-iqmpx_sn	1	現在のマルチプレックス・ノードをシングルノード・モードで実行します。シングルノード・モードは、マルチプレックス設定の問題を解決する場合にのみ使用してください。マルチプレックス内の他のすべてのノードを停止する必要があります。シングルノード・モードはコーディネータのみで使用することをおすすめします。

起動スイッチ	値	説明
-iqmpx_reclaim-writerfreelist	サーバ名	このオプションは、コーディネータ・ノードを再起動している間にのみ適用されます。コーディネータは、サーバ名で識別されているライタ・ノードのフリー・リストを強制的に要求します。このスイッチは、ライタに障害が発生し、再起動できない場合にのみ使用してください。
-iqmsgnum*num*	0 〜 64 (両端を含む)	サーバで維持する古いメッセージ・ログのメッセージ・ログ・アーカイブ数を指定します。デフォルトは 0 で、メッセージはメインのメッセージ・ログ・ファイル内でラッピングされます。**-iqmsgsz** サーバ・プロパティまたは **IQMsgMaxSize** サーバ・プロパティが 0 以外の場合にのみ有効になります。**IQMsgNumFiles** サーバ・プロパティは -iqmsgnum に相当し、**-iqmsgnum** の値より優先されます。値が設定されていない場合、デフォルトの最小プール・サイズは `MIN (MAX (4, number of cores/4) , mipcmaxt (if set)).` です。
-iqmsgsz*size*	0 〜 2047 (両端を含む) の整数値 (MB 単位)	メッセージ・ログの最大サイズを制限します。デフォルトは 0 で、メッセージ・ファイルのサイズの制限がないことを示します。
-mipcmint*size*	0 〜 256 (両端を含む) の整数値	MIPC 要求処理用の共有スレッド・プールで許容されるスレッドの最小数を指定します。MIPC サーバ接続が新規作成されるたびに、2 つのスレッドがプールに追加されます。**-mipcmint** の値は、デフォルトでは 0 で、**-mipcmaxt** 値を超えることはできません。Sybase 製品の保守契約を結んでいるサポート・センタから指示があった場合にのみ、この値を設定してください。値が設定されていない場合、デフォルトの最小プール・サイズは `MIN (MAX (4, number of cores/4),  mipcmaxt (if set))` です。

起動スイッチ	値	説明
-mipcmaxt*size*	0 ～ 256 (両端を含む) の整数値	MIPC 要求処理用の共有スレッド・プールで許容されるスレッドの最大数を指定します。MIPC サーバ接続が新規作成されるたびに、2 つのスレッドがプールに追加されます。**-mipcmaxt** の値は、デフォルトでは 0 で、**-mipcmint** 値を超える必要があります。Sybase 製品の保守契約を結んでいるサポート・センタから指示があった場合にのみ、この値を設定してください。値が設定されていない場合、デフォルトの最大プール・サイズは `MAX (number of cores, mipcmint)` です。

その他のスイッチについては、Sybase IQ『ユーティリティ・ガイド』の「start_iq データベース・サーバ起動ユーティリティ」>「start_iq サーバ・オプション」を参照してください。

注意： **-iqmc** スイッチと **-iqtc** スイッチを指定すると、マルチプレックスの各ノードで異なるキャッシュ・サイズを使用できますが、これが悪影響を及ぼす可能性があります。たとえば、ワーカ・ノードのキャッシュがリーダよりはるかに少なく設定されている場合、リーダのハッシュ・ジョインは、並行処理を禁止するページング・モードで動作します。

参照：

- ジョイン・インデックス内のテーブル (51 ページ)

索引

記号

-n スイッチ 38
[サーバ] タブ 28
[収集レートの設定] ダイアログ 88

A

ALTER DBSPACE 文
　制限 55
　例 17
ALTER LOGICAL SERVER 文
　構文 107
ALTER LOGIN POLICY 文
　構文 108
ALTER LS POLICY 文
　構文 113
ALTER MULTIPLEX RENAME statement 114
ALTER MULTIPLEX RENAME 文 114
ALTER MULTIPLEX SERVER 文 114
　自動停止 40

C

Command not replayed 56
COMMENT ON LOGICAL SERVER 文 116
CPU 使用率統計 91
CREATE DBSPACE 文 17
　制限 55
CREATE LOGICAL SERVER 文 117
CREATE MULTIPLEX SERVER 文 118
CREATE TEXT CONFIGURATION 69, 71, 72, 75

