

Oracle Services for Microsoft Transaction Server

開発者ガイド

リリース 9.2

2002 年 7 月

部品番号 : J06334-01

ORACLE®

Oracle Services for Microsoft Transaction Server 開発者ガイド, リリース 9.2

部品番号 : J06334-01

原本名 : Oracle Services for Microsoft Transaction Server Developer's Guide, Release 9.2 for Windows

原本部品番号 : A95496-01

原本著者 : Herbert Kelly, Mark Kennedy and Tamar Rothenberg

原本協力者 : Vivek Raja and Eric Wang

グラフィック・デザイナー : Valarie Moore

Copyright © 1996, 2002, Oracle Corporation. All rights reserved.

Printed in Japan.

制限付権利の説明

プログラム（ソフトウェアおよびドキュメントを含む）の使用、複製または開示は、オラクル社との契約に記された制約条件に従うものとします。著作権、特許権およびその他の知的財産権に関する法律により保護されています。

当プログラムのリバース・エンジニアリング等は禁止されています。

このドキュメントの情報は、予告なしに変更されることがあります。オラクル社は本ドキュメントの無謬性を保証しません。

* オラクル社とは、**Oracle Corporation**（米国オラクル）または**日本オラクル株式会社**（日本オラクル）を指します。

危険な用途への使用について

オラクル社製品は、原子力、航空産業、大量輸送、医療あるいはその他の危険が伴うアプリケーションを用途として開発されておりません。オラクル社製品を上述のようなアプリケーションに使用することについての安全確保は、顧客各位の責任と費用により行ってください。万一かかる用途での使用によりクレームや損害が発生いたしましても、日本オラクル株式会社と開発元である **Oracle Corporation**（米国オラクル）およびその関連会社は一切責任を負いかねます。当プログラムを米国国防総省の米国政府機関に提供する際には、『**Restricted Rights**』と共に提供してください。この場合次の **Notice** が適用されます。

Restricted Rights Notice

Programs delivered subject to the DOD FAR Supplement are "commercial computer software" and use, duplication, and disclosure of the Programs, including documentation, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement. Otherwise, Programs delivered subject to the Federal Acquisition Regulations are "restricted computer software" and use, duplication, and disclosure of the Programs shall be subject to the restrictions in FAR 52.227-19, Commercial Computer Software - Restricted Rights (June, 1987). Oracle Corporation, 500 Oracle Parkway, Redwood City, CA 94065.

このドキュメントに記載されているその他の会社名および製品名は、あくまでその製品および会社を識別する目的にのみ使用されており、それぞれの所有者の商標または登録商標です。

目次

はじめに	v
対象読者	vi
このマニュアルの構成	vi
関連文書	vii
表記規則	viii
 Oracle Services for Microsoft Transaction Server の新機能	xiii
Oracle9i リリース 1 (9.0.1) の Oracle Services for Microsoft Transaction Server における新機能	xiv
 1 Oracle と Microsoft Transaction Server の連携	
Microsoft Transaction Server の概要	1-2
Microsoft Transaction Server と Oracle の統合の概要	1-2
Microsoft Transaction Server および Oracle に関する参照先	1-5
 2 Oracle 製品のインストールおよび移行	
Microsoft Transaction Server および Oracle のインストールの概要	2-2
Oracle Services for Microsoft Transaction Server のインストール要件	2-3
Oracle Services for Microsoft Transaction Server の以前のリリースからの移行	2-5
Oracle Manager for MTS Services スナップインを使用した Oracle Services の削除	2-6
Oracle Services の非アクティブ・ユーザーのロールおよび権限の削除	2-10
レジストリを使用した Oracle Services の手動削除	2-11
タスク 1: レジストリを使用した Oracle Services の手動削除	2-12
タスク 2: OracleMTSServicen サービスの削除	2-14

3 リカバリのシナリオ管理

Microsoft Transaction Server の構成要件	3-2
Microsoft Transaction Server トランザクション・リカバリの概要	3-3
Microsoft Transaction Server 自動トランザクション・リカバリのスケジューリング	3-4
タスク 1: SNP プロセスの設定および起動	3-4
タスク 2: 自動トランザクション・リカバリの作成およびスケジューリング	3-6
utl_oramts.show_indoubt プロシージャ	3-7
utl_oramts.recover_automatic プロシージャ	3-8
utl_oramts.forget_RMs プロシージャ	3-9
oramts_2pc_pending ビュー	3-9
Microsoft Transaction Server インダウト・トランザクションの表示	3-10
Oracle Fail Safe 構成用のレジストリ値の変更	3-11

