

アサーション・エラーの対処方法

本書では、アサーション・エラーのメッセージが表示されたときにデータベースをリカバリする手順について説明します。また、アサーション・エラーについてテクニカルサポートに問い合わせるときに用意する情報チェックリストも収録しています。

本書は、Adaptive Server Anywhere バージョン 5.5.03 ~ 9.0.2 を対象にしています。参照先のマニュアルは、すべて SQL Anywhere Studio 9.0.2 のマニュアルです。

アサーションとは何か？

Adaptive Server Anywhere には、早期にデータベースの破損を検出するための多くの内部チェックがあります。アサーションに失敗すると、データベース・サーバは現在の要求をすべてキャンセルし、データベース・サーバが終了するまで、それ以降の要求に対してエラーをレポートします。データベース・サーバは、アサーション・エラーの発生後にデータベース処理を拒否することで、破損がデータベース全体に広がる危険性を最小限に抑えます。

アサーションが実行されると、データベース・サーバが処理を中断し、[サーバ・メッセージ] ウィンドウまたは出力ログに以下の内容が出力されます。

```
*** ERROR *** Assertion failed: 123456 (x.0.x.xxxx)
Error message follows with information about the cause of the assertion.
```

注意: アサーション・エラーは、SQL Anywhere Studio のバージョンによって異なる場合があります。アサーション・エラーの発生時に表示されるエラー・メッセージは、SQL Anywhere Studio のリリースごとに少し異なります。アサーション・エラー・メッセージの内容をすべてメモしておくことをお勧めします。

アサーション・エラーの発生後に、アサーション番号とアサーションの原因となったエラーの詳細が表示されます。アサーション・エラーは、データベース・サーバが予想しない状態に遭遇したことを示します。予想しない状態が発生すると、データベースの破損の進行を防止するため、データベース・サーバは現在のすべての接続を終了し、新しい接続を拒否し、データベースを停止します。

予想しない状態が発生すると、ハード または ソフト のいずれかのアサーション・エラーが発生することがあります。また、ソフトウェアのバグが原因の場合があります。ハード アサーションは、通常は外部プロセスやハードウェアの障害が原因で、物理的なデータベース・ファイルまたはトランザクション・ログの整合性に問題が発生したことを示します。ソフト アサーションは、複数の接続に影響する危険性がある予想しない障害（誤ったクエリ結果、バックグラウンド・プロセスの不正動作など）の原因となる状況が、データベース・サーバで発生した場合に発生するこ

とがあります。データベースのアサーション・エラーは、物理データベース・ファイルの問題が根本的な原因となっていることがあります。

アサーション・メッセージが表示された時に行うこと

1. [サーバ・メッセージ] ウィンドウまたはログ・ファイルに出力されたアサーション番号とメッセージを記録します。アサーション番号とメッセージは、アサーションの原因を特定する場合に重要になります。テクニカルサポートへお問い合わせ頂くときに、この情報が必要になります。
2. データベース・サーバがまだ実行されている場合は終了します。これは、複数のデータベースを 1 つのデータベース・サーバで実行している場合に非常に重要です。これにより、同一のデータベース・サーバで実行されている他のデータベースの破損を防止します。
3. データベース・ファイルとトランザクション・ログのバックアップ・コピーを作成します。また、まだ削除していない以前のトランザクション・ログのバックアップ・コピーも作成します。これらのファイルのバックアップ・コピーは、単純なファイル・コピーで作成する必要があります (dbbackup または他のバックアップ・プロシージャは使用しないでください)。以前のデータベース・ファイルとトランザクション・ログのバックアップ・コピーは上書きしないでください。リカバリには、データベース・ファイルとトランザクション・ログをアサーション・エラーの発生直後の状態に維持することが重要です。
4. データベース・サーバを再起動して、データベース・ファイルをロードします。
5. データベースが正常に起動した場合は、検証ユーティリティ (dbvalid や同等ユーティリティを使用する、または sa_validate プロシージャを使用する) を使用してデータベースを検証します。
 - dbvalid でエラーがレポートされた場合、またはアサーション・エラーが再発した場合は、データベース・ファイルが破損している可能性があります。この場合は、検証済みのバックアップとリカバリ方法を実行することをお勧めします。
 - dbvalid が正常に終了し、エラーがレポートされなかった場合は、そのデータベースを再度運用状態に戻すことができます。それ以降にアサーション・エラーが発生した場合は、検証済みのバックアップとリカバリ方法を実行してください。検証済みのバックアップとリカバリ方法を実行した後も問題が発生する場合は、テクニカルサポートにお問い合わせください。

