

ASA バージョン 5.5 からバージョン 7.0 へのマイグレート

ここでは、バージョン 5.5 から 7 へマイグレートするときに考慮すべき問題および互換性について説明します。

概要

ASA バージョン 5.5 から ASA バージョン 7 に直接マイグレートする場合の問題点について、質問を受けることがよくあります。ASA バージョン 5.5 から ASA バージョン 7 へのアップグレードは、他のアップグレードの場合と何も変わりません。ただし、対処すべき互換性の問題があり、新機能についてすぐに利用するかどうかを検討する必要があります。

この文書にはすべての相違点の詳細な説明は記載されていないため、詳細についてはバージョン 6 とバージョン 7 の『リリース・ノート』を参照してください。

一般的なアップグレードのガイドライン

アプリケーションをアップグレードする前に、注意する点がいくつかあります。また、これらの注意点は、他のソフトウェアの場合と同様に Adaptive Server Anywhere のアップグレードにも適用されます。

動作変更のチェック - アップグレードの前に、記載されているバージョン 7 の動作変更がどれもアプリケーションに影響しないことを確認する必要があります。

テスト - 運用中のアプリケーションをアップグレードする前に、バージョン 7 の環境で十分にアプリケーションをテストする必要があります。

バックアップ - アップグレードの前に、既存のソフトウェアとデータをバックアップする必要があります。また、データベースのアップグレード中にはリカバリは行われないため、アップグレード後にもバックアップを作成してリカバリ性を確実に維持する必要があります。

アップグレード手順のテスト - 多数のエンド・ユーザをアップグレードする場合は、事前にアップグレード手順を慎重にテストします。

SQL Anywhere は非常に多くの異なる設定で使用されるため、すべての場合についてアップグレードのガイドラインを保証することはできません。

互換性の問題

互換性の問題とは、Adaptive Server Anywhere でのアプリケーションの動作方法に直接影響する可能性がある変更のことを示します。バージョン 5.5 からバージョン 7 にマイグレートする場合、これらの問題を次のように大きく分類できます。

- データベース・ファイルの変更点
- ネーミングの変更点
- コネクティビティの変更点

- SQL ダイレクトの変更点
- ツールの変更点
- SQL Remote の変更点
- サポートされるオペレーティング・システム

次に、これらの各分野について個別に説明します。

データベース・ファイルの変更点

バージョン 7 のデータベース・サーバを使用するためにデータベースをアップグレードする必要はありません。旧バージョンの場合と同様に、バージョン 7 のデータベース・サーバも以前のデータベース・ファイル・フォーマットをサポートしているので、既存のアプリケーションの操作を継続するためにデータベースをアップグレードする必要はありません。データベース・オプション、システム・テーブルの変更、またはデータベース記憶領域の変更に関係しない新しい機能 (SMP サポートや動的キャッシュ・サイズ変更など) を利用できます。

新しいデータベース・オプションやシステム・テーブルの変更点に関するバージョン 7 の機能を利用する場合は、データベースをアップグレードする必要があります。たとえば、スケジューリングとイベント処理、またはデータベースで Java を利用するには、システム・テーブルを変更する必要があります。Upgrade ユーティリティまたは ALTER DATABASE 文を使用すると、データベースをアップグレードできます。

一部の機能は、データベースの変更済みファイル・フォーマットに依存しています。これらの機能を使用するには、データベースをアンロードしてから再ロードする必要があります。次にその機能を示します。

- 可変ハッシュ・サイズ・インデックス
- 独立したプライマリ・キー・インデックスと外部キー・インデックス
- blob 用に再編成された記憶領域
- GLOBAL DEFAULT AUTOINCREMENT カラム

新機能およびそれらの新機能に必要なアップグレードのタイプについては、バージョン 6 とバージョン 7 の『リリース・ノート』を参照してください。

ネーミングの変更点

バージョン 5.5 とバージョン 7 では、デフォルトのインストール・ディレクトリと多数のファイル名が異なっています。具体的な変更点については、数が多すぎるためこの文書には記載していません。変更されていない名前は、通常は DBBackup.exe や DBLog.exe などのメイン・データベース・ツールのプログラムだけです。これらの名前は、バッチ・ファイルが引き続き機能することを保証するために残してあります。一般に、これはユーザが上記の実行プログラムの複数のバージョンをマシンにインストールしており、システム検索パスによってそれらを検出する場合にのみ問題となります。その場合は、オペレーティング・システムが誤ったバージョンを見つけて起動する可能性があります。このような状況では、実行プログラムの起動時に明示的にロケーションを指定することをおすすめします。

コネクティビティの変更点

クライアント実行プログラムがない

バージョン 5.5 からバージョン 7 に移行するときの最も大きな問題は、ネットワーク・サポートにおける変更点です。バージョン 6 では、独立したクライアント実行プログラムが削除され、ネットワーク機能がプログラミング・インタフェース (つまり ODBC) に統合されました。この変更により、ネットワーク通信の設定が簡略化されています。Adaptive Server Anywhere のクライアント/サーバ使用は、旧リリースの場合よりもはるかに単一マシン使用に類似しています。

この変更では、クライアント/サーバ接続文字列と DBClient.exe コマンド・ラインをすべて確認することが必要になります。ネットワーク制御パラメータを指定した場合は、それらのパラメータが引き続き必要かどうかを確認し、引き続き必要な場合は、CommLinks パラメータを使用してアプリケーションの接続文字列にそれらのパラメータを統合する必要があります。