4 Microsoft アプリケーション・デモの実行

Microsoft アプリケーション・デモでの OCI の構成	4-2
Microsoft アプリケーション・デモの概要	4-2
データベースに Microsoft アプリケーション・デモの適切な表が含まれていることの確認	4-3
Microsoft アプリケーション・デモの実行	4-5
Microsoft アプリケーション・デモでの Oracle ODBC ドライバの構成	4-7
Microsoft アプリケーション・デモでの Oracle Provider for OLE DB の構成	4-7

5 Microsoft Transaction Server および Oracle データベース・サーバーを使用したプログラミング

トランザクション内での COM コンポーネント統合の概要	5-2
Microsoft Transaction Server アプリケーションの開発の概要	5-5
Microsoft Transaction Server コンポーネントの登録の概要	5-5
Microsoft Transaction Server によって調整されるコンポーネント・トランザクションの概要	5-7
MS DTC によって調整されるコンポーネント・トランザクションの概要	5-8
OCI と Microsoft Transaction Server の統合の概要	5-9
OCI および Microsoft Transaction Server の関数の概要	5-9
OraMTSSvcGet() 関数	5-13
OraMTSSvcRel() 関数	5-16
OraMTSSvcEnlist() 関数	5-17
OraMTSSvcEnlistEx() 関数	5-18
OraMTSEnlCtxGet() 関数	5-20

OraMTSEnlCtxRel() 関数	5-21
OraMTSJoinTxn() 関数	5-22
OraMTSTransTest() 関数	5-23
OraMTSOCIErrGet() 関数	5-23
ODBC と Microsoft Transaction Server の統合の概要	5-25
接続属性の設定	5-25
Oracle ODBC ドライバの使用	5-25
Microsoft 社製 Oracle 用 ODBC ドライバの使用	5-27
OO4O と Microsoft Transaction Server の統合の概要	5-28
Oracle Provider for OLE DB と Microsoft Transaction Server の統合の概要	5-28
その他の API と Microsoft Transaction Server の統合の概要	5-28

6 Microsoft Transaction Server パフォーマンス・チューニング

Microsoft Transaction Server アプリケーションのパフォーマンス向上	6-2
Microsoft Transaction Server 接続の管理	6-3
Windows NT でのトランザクション・タイムアウト・パラメータ値の増加	6-6
初期化パラメータ・ファイルの設定の変更	6-8
MS DTC の開始	6-9

7 Oracle Services for Microsoft Transaction Server のトラブルシューティング

Oracle Services for Microsoft Transaction Server のパフォーマンスの追跡	7-2
Windows NT エクスプローラの問題への対処	7-4
接続プールに影響する Oracle Net の変更の修正	7-4
Oracle Services for Microsoft Transaction Server に関するよくある質問	7-5
Microsoft Transaction Server 管理者ユーザー・アカウントの削除	7-8

A Windows 2000 での Oracle Services for Microsoft Transaction Server の使用

Windows NT と Windows 2000 での Microsoft Transaction Server の違い	A-2
Windows 2000 でのトランザクション・タイムアウト・パラメータ値の増加	A-3

用語集

索引

はじめに

このマニュアルでは、Windows NT および Windows 2000 のオペレーティング・システムに適用される Oracle Services for [Microsoft Transaction Server](#) を使用するための概要、インストール、構成、使用方法および管理情報について説明します。

このマニュアルで説明するのは、Windows NT、Windows 2000、Windows XP および Windows 98 オペレーティング・システムに適用される Oracle9i for Windows ソフトウェアの機能のみです。

次の項目について説明します。

- [対象読者](#)
- [このマニュアルの構成](#)
- [関連文書](#)
- [表記規則](#)

対象読者

このマニュアルは、次の作業を行うアプリケーション開発者を対象としています。

- Microsoft Transaction Server で **Component Object Model (COM)** コンポーネントを使用
- COM コンポーネントをトランザクションとして登録し、Microsoft Transaction Server でトランザクションを制御
- Microsoft Transaction Server でクライアント側の接続プーリングを使用