dbvalid でデータベース検証後にエラーがレポートされない場合でも、わずかな破損が発生している可能性があります。データベース・ファイルの破損を解消するには、unload/reload プロセスを使用してデータベースを再構築する必要があります。データベースを再起動するだけでは、データベース・ファイルに破損がないことを確認できません。

6. データベース・サーバが正常に起動した場合は、データベースのリカバリプロセスが試行されます。リカバリプロセスが成功したときに、検証して破損の修復を完了する前にデータベースを運用状態に戻す必要がある場合は、新しいトランザクション・ログを使用してデータベースを再起動する必要があります。データベースを新しいトランザクション・ログで起動するには、実行中のデータベースを終了し、既存のトランザクシ

ン・ログの名前を変更して、データベースを再起動します。

手順 5 でデータベースが正常に起動しない場合は、データベースをトランザクション・ログなしで強制起動することができます。ただし、データベース環境がレプリケーションや同期 (Mobile Link、SQL Remote、または Replication Server) に関係していない場合に限りです。これには、次のコマンドを使用します。

```
dbengX -f dbfile.db
```

- X には 50, 6, 7, 8, または 9 を指定します。
- *dbfile.db* にはデータベース・ファイルのパスを指定します。

通常は、データベース・サーバをリカバリするときに、データベース・ファイルが前のチェックポイントまでロールバックされ、そのチェックポイント以降のトランザクション・ログ内のすべてのトランザクションが適用されます。dbengX -f を実行すると、トランザクション・ログ内のトランザクションを適用せずに、最後のチェックポイントまでリカバリすることができます。これは、データベース・ファイルが想定しているトランザクション・ログの場所にトランザクション・ログがない場合に限りです。トランザクション・ログがある場合は、データベース・サーバのコマンド・ラインで -f を使用しているかどうかに関係なく、サーバはログ内で最後のチェックポイント以降のトランザクションを適用しようとします。トランザクション・ログの破損が疑われる場合は、トランザクション・ログが使用されないようにするため、トランザクション・ログの名前を変更してから dbengX -f を実行する必要があります。

どうすればアサーションからデータベースを保護することができるか？

アサーション・エラーに対する最適な予防策は、オペレーティング・システムのクラッシュ、ディスク障害、ファイル破損、マシンの全体的な故障を考慮した、検証済みのバックアップとリカバリ方法です。検証済みのバックアップとリカバリ方法により、アサーション・エラー発生時のダウンタイムを最小限に抑えることができます。

データベースとトランザクション・ログの有効なバックアップからリカバリする方法

データベースのアサーション・エラーの発生または破損からのリカバリには、データベース・ファイルとトランザクション・ログの有効なバックアップからデータベースをリカバリすることが最適な方法です。最後の有効なバックアップからアサーション・エラーの発生時までの間のシーケンスで失われたトランザクション・ログがなければ、アサーション・エラーによるデータ損失はありません。

以下の手順は、最適なバックアップ・ストラテジの一環としてすでに実行されているはずですが、理想的には、データベースのバックアップのコピーを事前に検証し、バックアップの整合性を確認しておいてください。

注意: データベースが有効かどうかをテストする場合は、必ずバックアップのコピーを使用するか、読み取り専用モードでデータベース・サーバを実行してください。検証ユーティリティがデータベース・オフセットを変更してしまうため、これらのテストは、バックアップのコピーに対して実行するか、読み取り専用モードでデータベース・サーバを実行して行う必要があります。