Sybase Central、Interactive SQL、コマンド・ライン管理ユーティリティなどの Embedded SQL アプリケーションでは、ODBC データ・ソース記述を使用してデータベースに接続できます。これにより、接続情報の一般的な格納方法として ODBC データ・ソースを使用することができます。

利用可能な接続パラメータの詳細については、『ASA ユーザーズ・ガイド』の第 2 章「データベースへの接続」と第 3 章「クライアント/サーバ通信」を参照してください。

プロトコルの変更点

Adaptive Server Anywhere によってサポートされるプロトコルに変更がいくつか加えられています。

SPX のサポート - SPX プロトコルを使用してデータベースに接続できます。この機能は、IPX/SPX をプライマリ・ネットワーク・プロトコルとして使用する Novell NetWare 環境で特に役立ちます。IPX より SPX を使用することをおすすめします。

TCP/IP の変更点 - TCP/IP でのクライアント/サーバ接続の確立が簡略化されています。デフォルトのポート番号 (2638) 以外のポートでサーバが実行されている場合であっても、クライアントがポート番号を指定する必要はありません。データベース・サーバの起動時にデフォルトのポート番号が使用中の場合は、サーバがオペレーティング・システムから空いているポート番号を取得します。

DBLocate - コマンド・ライン・ユーティリティによって、ネットワーク上で TCP/IP を実行している Adaptive Server Anywhere データベース・サーバがすべて表示されます。

サーバ・ネーム・スペースの変更 - ネットワークにおいて名前が同じ複数のデータベース・サーバを TCP/IP で実行することができなくなりました。これまでは、それぞれ独立したポート上にあるかぎり、名前が同じ複数のサーバを使用することができました。

最小パケット・サイズ - クライアント/サーバ通信の最小パケット・サイズは、200 バイトから 280 バイトに変更されています。この設定は、データベース・サーバの -p コマンド・ライン・スイッチと CommBufferSize 接続パラメータによって制御されます。

IPX のサポート終了 - 今後のリリースでは、IPX をサポートしない予定です。また、デフォルトでは IPX が起動しません。代わりに SPX プロトコルを使用することを強くおすすめします。IPX を使用するクライアントについては、独立した DLL (dbipx7.dll) が引き続き必要になります。SPX プロトコ

ルを使用する場合は、このライブラリは必要ありません。SPX/IPX ネットワークでは、SPX プロトコルを使用することをおすすめします。

サポート対象外のプロトコル - DDE、NetDG、および HLI は、サポート対象外となりました。

Broadcast と CommAutoStop - 通信パラメータの Broadcast と CommAutoStop は引き続き使用できますが、効力はありません。Adaptive Server Anywhere の今後のリリースでは、これらのパラメータをサポートしない予定です。

Agent 接続パラメータ - この接続パラメータに対するエンジンの値は無視されます。

接続動作の変更

一部のマシン接続では、共有メモリ通信プロトコルが使用されます。バージョン 7 では、接続文字列または DSN で CommLinks が指定されていない場合にのみ共有メモリでの接続が行われます。以前のリリースでは、次のパラメータを含んだ接続文字列を使用すると、TCP/IP で myengine が検索され、その検索が失敗した場合に共有メモリ接続が行われていました。

```
eng=myengine;links=tcip
```

バージョン 7 では、共有メモリ・リンクは一切実行されません。

Open Server Gateway の置き換え

SQL Anywhere バージョン 5 では、独立した Open Server Gateway 実行プログラムを使用して Open Client アプリケーションをサポートしていました。この実行プログラムを使用すると、Open Client アプリケーションに対して SQL Anywhere を Open Server のように見せることが可能でした。

Adaptive Server Anywhere バージョン 7 には、Open Server Gateway はありません。その代わりに、Adaptive Server Anywhere データベース・サーバを設定して、Open Server のように機能させることができます。この変更により設定が簡略化され、パフォーマンスおよび互換性が改善されます。また、Open Client アクセスをデータベースに提供するために必要なメモリとディスク・リソースも大幅に削減されます。

ODBC

Adaptive Server Anywhere バージョン 7 の ODBC サポートには、2 つの大きな変更点があります。

ネイティブ ODBC - ODBC ドライバは、Adaptive Server Anywhere に対するネイティブ・インタフェースになりました。ネイティブ・インタフェースは、コマンドをデータベース・サーバに送信する前に別のインタフェースに変換することを必要としないインタフェースです。

ODBC 3.0 のサポート - バージョン 3.0 の ODBC プログラミング・インタフェースは 1997 年に発表されました。Adaptive Server Anywhere バージョン 6 では、ODBC 3.0 をサポートしています。

SQL ダイアレクトの変更点

更新制約

新しい ANSI_UPDATE_CONSTRAINTS オプションが追加されました。このオプションを CURSORS または STRICT に設定すると、ANSI 規格によって許可されている更新だけに制限されます。このオプションを OFF (従来の動作) に設定すると、より広範囲の更新が許可されます。

Transact SQL 外部ジョインの変更点

Transact SQL 外部ジョインは WHERE 句で演算子 ($=$ and $=$) を使用しますが、SQL/92 外部ジョインは FROM 句で ON キーワードを使用してジョインを表します。バージョン 7 では、Transact SQL 外部ジョインの動作にいくつか変更が加えられています。そのため、旧バージョンのソフトウェアで使うことができたジョインのいくつかは、構文エラーを引き起こします。