このマニュアルの構成

このマニュアルは、次のように構成されています。

第 1 章「Oracle と Microsoft Transaction Server の連携」

この章では、Microsoft Transaction Server と Oracle データベース・サーバーの統合について説明します。

第 2 章「Oracle 製品のインストールおよび移行」

この章では、Microsoft Transaction Server とデータベース・サーバーの環境におけるインストールおよび移行の要件について説明します。

第 3 章「リカバリのシナリオ管理」

この章では、Microsoft Transaction Server トランザクション・リカバリのスケジューリング方法について説明します。

第 4 章「Microsoft アプリケーション・デモの実行」

この章では、Microsoft Transaction Server に組み込まれている、サンプルの Microsoft COM ベースのアプリケーション・デモを使用する方法について説明します。

第 5 章「Microsoft Transaction Server および Oracle データベース・サーバーを使用したプログラミング」

この章では、Microsoft Transaction Server およびデータベース・サーバーを使用したプログラミング方法について説明します。

第 6 章「Microsoft Transaction Server パフォーマンス・チューニング」

この章では、Microsoft Transaction Server のパフォーマンスをチューニングする方法について説明します。

第7章「Oracle Services for Microsoft Transaction Server のトラブルシューティング」

この章では、Oracle Services for Microsoft Transaction Server に関するトラブルシューティング方法について説明します。

付録 A「Windows 2000 での Oracle Services for Microsoft Transaction Server の使用」

この付録では、Oracle Services for Microsoft Transaction Server を Windows NT で使用した場合と Windows 2000 で使用した場合の違いについて説明します。

用語集

関連文書

詳細は、次の Oracle リソースを参照してください。

- Oracle ドキュメント・セット。特に次のドキュメントを参照してください。
 - 『Oracle9i Database for Windows インストレーション・ガイド』
 - 『Oracle9i Database for Windows スタート・ガイド』
 - 『Oracle9i データベース・リファレンス』
 - 『Oracle Provider for OLE DB 開発者ガイド』
 - 「Oracle Objects for OLE オンライン・ヘルプ」
 - 『Oracle9i Net Services 管理者ガイド』

このマニュアルの多くの例では、Oracle のインストール時にデフォルトでインストールされるシード・データベースのサンプル・スキーマを使用します。これらのスキーマの作成方法およびその使用方法については、『Oracle9i サンプル・スキーマ』を参照してください。

リリース・ノート、インストール・ドキュメント、ホワイト・ペーパー、またはその他の関連資料を無償でダウンロードするには、OTN-J (Oracle Technology Network Japan) にアクセスしてください。OTN-J を利用する前に、オンライン登録が必要です。次の URL で登録できます。

<http://otn.oracle.co.jp/membership/>

OTN-J のユーザー名およびパスワードをすでにお持ちの場合は、次の OTN-J の Web サイトのドキュメント・セクションに直接アクセスできます。

<http://otn.oracle.co.jp/document/>

表記規則

ここでは、このマニュアルの本文およびサンプル・コードで使用される表記規則について説明します。表記規則は次の3種類です。

- 本文の表記規則
- サンプル・コードの表記規則
- Windows オペレーティング・システムの表記規則

本文の表記規則

本文中では、特定の用語をより簡単に識別できるように、様々な表記規則を使用しています。次の表は、本文中で使用される表記規則とその使用例を説明したものです。

規則	意味	例
太字	太字は、本文中で定義されている用語、または用語集で説明されている用語、あるいはその両方を示します。	この句を指定する場合、 索引構成表 を作成します。
大文字（固定幅） フォント	大文字固定幅フォントは、システムによって指定される要素を示します。これらの要素には、パラメータ、権限、データ型、Recovery Manager のキーワード、SQL のキーワード、SQL*Plus またはユーティリティのコマンド、パッケージ、メソッドの他に、システムで表示される列名、データベースのオブジェクトおよび構造、ユーザー名およびロールがあります。	この句は NUMBER 列に対してのみ指定できます。 BACKUP コマンドを使用して、データベースをバックアップできます。 USER_TABLES データ・ディクショナリ・ビューの TABLE_NAME 列を問い合わせます。 DBMS_STATS.GENERATE_STATS プロシージャを使用します。
小文字（固定幅） フォント	小文字固定幅フォントは、実行可能ファイル、ファイル名、ディレクトリ名、およびサンプルのユーザー指定要素を示します。これらの要素には、コンピュータ名およびデータベース名、ネット・サービス名、および接続識別子の他に、ユーザー指定のデータベースのオブジェクトおよび構造、列名、パッケージおよびクラス、ユーザー名およびロール、プログラム・ユニット、およびパラメータ値があります。 注意： 一部のプログラム要素には、大文字と小文字の両方が使用されます。これらの要素は、記載されているとおりに入力してください。	sqlplus を入力して、SQL*Plus を開きます。 パスワードは、orapwd ファイルで指定されます。 ¥disk1¥oracle¥dbs ディレクトリのデータ・ファイルと制御ファイルをバックアップします。 department_id、department_name および location_id 列は、hr.departments 表にあります。 QUERY_REWRITE_ENABLED 初期化パラメータを true に設定します。 oe ユーザーとして接続します。 JRepUtil クラスは、これらのメソッドを実装します。