バックアップの整合性をまだ確認していない場合は、バックアップのコピーに対して 検証ユーティリティ (dbvalid) を使用し、そのコピーが有効かどうかを確認してください。このプロセスを実行することで無効なデータベースの大部分が検出されますが、それでもデータベースにあるわずかな破損が検出されない場合があります。バックアップが破損していないことを確認するには、dbunload を実行し、バックアップ・コピーから新しいデータベースを作成します。リロード時にエラーが発生せずにデータベースが再構築された場合は、バックアップ・コピーは有効です。

これで、データベースのバックアップのコピーの検証が完了しました。次に、バックアップの別のコピーをリカバリします。ここでリカバリしたデータベースのコピーに対し、トランザクション・ログを適用します。トランザクション・ログを適用すると、アサーション・エラーの発生時までデータベースに対して実行されたすべてのトランザクションが実行されます。最後のバックアップ以降にトランザクション・ログが切り捨てられていない場合は、適用する必要があるトランザクション・ログは 1 つだけです。これは、次のようにして実行します。

トランザクション・ログの名前が変更されて切り捨てられている場合は、このコマンドを使用して、各トランザクション・ログを順に適用します。最初に、データベース・ファイルと共にバックアップされたトランザクション・ログを適用します。名前の矛盾を避けるため、適用するすべてのトランザクション・ログは、データベース・ファイルとは別のディレクトリに保存する必要があります。すべてのトランザクション・ログを適用したら、データベースを再起動します。データベースに新しいトランザクション・ログが作成され、現行のトランザクション・ログになります。

このプロセスでは同期やレプリケーションは中断されませんが、これらのトランザクション・ログのいずれかに、同期またはレプリケーションに必要なオフセットがある場合に備えて、適用したトランザクション・ログを保存しておく必要があります。これが同期またはレプリケーション用環境でない場合は、適用したトランザクション・ログを保存しておく必要はありません。

有効なバックアップが存在しないが、運用開始時からの有効なトランザクション・ログがある場合のリカバリ方法

Adaptive Server Anywhere のデータベース・アーキテクチャでは、トランザクション・ログとデータベース・ファイルを使用します。トランザクション・ログには、データベースに対して実行されたすべての文が記録されます。その結果、すべての処理が記録された単一のトランザクション・ログを使用して、データベース全体を作成し直すことができます。このプロセスは、同期およびレプリケーションを中断します。

データベースのリカバリ方法

1. 初期化ユーティリティを使用して、既存のデータベースと同一の初期化パラメータで新しいデータベースを作成します。ほとんどの初期化パラメータは、情報ユーティリティ (dbinfo) を使用して取得できます。場合によっては、破損したデータベースに対して dbinfo を実行できないことがあります。

注意: 情報ユーティリティは、Adaptive Server Anywhere 8.0.0 以降のデータベース内の Java についての情報だけを返します。Adaptive Server Anywhere 7 以前のデータベースで Java を使用している可能性がある場合は、以下の手順を参照してください: ["Recovery From a Single, All-Inclusive Log File"](#)

初期化ユーティリティの詳細については以下を参照してください:

http://www.ianywhere.com/developer/product_manuals/sqlanywhere/0902/ja/html/dbdaja9/00000585.htm

情報ユーティリティの詳細については以下を参照してください:

http://www.ianywhere.com/developer/product_manuals/sqlanywhere/0902/ja/html/dbdaja9/00000584.htm

2. トランザクション・ログの変換時に問題が発生した場合に備えて、トランザクション・ログのバックアップ・コピーを作成します。
3. Log Translation ユーティリティを使用して、トランザクション・ログを変換します。

これで、データベースに対して過去に実行されたすべての文を含む SQL スクリプト・ファイルが作成されます。(デフォルトで、そのファイルは *transaction-log-file.sql* という名前になります)

```
dbtran dbfile.log
```