Transact SQL 外部ジョインは、Sybase SQL Server (現在の Adaptive Server Enterprise) との互換性を維持するために実装されました。変更された内容は、最近のバージョンの Adaptive Server Enterprise で変更された内容を反映しています。これらの変更により、潜在的にあいまいな構文を持つ外部ジョインに対してエラーが発生します。クエリで必要となる精度を上げると、クエリ結果の信頼性は向上します。

特に、1 つのクエリ内で SQL/92 構文と Transact SQL 外部ジョイン構文を混在させることはできません。これは、クエリが使用するビューにも適用されます。外部ジョインに対する特定のダイアレクトを使用してビューを定義した場合は、そのビューの外部ジョイン・クエリに対しても同じダイアレクトを使用する必要があります。

Transact SQL 外部ジョイン句と通常のジョイン句の両方にテーブルを指定することはできません。たとえば、次のような WHERE 句は使用できません。

```
WHERE R.x *= S.x
AND S.y = T.y
```

外部ジョイン句と通常のジョイン句の両方でテーブルを使用しないようにクエリを書き直すことができない場合は、文を 2 つの独立したクエリに分ける必要があります。

外部ジョインの null 入力テーブルが含まれているサブクエリを使用することはできません。たとえば、次のような WHERE 句は使用できません。

```
WHERE R.x *= S.y
AND EXISTS ( SELECT *
             FROM T
             WHERE T.x = S.x )
```

NUMBER 関数のサポート終了

現行リリースでは NUMBER 関数をサポートしていますが、Adaptive Server Anywhere の今後のバージョンではサポートしない予定です。

トリガ内の @@identity

これまでは、オートインクリメント・カラムを持つテーブル (T1) に、オートインクリメント・カラムを持つ第 2 のテーブル (T2) に対して挿入を行う挿入トリガがある場合は、挿入の完了後に T1 に対して割り当てられたオートインクリメント値を取得することは不可能でした。その時点では、T2 に割り当てられた値が @@identity の値になります。この値をアクセスできるようにするために、@@identity の動作が変更されました。

ユーザ定義エラー

プロシージャとトリガの内部において、99000~99999 の範囲の例外を複合文内のユーザ定義エラーとして宣言できます。SIGNAL 文を使用すると、これらのエラーを処理できます。

ROLLBACK TO SAVEPOINT で閉じられるカーソル

旧バージョンのソフトウェアでは、ROLLBACK TO SAVEPOINT でカーソルは閉じられませんでした。現行バージョンでは、セーブポイント後に開いたカーソルは、そのセーブポイントへのロールバック時に閉じられます。

% コメント

% フォーマットのコメントは、サポート対象外の機能になりました。新しい Percent_as_comment オプションを使用して、% の意味を制御できます。互換性を維持するため、デフォルトの設定は ON です (つまり % はコメント・インジケータとして扱われます)。% フォーマットのコメントを使用して作成されたプロシージャ、トリガ、およびビューは、カタログに保存するときに二重ダッシュ記号のコメントに変換されます。既存のプロシージャは、オプションを変更する前に再作成する必要があります。% フォーマットのコメントを含んだ既存のプロシージャは、オプション設定を変更する前に再作成する必要があります。再作成しないと、プロシージャのロードに失敗します。これを行うには、データベースのアンロード／再ロードを使用します。

無視されるオプション

一部のデータベース・オプションは、コマンド・ライン・スイッチの機能と重複していました。これらのオプションは無視され、コマンド・ライン・スイッチだけに基づいて処理が行われます。これらのオプションには、THREAD_COUNT や THREAD_STACK が含まれます。

デフォルトの NEAREST_CENTURY オプション

NEAREST_CENTURY オプションのデフォルト設定が、0 から 50 に変更されました。この変更は既存のデータベースには影響せず、バージョン 6 以降で作成されたデータベースでのみ有効になります。

日付フォーマットの変更点

DATE_FORMAT オプションは、一連の定義済み文字 (月を表す MM など) で構成された文字列の引数をとります。フランス語の月名を供給する引数 f は、サポート対象外になりました。月と曜日の名前は、ソフトウェアの言語バージョンに従って供給されます。

START DATABASE 文、STOP DATABASE 文、STOP ENGINE 文

旧バージョンでは、Interactive SQL からのみ、これらの文を利用することができました。新しいバージョンでは、すべてのアプリケーションから利用できるようになっています。

ツールの変更点

Sybase Central

Sybase Central を Java アプリケーションとして利用できるようになりました。これにより、サポート対象のプラットフォームから Adaptive Server Anywhere への統一インタフェースが提供されます。

新しい Sybase Central にまだ実装されていない機能を利用する場合に備えて、Windows バージョンの Sybase Central も用意されています。この Windows バージョンには、新機能は追加されていません。C++ バージョンにアクセスする場合は、scview.exe 実行プログラムを使用して Sybase Central から C++ バージョンにアクセスできます。

Interactive SQL

Interactive SQL ユーティリティが強化され、Java アプリケーションとして利用できるようになりました。この新しい Interactive SQL には複数のウィンドウがあり、その各ウィンドウでは独立したデータベース接続が開かれます。

新しい Interactive SQL にまだ実装されていない機能を利用する場合に備えて、Windows バージョンの Interactive SQL も用意されています。