D

DB ファイル
　Error opening DBFILE メッセージ 57
　参照できないパス 57
　読み取り専用 57
DB 領域
　マルチプレックス内の更新 55
　記憶領域デバイス 13
　使用状況統計 97
　要件 13
dbbackup 156
DML コマンド
　マルチプレックスでの実行 51
DML 操作
　ジョイン・インデックス内のテーブル 51
DQP_ENABLED オプション 122
DROP DBSPACE 文
　制限 55
DROP LOGICAL SERVER 文 120
DROP MULTIPLEX SERVER 文 121

I

IQ Agent
　UNIX 上でのリモート・ファイル参照の無効化 33
　ホスト・エイリアス 34
　ホスト名 33
　実行 29
　設定 29
　目的 29
IQ Agent Windows サービス
　アンインストール 32
　インストール 33
IQ ストア
　バックアップ 99, 100
IQ メイン・ストア
　共有 14
IQ_SHARED_TEMP
　ロー・デバイスの要件 13
　更新 60
　領域の追加 61
IQ_SYSTEM_MAIN
　ファイル・パスの変更 57
　ファイルの追加 57, 58
　ロー・デバイスの要件 13
　領域の追加 58
　領域の不足 58

IQ_SYSTEM_MAIN への領域の追加 58
IQ_SYSTEM_TEMP
セカンダリ・サーバ 22, 39
更新 56
IQAGENTNOBROWSE 環境変数 33
IQLOGDIR15 環境変数 29
iqmsgnum 起動スイッチ 157
iqmsgsz 起動スイッチ 157
ISYSIQLOGICALSERVER システム・テーブル 128
ISYSIQLOGINPOLICYLSINFO システム・テーブル 128

J

Java JAR
マルチプレックス 14
Java クラス
マルチプレックス 14

L

LOCK TABLE 文 54

M

MIPC 10, 81
MPX_AUTOEXCLUDE_TIMEOUT オプション 123
MPX_HEARTBEAT_FREQUENCY オプション 123
MPX_IDLE_CONNECTION_TIMEOUT オプション 124
MPX_MAX_CONNECTION_POOL_SIZE オプション 124
MPX_MAX_UNUSED_POOL_SIZE オプション 125
MPX_WORK_UNIT_TIMEOUT オプション 126
MPXServerName カラム 135
multiplex
renaming 114
MULTIPLEX ADMIN 権限
説明 42

R

RESOURCE 権限 22

S

sp_iqcheckdb システム・プロシージャ 134
sp_iqconnection システム・プロシージャ 135
sp_iqdbsize システム・プロシージャ 138
sp_iqdbspac システム・プロシージャ 139
sp_iqdbspace プロシージャ 55
sp_iqemptyfile プロシージャ 57
sp_iqfile システム・プロシージャ 55, 139
sp_iqindexinfo システム・プロシージャ 139
sp_iqmpxdumptlvlog ストアド・プロシージャ 142
sp_iqmpxfilestatus システム・プロシージャ 143
sp_iqmpxincconnpoolinfo ストアド・プロシージャ 139
sp_iqmpxincheartbeatinfo ストアド・プロシージャ 144
sp_iqmpxinfo ストアド・プロシージャ 145
sp_iqmpxvalidate システム・プロシージャ 22, 39, 147
sp_iqmpxversioninfo システム・プロシージャ 148
sp_iqsharedtempdistrib システム・プロシージャ 148
sp_iqspaceinfo システム・プロシージャ 150
sp_iqspaceused システム・プロシージャ 150
sp_iqstatus システム・プロシージャ 150
sp_iqtransaction システム・プロシージャ 151
sp_iqversionuse システム・プロシージャ 154
SQL 文 107
start_iq パラメータ
マルチプレックス 21
Sybase Central
トポロジ・ビュー 85
起動 35
Sybase Control Center 41
論理サーバ 73

T

TLV ログ 4
表示内容 142

U

UNIX
  共有ディスクへのアクセス 7
UNIX 上での共有ディスクへのアクセス 7

W

Windows
  共有ディスクへのアクセス 8
Windows 上での共有ディスクへのアクセス 8

あ

アプリケーション
  マルチプレックス用の起動パラメータ 21
アロケーション・ユニット 148

お

オーバーヘッド
  ロード 6
オプション
  DQP_ENABLED 122
  MPX_AUTOEXCLUDE_TIMEOUT 123
  MPX_HEARTBEAT_FREQUENCY 123
  MPX_IDLE_CONNECTION_TIMEOUT 124
  MPX_MAX_CONNECTION_POOL_SIZE 124
  MPX_MAX_UNUSED_POOL_SIZE 125
  MPX_WORK_UNIT_TIMEOUT 126
  ログイン・ポリシー 111, 132
  論理サーバのログイン・ポリシー 132