- o *Dbfile.log* にはトランザクション・ログのパスを指定します。

ログ変換ユーティリティの詳細については以下を参照してください:

http://www.ianywhere.com/developer/product_manuals/sqlanywhere/0902/ja/html/dbdaja9/00000598.htm

[m](#)

4. データベース・サーバを起動します。

```
dbengX -n reload dbfile.db
```

- X には 50, 6, 7, 8, または 9 を指定します。
- *dbfile.db* にはデータベース・ファイルのパスを指定します。

5. Interactive SQL を使用して、ログ変換ユーティリティで作成された SQL スクリプト・ファイルを適用します。

```
dbisql -c "UID=dba;PWD=sql;ENG=reload" c:¥transaction-log-file.sql
```

transaction-log-file.sql にはログ変換ユーティリティで作成した SQL スクリプトを指定します。

Interactive SQL の詳細については以下をご参照下さい：

http://www.iAnywhere.com/developer/product_manuals/sqlanywhere/0902/ja/html/dbdaja9/00000588.htm

データベースをレプリケーションや同期する場合に注意する点は何か？

レプリケーションまたは同期用環境では、トランザクション・ログとトランザクション・ログのオフセットについて注意すべき点があります。データベースを再構築すると、トランザクション・ログのオフセットが元のデータベースとは同一でないため、これらの環境では問題が発生します。このため、トランザクション・ログなしでデータベースを強制起動しないでください。データベースの再構築が必要な場合は、以下の手順で実行します。

```
dbunload -ar c:¥path-for-old-dbfiles -c "UID=dba;PWD=sql;ENG=dbserver"
```

注意: Mobile Link 統合データベースの場合は、トランザクション・ログのオフセットに依存しないため、この制限は適用されません。ただし、この場合もトランザクション・ログの扱いには注意が必要です。Mobile Link のリモート・データベースの場合は適用されます。

レプリケーションまたは同期に関係する環境でアサーションに失敗した場合、検証済みのバックアップとリカバリ方法を使用してリカバリするのが最適です。

データベースの起動には成功し、データベースに対して特定の処理を実行したときにだけアサーション・エラーが発生する場合は、インデックスまたはテーブルの破損が原因になっていることがあります。このような破損は、多くの場合、データベースを再構築し、破損したインデックスまたはテーブルを回避することで解決できます。

破損したインデックスがあるデータベースでは、以下のようなアサーション・メッセージが表示される可能性があります：

```
*** ERROR *** Assertion failed: 100305 (9.0.2.3124)
Invalid index page encountered during an index scan @1
(table id 12, page 0x1a8)
```

このメッセージは、インデックス破損により発生する可能性のある多数のエラーのうちの 1 つです。インデックス破損が原因のアサーション・エラー・メッセージは多数あります。

アサーション・エラー・メッセージだけで、破損しているインデックスを特定できることもあります。破損しているインデックスを特定できない場合、または上記の手順によりデータベースに他のアサーション・エラーが発生した場合は、データベースを無順序で再構築します。すでに再構築を行っている場合は、データベース・ファイルとトランザクション・ログのバックアップ・コピーに戻してから、無順序の再構築を実行します。無順序の再構築を実行するには、次のコマンドを使用してデータベースをアンロードします。ここで、X には 50、6、7、8、または 9 を、*dbfile*.db にはデータベース・ファイルのパスを指定します。

```
dbunload -u -ar c:¥path-for-old-dbfiles -c "UID=dba;PWD=sql;DBF=c:¥dbfile.db"
```

テーブル破損が原因のアサーション・エラーからのリカバリでは、まず破損したテーブルを特定します。Validation ユーティリティを使用してデータベースを検証することで、テーブルを特定できます。

```
dbunload -u -c "UID=dba;PWD=sql;DBF=c:¥dbfile" c:¥unload
dbinit newdbfile.db
dbengX -n new newdbfile.db
```

reload.sql ファイルを適用するために以下のコマンドを実行します：

```
dbisql -c "UID=dba;PWD=sql;ENG=new" c:¥reload.sql
```