Windows バージョンには、新しい機能は追加されていません。この Windows バージョンには、dbisqlc.exe という名前が付けられています。

ISQL 名の変更

ISQL ユーティリティのファイル名が dbisql.exe に変更されました。これにより、Adaptive Server Enterprise の isql ユーティリティと名前が競合することはなくなります。

Interactive SQL DBTOOL 文

Interactive SQL DBTOOL 文は削除されました。この文は、Interactive SQL からデータベース管理タスクを実行するために用意されていました。他のアプリケーションからのスクリプティングを可能にするため、CREATE DATABASE などの管理用の SQL 文が追加されています。

INPUT と OUTPUT のフォーマット

INPUT 文と OUTPUT 文についてサポートされるフォーマットが変更されました。次に、そのフォーマットを示します。

- **INPUT** ASCII、DBASE、DBASEII、DBASEIII、EXCEL、FIXED、FOXPRO、LOTUS
- **OUTPUT** ASCII、DBASE、DBASEII、DBASEIII、EXCEL、FIXED、FOXPRO、HTML、LOTUS、SQL

強化された Interactive SQL ログ

Interactive SQL ログ・ファイルでは、エラーを引き起こした文の後にそのエラーがコメントとしてリストされます。

read コマンドによって読み込まれたファイル内容のログも取られます。実際の read 文は、コメント内にリストされます。

Interactive SQL の各セッション (または接続の各変更) では、ログがトランケートされなくなりました。ログを取られた文は既存のログ・ファイルに追加され、ログのトランケートはユーザの責任で行います。

SQL 文によって共通の管理タスクが実行される

管理ユーティリティによって実行されるほとんどのタスクを実行するために、SQL 文が導入されました。これらの SQL 文には、CREATE DATABASE、VALIDATE、CREATE WRITEFILE、BACKUP、および DROP DATABASE が含まれます。データベース管理タスク用の SQL 文の導入によってデータベース管理ユーティリティが機能の面で影響を受けることはありません。ただし、これらのユーティリティは、インストール・ディレクトリの scripts サブディレクトリ内にある SQL スクリプトを使用してタスクを実行します。

LOAD TABLE と UNLOAD TABLE

LOAD TABLE 文と UNLOAD TABLE 文を実行するのに必要なパーミッションを制御するために、データベース・サーバのコマンド・ライン・オプションが追加されました。

これらの文の FORMAT 句は、文字列ではなくキーワード (ASCII) をとります。文字列 'ascii' は、互換性のために許可されます。

DBWATCH の置き換え

dbwatch サーバ監視機能は、dbconsole というコンソール・ユーティリティに置き換えられました。このコンソール・ユーティリティは、ネットワーク・サーバのほかにパーソナル・サーバも監視できます。

DBINFO オプション

dbinfo ユーティリティは、データベース・ファイル名ではなく接続文字列を使用してデータベースを見つけます。

ユーティリティにおける dbo ユーザ

dbinit、dbupgrad、dbunload、および dbextract ユーティリティに対する -g コマンド・ライン・スイッチが削除されました。このスイッチは、Adaptive Server Enterprise データベース所有者のユーザ ID である dbo に対して別のユーザ名を指定するために使用されていました。

DBUNLOAD の変更点

アンロードするテーブルのリストを dbunload ユーティリティに渡すには、-t コマンド・ライン・スイッチを使用します。

カスタム照合ファイル

旧バージョンでは、照合名 (内部照合用) またはファイル名 (カスタム照合用) を `-z` の値にすることができました。現在のバージョンでは、照合名だけを指定できます。対応する照合は、`scripts` サブディレクトリ内の `collseqs.sql` または `custom.sql` で定義する必要があります。

RTSQL の削除

ランタイム SQL プロセッサは提供されません。`-q` を付けて `Interactive SQL` を使用すると、これまで RTSQL によって実行されていたタスクを実行することができます。

SQLRemote の変更点

DELETE_OLD_LOGS が ON のときにはミラー・ログが削除される

これまでは、古いトランザクション・ログのプライマリ・コピーが削除されても古いトランザクション・ログのミラーは削除されませんでした。

メッセージのコード化

電子メール・システムとの互換性を高めるために、SQL Remote バージョン 5 で使用したコード化が修正されました。コード化により、メッセージを壊さずにメッセージ・システムでメッセージを転送できます。

メッセージのサイズを小さくするために、メッセージ圧縮も追加されています。

また、ユーザ定義コード化の機能が追加されているため、独自の機能をメッセージ・フォーマットに追加することができます。

COMPRESSION データベース・オプションの値が `-1` に設定されている場合、バージョン 5 の Message Agent はバージョン 7 の Message Agent とメッセージを交換できます。インストールと同時にソフトウェアをアップグレードする必要はありません。

バージョン 7 のデータベース・サーバが必要

Adaptive Server Anywhere バージョン 7 の SQL Remote Message Agent には、バージョン 7 のデータベース・サーバが必要となります。この SQL Remote Message Agent は、SQL Anywhere バージョン 5 のデータベース・サーバに対しては動作しません。

DBXtract は内部アンロードを使用する

`dbxtract` のデフォルト動作は、`OUTPUT` 文を使用してクライアント側でデータをアンロードするのではなく、`UNLOAD` 文を使用してサーバ側でデータをアンロードするようになりました。`-ii`、`-ix`、`-xi`、`-xx` コマンド・ライン・オプションでは、使用する内部操作と外部操作の組み合わせを選択することができます。また、これらのオプションは、以前のリリースに用意されていた `-I` オプションと `-x` オプションを置き換えます。