か

カタログ・ストア
  バックアップ 99, 100
カタログ・フォーマット番号 127

き

キャッシュ統計 92

く

クエリ
  パフォーマンス 82
クエリ部分 82

こ

コーディネータ
  コーディネータの機能 5
  シンプレックスからマルチプレックスへの変換 16
  起動不可能 58
  共有テンポラリ・ストアの使用状況 148
  作成 16, 17
  置き換え 44
  読み取り専用バックアップ 105
  利点 6
  領域の不足 58
コーディネータの置き換え 44

さ

サーバ
  トポロジ・ビューでのステータス 88
  トポロジ・ビューでの状態 88
  パフォーマンスのモニタリング 88
  フェールオーバ 44
  プロパティ 36
  マルチプレックス 27
  マルチプレックスの変更 114
  マルチプレックスの有効化 16, 17
  リストア 100, 101, 103
  起動 35
  起動パラメータ 21
  削除 35
  除外 35, 40, 41
  停止 35
  包含 35, 41
  名前の変更 23, 40
  役割の変更 40
  論理サーバ 116
  論理サーバの作成 117
  論理サーバの削除 120
サーバの作成
  パーミッション 42
サーバの削除 35
  パーミッション 42

サーバの除外 35, 40
SQL 24
Sybase Central 41
サーバの包含 35
SQL 24
Sybase Central 41
サーバ起動スイッチ
iqmpx_failover 156
iqmpx_ov 156
iqmpx_reclaimwriterfreelist 156
iqmpx_sn 156
サーバ設定
パーミッション 42

し

シェル・スクリプト
管理 36
システム・テーブル
ISYSIQLOGICALSERVER 128, 129
ISYSIQLOGINPOLICYLSINFO 128, 130
ISYSIQLSLOGINPOLICIES 131
ISYSIQLSLOGINPOLICYOPTION 128, 132
ISYSIQLSMEMBER 128, 130
ISYSIQLSMEMBERS 131
ISYSIQLSPOLICY 128, 134
ISYSIQLSPOLICYOPTION 129, 132
ISYSIQMPXSERVER 129
SYSIQINFO 126
システム・ビュー
SYSIQLOGICALSERVER 129
SYSIQLOGINPOLICYLSINFO 130
SYSIQLSLOGINPOLICIES 131
SYSIQLSLOGINPOLICYOPTION 132
SYSIQLSMEMBER 130
SYSIQLSMEMBERS 131
SYSIQLSPOLICY 134
SYSIQLSPOLICYOPTION 132
SYSIQMPXSERVER 133
システム・プロシージャ
sp_iqconnection 135
sp_iqdbspace 55, 139
sp_iqfile 55, 139
sp_iqmpxfilestatus 143
sp_iqmpxincconnpoolinfo 139
sp_iqmpxincheartbeatinfo 144
sp_iqmpxinfo 145
sp_iqtransaction 151
ジョイン・インデックス 51
シンプレックスからマルチプレックスへの変換 17
シンボリック・リンク 99

す

ステータス
チェック 28
モニタリング 85
ステータスのモニタリング 85
ストア
サポートされるデバイス 13
ストア I/O 統計 96
ストアド・プロシージャ
sp_iqmpxdumptlvlog 142
sp_iqmpxincconnpoolinfo 139
sp_iqmpxincheartbeatinfo 144
sp_iqmpxinfo 145
フォーマット番号 127
実行パーミッションの付与 43
スレッド統計 94

せ

セカンダリ・サーバ
共有テンポラリ・ストアの使用状況 148
作成 16, 17
セカンダリ・ノード
停止メッセージ 57, 58

そ

ソフト・リンク 7

て

データベース
システム・テーブル内のブロック・サイズ 127
ファイル・パスの変更 40
ファイル・フォーマット 127
マルチプレックスへの変換 13, 15, 18
マルチプレックス機能の有効化 16, 17
作成時刻 127
名前の長さ制限 156

データベース・オプション
　マルチプレックス 122
データベース・ファイルのパスの変更 40
テーブル
　ジョイン・インデックス内 51
テーブル・バージョン・ログ 4
テキスト設定オブジェクト
　作成 69, 71, 72, 75