- dbunload によって作成される SQL スクリプト・ファイルの、reload.sql にパスを指定します。

破損したテーブルがあるデータベースでは、以下のようなアサーション・メッセージが表示される可能性があります：

```
*** ERROR *** Assertion failed: 201501 (9.0.2.3124)
Page for requested record not a table page or record not present on page
```

テーブル破損によって引き起こされるアサーションからリカバリする第一歩は、破損しているテーブルを見つけることです。

これは、検証ユーティリティを使用して、データベースを検証にすることにより行うことができます。

検証ユーティリティの詳細については以下をご参照下さい：

http://www.ianywhere.com/developer/product_manuals/sqlanywhere/0902/ja/html/dbdaja9/00000627.htm

有効なバックアップが存在しないが、データベースの起動ですぐにアサーションが発生しない場合のデータ・サルベージ

上記の提案のいずれでもデータベース・アサーションを解決することができない場合、テクニカルサポートへお問い合わせを行うことをお奨めします。

サポートプログラムに関する詳細は以下をご参照下さい：<http://www.ianywhere.jp/sup/index.html>

iAnywhere Solutions は、破損データベースからデータをリカバリするデータ・サルベージ・サービスを提供しています。これは有料サービスで、テクニカルサポートへのお問い合わせを開始することで利用します。データ・サルベージ料金は、テクニカルサポートへのお問い合わせへの追加料金となります。データ・サルベージ・サービスでは、テクニカルサポートを開始する企業と iAnywhere Solutions の間での署名入りの契約書が必要です。契約では、サルベージ可能なデータの容量、データの整合性、サービス時のサルベージに要する時間について、iAnywhere Solutions が一切の保証を行わないことが明示されています。このサービスは、他の方法でデータをリカバリできなかった場合の最終手段として利用してください。データベース内のデータを保護するには、検証済みのバックアップとリカバリ方法を利用するのが最適な方法です。破損したデータベースからデータをリカバリするデータ・サルベージ・サービスは、検証済みのバックアップとリカバリ方法の代替方法ではありません。

データ・サルベージのためにデータベースを送信する場合、以下の質問に対してできるだけ多くの回答がテクニカルサポートで必要です。この情報は、データベースをできるだけ効率的にリカバリし、破損の原因を特定するのに役立ちます。

データベース破損のチェックリスト

1. DBA（または DBA 権限）のユーザ名とパスワード
2. アサーション・エラー・メッセージの番号と内容

アサーション・エラー・メッセージの番号と内容は、アサーションの原因を特定する場合に重要になります。アサーションの原因を特定することで、アサーションに対処する手順を特定するための情報を得ることができます。

3. このデータベースの作成に使用したソフトウェアのバージョン

アサーションが、データベース作成に使用したソフトウェアのバージョンに固有のバグが原因であった場合は、問題がすでに解決されている可能性があります。この情報があれば、アサーションの原因を特定してケースを解決するまでの時間を大幅に短縮できます。

4. データベース・サーバとして使用している Adaptive Server Anywhere のバージョンとビルド、およびデータベース・サーバを実行しているオペレーティング・システムとそのバージョンとビルド

Adaptive Server Anywhere のバージョンとビルドは、アサーション・エラーがソフトウェアのバグに関係するものかどうかを特定する場合に重要です。データベース・サーバが最新の EBF でない場合は、最新の EBF の README ファイルを読み、データベースのアサーション・エラーの原因となっている可能性のあるソフトウェア・バグがないかどうかを確認することをお勧めします。テクニカルサポートの担当者がデータベースのアサーション・エラーの原因を診断するには、オペレーティング・システム、Adaptive Server Anywhere のバージョン、ビルドの情報が必要です。

5. 可能であれば、（必要に応じてバックアップからの）dbinfo 出力

Adaptive Server Anywhere のバージョン 6、7、8、9 を使用している場合は、次のコマンドを使用して dbinfo 出力を取得します。このとき、ユーザ ID、パスワード、データベース・ファイル名は自身の環境での値を指定します。

```
dbinfo -o c:¥info.txt -c "UID=dba;PWD=sql;DBF=c:¥asdemo.db"
```