データベース・サーバの変更点

サーバ・インタフェース

ネットワーク・サーバ・インタフェースは、dbconsole という名前の独立したユーティリティに移動されました。ネットワーク・サーバ自体は、パーソナル・サーバと同じウィンドウを表示します。

最小化されたサーバ・アイコン

Windows 95 および Windows NT 用のデータベース・サーバを最小化すると、タスク・バーではなくシステム・トレイにそのサーバがアイコンとして表示されます。

データベース起動パーミッション

データベース・サーバを起動するときには、-gd コマンド・ライン・オプションを使用して、データベースを起動できるユーザを制御できます。バージョン 6.0 以降の場合、ネットワーク・データベース・サーバのデフォルト設定では、DBA 権限を持つユーザだけがデータベースを起動できます。パーソナル・サーバについては、これまでどおりデフォルトですべてのユーザがデータベースを起動できます。

サポートされるオペレーティング・システム

サポートされるオペレーティング・システムのリストに、いくつかの変更があります。

- いくつかの UNIX オペレーティング・システムのサポートが開始された。
- Windows CE 2.11 がサポートされた。
- DOS と Macintosh オペレーティング・システムは、クライアントまたはサーバについてサポート対象外となった。
- Windows 3.1 と Windows 3.11 は、サポート対象外となった。

新機能

Adaptive Server Anywhere 7 に移行すると、数多くの新機能を利用できます。この項では、バージョン 7 へのマイグレーションを計画するときにユーザが詳しく調査する可能性のある機能について説明します。これは機能強化の完全なリストではないため、各機能に関する文書を十分に確認してから最終的な決定を行ってください。

エンジン

パフォーマンスとチューニングの容易さを同時に向上させるために、大幅な変更が Adaptive Server Anywhere に加えられました。次に、これらの変更について説明します。

マルチプロセッサのサポートと改良されたマルチタスキング - Adaptive Server Anywhere のデータ処理エンジンは、複数のオペレーティング・システム・スレッドを使用して実行できるため、複数のプロセッサを利用できます。

動的なキャッシュ・サイズ変更 - Windows NT と UNIX では、データベース・サーバ上の負荷とその他のシステム・メモリ要求に応じてデータベース・サーバ・キャッシュのサイズが増減します。この

機能により、多くの環境で明示的にキャッシュ・サイズを選択する必要がなくなります。また、この機能によってパフォーマンスを向上させることもできます。Windows 95/98 では、より包括性の低いキャッシュ・サイズ変更が実装されています。

インデックス処理の強化 - インデックスに格納された情報の量 (ハッシュ・サイズ) を制御してインデックスの選択性を改善するために、さらに柔軟性を向上しました。また、プライマリ・キー・インデックスと外部キー・インデックスのアーキテクチャが変更されています。

long 値のための独立した記憶領域 - long 値 (LONG BINARY、LONG VARCHAR、TEXT、IMAGE) の物理記憶領域が再編成されました。long 値は、独立したデータベース・ページ上に保存されます。この変更により、特に long 値が格納されているテーブルのスキャンを必要とするクエリでパフォーマンスが改善されます。

新しいデータベース・ページ・サイズ - 1K、2K、4K のページ・サイズのほかに、8K、16K、または 32K のページ・サイズでデータベースを作成できます。

クエリ・リライトの最適化が強化されたことにより、多くの場合、クエリの特定の構成方法は重要ではなくなります。つまり、データベース・システム自体がクエリをリライトするので、そのクエリを効果的な方法で実行できます。クエリ・リライト最適化機能を備えていないデータベース・システムでは、アプリケーション開発者が各クエリを最も効果的な方法でコーディングする必要があります。

サブクエリ・キャッシング - Adaptive Server Anywhere は、サブクエリのキャッシングを実行するようになりました (このキャッシングはメモライゼーションとも呼ばれます)。このキャッシングは、過剰なメモリ使用を避けるために 4 つのデータベース・ページに制限されます。

接続

JDBC のサポート

Sybase jConnect JDBC ドライバは、Open Client ライブラリが必要でない場合でも TDS を使用します。

Adaptive Server Anywhere は、Sybase jConnect を使用して JDBC 接続をサポートします。

OLE DB プロバイダ

OLE DB は、Microsoft のデータ・アクセス・モデルです。OLE DB では、COM (Component Object Model) インタフェースが使用されます。また、OLE DB は ODBC とは異なり、データ・ソースが SQL クエリ・プロセッサを使用することを想定していません。これまでは、Microsoft が提供する OLE DB/ODBC ブリッジを使用して OLE DB 経由で Adaptive Server Anywhere にアクセスすることが可能でしたが、このリリースの Adaptive Server Anywhere には OLE DB プロバイダが用意されています。このプロバイダには、次のような利点があります。

- OLE DB は、次期バージョンの Windows CE の最も重要なデータ・アクセス・オプションである。
- カーソルによる更新などの一部の機能は、OLE DB/ODBC ブリッジを使って利用することはできない。
- Adaptive Server Anywhere OLE DB プロバイダを使用した場合、展開において ODBC は必要ない。