と

トポロジ・ビュー 85
　更新 86
トランザクション・ステータス
　モニタリング 88
トランザクションのパフォーマンス 6
トランザクション統計 96

ね

ネットワーク
　冗長 10
ネットワーク統計 98

の

ノード
　コーディネータの置き換え 44
ノード間接続 8

は

バージョン・ステータス
　システム・テーブル 129
ハートビート接続 9
パーミッション
　MULTIPLEX ADMIN 権限 42
パーミッション拒否エラー 55
パス
　絶対パス 7
バックアップ・ユーティリティ 156
パフォーマンス
　モニタリング 41
　動的モニタ 88
パフォーマンス・モニタ
　2 次元棒グラフ 90
　3 次元棒グラフ 90
　サーバ・レベル 88
　チャート・ビューの切り替え 90
　チャートのカスタマイズ 90
　チャートの印刷 90
　チャートの保存 90
　マルチプレックス・レベル 88
　モニタリング対象の統計 91
　時系列ビュー 90
　統計のカスタマイズ 88, 89
パフォーマンスのモニタリング 41
パラレル・ロード 6

ふ

ファイル
　IQ_SYSTEM_MAIN に追加 58
　削除 55
ファイル・パス
　IQ_SYSTEM_MAIN での変更 57
　変更 40
ファイル・フォーマット 127
プール済み接続 9
フェールオーバ 44, 46
　Sybase Central の使用 46
　Sybase Control Center の使用 48
フェールオーバ・ノード
　指定 41
ブロック・サイズ
　システム・テーブル 127

ほ

ポート
　変更 40
ホスト
　変更 40
ポリシー
　論理サーバ 134

ま

マルチプレックス
　DB 領域の更新 55
　DDL コマンド 52
　DML コマンド 51

SQL 文 107
start_iq パラメータ 21
Sybase Central の起動 35
Windows の共有ディスク 8
グラフィカル・ビュー 85
コーディネータ 5
コーディネータの置き換え 44
サーバのポートの変更 40
サーバのホストの変更 40
サーバの除外 40
サーバの包含 40
サーバの役割の変更 40
システム・テーブル 126
システム・プロシージャ 135
セカンダリ・ノード 5
データベース・オプション 122
データベース・ファイルのパスの変更 40
トポロジ・ビュー 85
トポロジ・ビューの更新 86
トポロジの印刷 86
トポロジの表示 85
ノード間通信 8
バージョンの混在 14
バージョン情報の表示 148
ハートビート接続 9
パフォーマンス・モニタ 88
プール済み接続 9
異機種間マルチプレックス 14
静的設定 4
設定のチェック 147
前提条件 13
動的衝突 63
動的設定 4
名前記憶領域 114
領域の追加 58
論理サーバのメンバシップ 130, 131
マルチプレックス・サーバ
起動 27
同期 37
名前の変更 23, 40
マルチプレックス・サーバの除外 40
マルチプレックス・サーバの設定の変更
パーミッション 42
マルチプレックス・サーバの包含 40
マルチプレックス・サーバの名前変更 23, 40
マルチプレックス・プロセス間通信 10
マルチプレックス・メンバシップ・プロパティ
システム・テーブル 129
マルチプレックス・ログイン・ポリシー
システム・テーブル 128
マルチプレックスのリストア 100, 101, 103
マルチプレックスの同期 37
マルチプレックスの名前の変更
パーミッション 42
マルチプレックス環境不正エラー 22, 39
マルチプレックス論理サーバ
システム・テーブル 128, 129

め

メッセージ・ログ
iqmsgnum サーバ・オプション 157
iqmsgsz サーバ・オプション 157
アーカイブの数の指定 157
サイズの指定 157
メモリ使用状況統計 92
メンバシップ
論理サーバ 130, 131

も

モニタリング
トランザクション・ステータス 88
役割 85

ゆ

ユーザ接続 26

り

リーダ・ノード 82
リストア
読み取り専用バックアップ 105
リモート・ファイル参照
UNIX 上での無効化 33

る

ルート論理サーバ・ポリシー 113

ろ

ロー・デバイス
アクセス 7
要件 13
ロード
パラレル 6
ログ
テーブル・バージョン 4
ログイン・ポリシー
システム・テーブル 128
変更 108, 111
論理サーバの割り当て 130, 131

わ

ワーカ・ノード 82