SQL Anywhere Studio バージョン 5.x.x を使用している場合は、次のコマンドを使用して dbinfo 出力を取得します。

```
dbinfo -o c:¥info.txt asdemo.db
```

6. データベース・ファイルに対して dbupgrad を実行したことがあるかどうか

これは、以前のバージョンの Adaptive Server Anywhere を使用してデータベース・ファイルを作成したかどうかを特定するために重要です。これがアサーション・エラーの原因である可能性は非常に低いですが、テクニカルサポートの担当者はこの情報が必要になります。

7. データベース・ファイルを保存しているファイル・システムの種類（SAN、FAT、NTFS、UFS）、FAT ファイル・システムの場合は、ハードディスク上のクラスタ・サイズ
8. 使用しているハードディスク・コントローラの種類。最新のハードディスク・コントローラ用ドライバをインストールしているかどうか。ディスク・コントローラにハードウェア・キャッシュがあるかどうか
9. サーバ・コンピュータが搭載している CPU の個数。CPU がハイパー・スレッド対応かどうか

10. データベース・サーバを起動したコマンド・ライン

アサーション・エラーからのリカバリにテクニカルサポートの支援が必要な場合は、データベース・サーバおよびデータベースをどのようにして起動しているかを担当者に伝える必要があります。

11. サーバ・コンピュータでウイルス・スキャナまたはスパイウェア防止ソフトウェアを実行しているかどうか。実行している場合は、その製品名とバージョン

ウイルス・スキャナおよびスパイウェア防止ソフトウェアは、データベースとデータベース・サーバの通常の処理を妨害することがあります。ウイルス・スキャンとスパイウェア防止ソフトウェアの実行リストで、データベース・ファイルとトランザクション・ログを指定する必要があります。

12. データベースのバックアップ方法。データベースの最新バックアップがあるかどうか

データベースとトランザクション・ログのバックアップ方法がわかれば、アサーション・エラーからの最適なリカバリ方法の特定に役立ちます。データベースのリカバリまたはサルベージにテクニカルサポートの支援が必要な場合は、この情報があれば時間を節約できます。

13. データベースのページ・サイズ

14. アサーション・エラーの発生時に実行されていた処理がわかる場合は、その内容（データベースの起動、異常終了後のリカバリ、クエリ入力など）

15. *ASSERT.DMP* ファイルがデータベースと同一のディレクトリにあるかどうか。*SA_DUMP.DMP* ファイルが、オペレーティング・システムの一時的フォルダまたはデータベース・ファイルと同一のディレクトリに作成されているかどうか（Adaptive Server Anywhere 9.0.2.3124 以降）

ASSERT.DMP ファイルは、データベースのアサーション・エラー発生時に作成され、データベース・ファイルと同一のディレクトリに保存されます。このファイルにはアサーションに関する情報が記録されており、テクニカルサポートが原因を特定する場合に使用できます。*ASSERT.DMP* ファイルが作成されている場合は、通常はデータベース破損が発生しています。すぐにデータベースを検証することをお勧めします。

Adaptive Server Anywhere 9.0.2.3124 以降を使用している場合は、*SA_DUMP.DMP* ファイルは、重大なエラー、アサーション・エラー、一般保護違反の発生時に作成されます。このファイルは、エンジン・スレッドのスタック・トレースを提供するため、予期しない状態の原因をデバッグするために利用できます。

16. データベースがレプリケーションまたは同期に関係しているかどうか（SQL Remote、Replication Server、Mobile Link）

レプリケーションまたは同期が実行される環境でリカバリを実行する場合は、トランザクション・ログとトランザクション・ログのオフセットの扱いに特別な注意が必要です。テクニカルサポートへお問い合わせ頂く場合は、環境がレプリケーションまたは同期に関係するかどうかを担当者に伝える必要があります。