SQL ダイアレクト

Adaptive Server Anywhere によって、いくつかの新しい機能が SQL ダイアレクトに追加されています。次に、その機能について説明します。

最終デフォルト・タイムスタンプ - 新しいグローバル変数 @@dbts は、DEFAULT TIMESTAMP を使用してカラムに対して生成された最後の値を表す TIMESTAMP 値を返します。

UNSIGNED データ型 - 符号なしバージョンの INT データ型と SMALLINT データ型が追加されました。これらのバージョンは、正の数値しか保持できませんが、符号付きバージョンよりも大きな最大値を持つことができます。

新しいデータ型 - 符号付きバージョンと符号なしバージョンの両方の 64 ビット整数データ型 (BIGINT) が追加されました。また、VARBINARY データ型もサポートされています。

新しい関数と句 - 多数の新しい関数と SQL 句が追加されました。次に、それらの関数と SQL 句を示します。

- STUFF および STR 文字列関数。
- CHAR_LENGTH 関数は、文字列式の文字数を返す。
- 潜在的に大きな結果セットの最初のいくつかを取得、更新、または削除可能な FIRST 句と TOP 句が追加された。
- FROM 句でクエリを使用することで、派生テーブルを指定できる。これにより、ビューを作成せずにグループ句のグループ化やグループ句付きのジョインを実行できるようになる。
- SETUSER 文により、データベース管理者は別のユーザ ID を持つユーザであるかのように行動できる。また、アプリケーション・サーバ環境では、SETUSER 文を使用して接続プーリングを行うことができる。
- UNLOAD TABLE は、binary データ型のカラムをアンロードするときに、バイナリ文字列ではなく 16 進文字列で書き出す。
- SQL REPLACE 関数は、見つかった部分文字列をすべて別の部分文字列に置き換える。
- LIST 関数は、オプションとして第 2 の値を受け入れる。この値は、リスト項目を区切るデリミタ文字列である。
- 新しい UNLOAD SQL 文により、クエリ結果セットをカンマ区切りのテキスト・ファイルにアンロードできる。
- LOCK TABLE 文は、現在の独立性レベルに関係なくテーブル・レベルで同時実行性を直接制御できる。
- 新しいシステム・プロシージャの sa_check_commit により、データベースに対する変更をコミットする前に参照整合性の競合をチェックできる。

Transact-SQL 外部ジョイン内の式 - *= and =* 演算子により、ジョイン演算子の両側が 1 つのテーブルを参照しているかぎり、これらのジョインで任意の式を使用できます。

新しいシステム・プロシージャ - システム・ストア・プロシージャから他の管理タスクを実行できます。それらのタスクには、データベース内の全テーブルの検証、ファイル使用情報の取得 (ページ数など) が含まれます。

ロックのトラブルシューティング - 新しいシステム・プロシージャの `sa_locks` は、データベース内のロックに関する情報を提供します。ロックの問題が検出された場合は、AppInfo 接続プロパティを使用して関係する接続プロセスの情報を見つけることができます。

イメージおよび文書の挿入とエクスポート - 2 つの新しい外部関数により、ファイル内容の読み込みと書き込みを行うことができます。これらの関数を使用すると、イメージや文書などを Interactive SQL などの環境からテーブルに直接挿入できます。

タスクのスケジューリング

スケジューリングした操作をデータベースに追加できるようになりました。これは、自動バックアップや、サマリ・テーブルに入力するための定期レポートなどのタスクで役立ちます。

また、データベース・ファイルまたはトランザクション・ログ・ファイルを保持しているドライブ上のディスク領域スレッシュホールドや接続試行の失敗などの特定のイベントが発生したときに、データベース・サーバにイベント・ハンドラを実行させることもできます。

Java

Adaptive Server Anywhere はデータベース内の Java に対するランタイム環境を提供し、Java クラスを SQL データ型として使用することを可能にします。

ユーザは、独自の Java クラスをデータベースにインストールできます。Java クラスのパブリック静的メソッドは、ストアド・プロシージャおよびユーザ定義関数として利用可能になります。

Java クラスは、SQL ユーザ定義データ型として使用できます。Java クラスをテーブル内のカラムのデータ型として使用すると、Java オブジェクトの形式でデータをデータベースに格納できます。

ツール

ユーティリティ・データベース

データベースに接続せずに新しい SQL データベース・ファイル管理文を実行可能なユーティリティ・データベースが追加されました。ユーティリティ・データベースは、物理的に表示されない幻データベースです。ユーティリティ・データベースにはデータベース・ファイルが存在しないため、データを含めることはできません。

バックアップ、リカバリ、および検証

この分野では、いくつかの機能強化が行われています。次に、それらの機能強化について説明します。

NetWare でのアーカイブ・バックアップ - NetWare でアーカイブ・バックアップ・フォーマットがサポートされます。テープへのアーカイブ・バックアップを行うには、NetWare 5 が必要です。

検証 - 新しい VALIDATE INDEX 文と強化された VALIDATE TABLE 文によって、データベースをさらに検証できます。

データベースのバックアップ・コピーの検証 - WAIT BEFORE START 句を使用してデータベースをバックアップした場合は、読み取り専用モードで起動して検証できるようにバックアップ・コピーが作成されます。

対話型のデバグ

ストアド・プロシージャとトリガを対象としたフル装備の対話型デバグが導入されました。このデバグでは、Java クラス、プロシージャ、トリガ、およびイベント・ハンドラのデバグ、クラスとストアド・プロシージャの参照、実行のトレース、ブレークポイントとブレーク条件の設定、ローカル変数およびロー変数の検査と修正、式の検査と中断、クエリの実行など、多数のタスクを実行できます。

