

max_statement_count データベース・オプションの使用

不要になった準備文が明示的に削除されない場合、その準備文を使用するアプリケーションはエラー "Resource governor for 'prepared statements' exceeded ('準備文' のリソース・ガバナーが制限を超えています)" を受信する可能性があります。max_statement_count データベース・オプションは、接続ごとに使用される準備文の数の制限を DBA に許可するリソース・ガバナーです。オペレーションが接続の制限を超えた場合は、リソースのガバナーを超過していることを示すエラーが生成されません。

接続がストアド・プロシージャを実行する場合、そのプロシージャはプロシージャ所有者のパーミッションのもとで実行されます。ただし、プロシージャによって使用されるリソースは、現在の接続に割り当てられます。

データベース・サーバは、接続によって作成される準備文ごとにデータ構造を管理します。これらの構造は、準備文が不要になっていることをアプリケーションがサーバに知らせた場合か、接続が切断された場合にのみ解放されます。接続のステートメント・カウントを減少させるには、DROP STATEMENT 要求に相当するコマンドを実行する必要があります。以下の表に、Adaptive Server Anywhere によってサポートされる API に対して実行できるコマンドを示します。

インターフェース	ステートメント
embedded SQL	DROP STATEMENT
ODBC	SQLFreeStmt(hstmt, SQL_DROP) または SQLFreeHandle(SQL_HANDLE_STMT, hstmt)
Java	resultSet.Close() または Statement.Close()
ADO	RecordSet.Close()
ADO.NET	ASADeveloper.Close() または AsaDeveloper.Dispose()

注：Java および .NET では、明示的にステートメントをドロップすることをお勧めします。言語ルーチンはサーバ呼び出しを発行してステートメント・リソースの割り当てを解除することは行わないため、ガーベジ・コレクションに依存してこのクリーンアップを実行しないでください。また、いつガーベジ・コレクション・ルーチンが実行されるのかということについても何も保証はありません。

任意の接続について任意の時点でサーバがデフォルトの数よりも多くの準備文をサポートする必要がある場合は、max_statement_count 設定をより大きな値に設定してください。ただし、アクティブな準備文の数が多くなるとサーバ・メモリがさらに消費されることに注意してください。max_statement_count オプションを 0 (ゼロ) に設定すると準備文のリソース・ガバナーを完全に無効にすることができますが、この操作はお勧めしません。そのようにすると、正しく準備文を解放しないアプリケーションの場合、サーバはメモリ不足状態で終了しやすくなります。