SQL Remote

デフォルト・グローバル・オートインクリメント - この機能を利用すると、SQL Remote レプリケーション環境内のデータベース全体で一意的な整数キーを簡単に生成できます。

SQL Remote 用のアンロードと再ロード - Adaptive Server Anywhere 用の dbunload および dblog ユーティリティには、SQL Remote インストールで統合データベースのアンロードと再ロードに使用可能な機能が追加されています。

データベース抽出のセキュリティ強化 - データベースの抽出時には、ディスクで作成されたファイルに重要なデータベース情報が含まれている場合があります。これが問題になる可能性があるサイトについては、Unload ユーティリティと Extraction ユーティリティの -an および -ac オプションによって、セキュリティを強化できます。これらのオプションを使用すると、どのデータもディスクに書き込まれません。セキュリティ強化の代償として、パフォーマンスが多少低下します。

ロケール情報 - Adaptive Server Enterprise 用の Message Agent において、供給された言語およびロケール情報を使用できるようになりました。

Adaptive Server Enterprise への LONG データ型 (blob) のレプリケーション - Adaptive Server Anywhere における LONG BINARY データ型と LONG VARCHAR データ型、および Adaptive Server Enterprise における IMAGE 型と TEXT 型は、バイナリ・ラージ・オブジェクト (blob) と呼ばれることがあります。ユーザは、Adaptive Server Enterprise を使用して、これらのオブジェクトのいくつかを SQL Remote インストールでレプリケートすることができます。

最小化された Message Agent - -q コマンドライン・スイッチを使用すると、最小化されたウィンドウで Message Agent を起動できます。

Message Agent のメッセージ再送信要求 - 失われたメッセージの再送信をどの時点で Message Agent が要求するかは、-rp コマンドライン・スイッチを使用してユーザが設定できます。

ステーブル・キューのクリーニング - Adaptive Server Enterprise では、Message Agent の新しい -fq コマンドライン・オプションを使用してステーブル・キューから確認済みメッセージをクリーニングすると、管理が容易になります。

SQL Remote メッセージ・ロギング - 新しいコマンドライン・オプションにより、これらのユーティリティからメッセージ・ロギングを調整できます。

データベースに格納されたメッセージ・リンク・オプション - 各メッセージ・システムで SQL Remote の動作を制御するメッセージ・リンク・パラメータを、レジストリではなくデータベースに保存できます。これにより、メッセージ・リンク・パラメータに関連した配備および管理の問題が簡素化されます。

日付と時刻のレプリケーション・フォーマット - 日付と時刻のレプリケート時に使用するフォーマットを SQL Remote に指示するためのデータベース・オプションが指定できるようになりました。これらのオプションは、SR_time_format、SR_date_format、および SR_timestamp_format です。

デーモンとして実行された Message Agent と SQL Remote Open Server - UNIX オペレーティング・システムでは、-ud コマンドライン・オプションを使用して、これらのアプリケーションをデーモンとして実行できます。

より簡単になった Adaptive Server Anywhere データベースのアンロードと再ロード - dbunload ユーティリティが強化され (-ar コマンド・ライン・オプション)、データベースがレプリケーションに関係しているかどうかにかかわらずシングル・ステップでデータベースのアンロードと再ロードを行うことができます。

強化された dblog 出力 - dblog ユーティリティは、オフセット情報を含む要約情報を追加表示します。

dbtran ユーティリティの機能強化 - dbtran コマンド・ライン・ユーティリティでは、トランザクション・ログ操作のフィルタリングを行って、操作のサブセットを分離することができます。これは、特に SQL Remote の管理者に役立ちます。

UNIX での FTP と SMTP/POP のサポート - UNIX オペレーティング・システムでサポートされるメッセージ・システムの範囲が広がり、FTP と SMTP/POP もサポート対象になりました。

メッセージの圧縮 - メッセージを圧縮して効率を上げることができます。

アップグレード手順に関する問題

アップグレードで利用できるオプションは、Adaptive Server Anywhere の使い方によって異なります。検討すべき主なシナリオは 3 つあります。

- スタンドアロン・インストール
- クライアント/サーバ・インストール
- SQL Remote インストール

アップグレード手順の詳しい例については、『バージョン 7 ASA 入門』の第 5 章「Upgrading Version 5 Applications」を参照してください。

上記の各シナリオを検討する前に、バージョン 5.5 のデータベースをアップグレードする必要があるかどうかを判断する必要があります。

いつデータベースをアップグレードすべきか

Adaptive Server Anywhere データベース・サーバでは、以前のデータベースを使用できます。既存のアプリケーションを引き続き使用するためにデータベースをアップグレードする必要はありません。

バージョン 7.0 の新機能には、次のような種類があります。

データベースの変更を必要としない新機能 - これらは、データベース・オプション、システム・テーブルの変更、データベース記憶領域の変更に関連しない機能です (例: 動的キャッシュ・サイズ変更)。

新しいデータベース・オプションやシステム・テーブルの変更に依存している新機能 - (例: スケジューリングおよびイベント処理、バージョン 7 のデバッガの使用)。これらの機能を利用するには、データベースをアップグレードする必要があります。Upgrade ユーティリティまたは ALTER DATABASE 文を使用すると、データベースをアップグレードできます。

変更されたデータベース・ファイル・フォーマットに依存している新機能 - これらの機能を利用するには、データベースのアンロードと再ロードを行う必要があります。バージョン 7.0 では、可変ハッシュ・サイズ・インデックス、独立したプライマリ・キー・インデックスと外部キー・インデックス、blob 用に再編成された記憶領域、および GLOBAL DEFAULT AUTOINCREMENT カラムがこれらの機能に含まれます。また、データベース記憶領域の再編成により、一部の操作のパフォーマンスが向上します。

スタンドアロン・インストールまたはクライアント/サーバ・インストール

スタンドアロン・インストールとクライアント/サーバ・インストールのアップグレード手順は、ほぼ同じです。どちらの場合も、SQL Anywhere ソフトウェアをアップグレードし、接続論理を調べて、その接続論理に互換性があることを確認する必要があります。クライアント/サーバ通信プロトコルを変更すると、バージョン 5 のインタフェース・ライブラリはバージョン 7 のデータベース・サーバと交信できなくなります。

システムのコンポーネント

SQL Anywhere バージョン 5 を現在実行している場合は、次に示すコンポーネントの一部または全部がシステムに含まれます。

アプリケーション - SQL Anywhere コンポーネント以外のアプリケーション。

接続パラメータ - SQL Anywhere バージョン 5.0 の接続パラメータ。これらのパラメータは、ODBC データ・ソースを使用してアセンブルするか、または他の方法でアセンブルできます。

ドライバ・マネージャ - ODBC アプリケーション用の ODBC ドライバ・マネージャ。

ODBC ドライバ - ODBC アプリケーション用の SQL Anywhere バージョン 5 ODBC ドライバ。ネットワーク・アプリケーションの場合は、クライアント・マシンに ODBC ドライバがあります。

インタフェース・ライブラリ - SQL Anywhere バージョン 5 のインタフェース・ライブラリは、ODBC および Embedded SQL アプリケーションによって使用されます。ネットワーク・アプリケーションの場合は、クライアント・マシンにインタフェース・ライブラリがあります。

SQL Anywhere クライアント - ネットワーク・アプリケーション用の dbclient.exe 実行プログラムとコマンド・ライン。コマンド・ラインでは、サーバ名と一連のネットワーク通信パラメータを指定できます。このクライアントは、バッチ・ファイルまたは ODBC データ・ソースの Start Line パラメータに格納できます。ネットワーク・アプリケーションの場合は、クライアント・マシンに SQL Anywhere クライアントがあります。

データベース・サーバ - SQL Anywhere バージョン 5 のデータベース・サーバ。ネットワーク・アプリケーションの場合は、クライアント・コンポーネントとは異なるマシンにこのサーバがあります。

データベース - SQL Anywhere バージョン 5 のデータベース。これは、データベース・サーバと同じマシンにあります。

バージョン 7 にアップグレードする場合は、バージョン 7 の ODBC または Embedded SQL アプリケーションで使用できるような方法でバージョン 5 の dbclient コマンド・ライン情報が取得されることを確認する必要があります。クライアントのコマンド・ライン情報を取得する方法は、その情報が配置されている場所によって異なります。既存のコマンド・ライン情報は、次のいずれかの場所に保存されています。

ODBC データ・ソース - ODBC データ・ソースには、クライアント実行コマンド・ラインを保持できる START パラメータが含まれています。

接続文字列として - アプリケーションは、クライアント情報を (たとえば初期化ファイルから) 取得して、それを START パラメータとして接続文字列に含めることができます。

バッチ・ファイル - 起動プロセスの一部としてクライアント実行コマンド・ラインが含まれているバッチ・ファイルを使用できます。

アイコンの下 - デスクトップのアイコンの下でクライアント実行コマンド・ラインを使用できます。

クライアント・コマンド・ライン情報を取得した後、バージョン 7 へのアップグレードを実行して、ODBC データ・ソースまたは接続文字列に適切なエントリを作成することができます。

SQL Remote インストールのアップグレード

SQL Remote インストールには、各サイトの Message Agent に加えて、統合データベースと多数のリモート・データベースが含まれます。

各サイトでは、Message Agent がメッセージの送受信を処理します。メッセージは SQL 文の形式であり、データベース・サーバがこれらの SQL 文を実際に実行します。

SQL Remote のアップグレード要件は次のとおりです。

データベースのアップグレードは不要 - バージョン 7 用にデータベースをアップグレードする必要はありません。ただし、データベースをアップグレードする場合は、任意のサイトで Upgrade ユーティリティを使用してその処理を実行できます。すべてのデータベースを同時にアップグレードする必要はありません。

ソフトウェアのアップグレードは一度に 1 サイトずつ行える - バージョン 5 の Message Agent は、COMPRESSION データベース・オプションの値が -1 に設定されている間は、バージョン 7 の Message Agent とメッセージを交換することができます。インストール環境全体にわたって同時にソフトウェアをアップグレードする必要はありません。

アンロードや再ロードを行わない - バージョン 5 からバージョン 7 に移行するときにデータベースのアンロードと再ロードを行う必要はありません。

レプリケーションはトランザクション・ログに基づいていますが、データベースのアンロードと再ロー

ドを行うと、古いトランザクション・ログは利用できなくなります。そのため、レプリケーションを行うときにはバックアップをとることが非常に重要になります。