規則	意味	例
小文字イタリック (固定幅) フォント	小文字イタリック固定幅フォントは、プ レースホルダまたは変数を示します。	<i>parallel_clause</i> を指定できます。 <i>Uold_release</i> .SQL を実行します。 <i>old_release</i> は、アップグレード前にインス トールしたリリースを表します。

サンプル・コードの表記規則

サンプル・コードは、SQL、PL/SQL、SQL*Plus またはその他のコマンドライン文を示しま
す。これらは固定幅フォントで示され、次の例のように、通常の本文とは区別されていま
す。

```
SELECT username FROM dba_users WHERE username = 'MIGRATE';
```

次の表は、サンプル・コードで使用する表記規則とそれらの使用例を説明したものです。

規則	意味	例
[]	大カッコは、1 つ以上のオプション項目を囲 みます。大カッコは入力しないでください。	DECIMAL (<i>digits</i> [, <i>precision</i>])
{ }	中カッコは複数の項目を囲み、そのうちの 1 つが必要であることを示します。中カッコ は入力しないでください。	{ENABLE DISABLE}
	縦線は、大カッコまたは中カッコ内にある 複数のオプションの選択肢を区切るために 使用します。オプションの 1 つを入力しま す。縦線は入力しないでください。	{ENABLE DISABLE} [COMPRESS NOCOMPRESS]
...	水平の省略記号は、次のいずれかを示しま す。 <ul style="list-style-type: none"> ■ 例に直接関係のないコードの一部を省 略 ■ コードの一部の繰り返しが可能 	CREATE TABLE ... AS <i>subquery</i> ; SELECT <i>col1</i> , <i>col2</i> , ... , <i>coln</i> FROM employees;

規則	意味	例
.	垂直の省略記号は、例に直接関係のないコードの数行を省略したことを示します。	SQL> SELECT NAME FROM V\$DATAFILE; NAME ----- /fsl/dbs/tbs_01.dbf /fsl/dbs/tbs_02.dbf . . . /fsl/dbs/tbs_09.dbf 9 rows selected.
その他の表記規則	大カッコ、中カッコ、縦線および省略記号以外の記号は、示されているとおりに入力してください。	acctbal NUMBER(11,2); acct CONSTANT NUMBER(4) := 3;
イタリック	イタリックの文字は、特定の値を指定する必要があるプレースホルダまたは変数を示します。	CONNECT SYSTEM/system_password DB_NAME = database_name
大文字	大文字は、システムによって指定される要素を示します。ユーザーが定義する語句と区別するために、大文字で示しています。語句が大カッコ内に表示されている場合を除き、記載されているとおりの順序とスペルで入力します。ただし、これらの語句には大文字と小文字の区別がないため、小文字で入力できます。	SELECT last_name, employee_id FROM employees; SELECT * FROM USER_TABLES; DROP TABLE hr.employees;
小文字	小文字は、ユーザーが指定するプログラム要素を示します。たとえば、小文字は表、列またはファイルの名前を示します。 注意： 一部のプログラム要素には、大文字と小文字の両方が使用されます。これらの要素は、記載されているとおりに入力してください。	SELECT last_name, employee_id FROM employees; sqlplus hr/hr CREATE USER mjones IDENTIFIED BY ty3MU9;

Windows オペレーティング・システムの表記規則

次の表は、Windows オペレーティング・システムの表記規則とその使用例を説明したものです。

規則	意味	例
「スタート」→を選択	プログラムの起動方法。たとえば、Oracle Database Configuration Assistant を起動するには、タスクバーの「スタート」ボタンをクリックし、「プログラム」→「Oracle - HOME_NAME」→「Configuration and Migration Tools」→「Database Configuration Assistant」を選択します。	「スタート」→「プログラム」→「Oracle - HOME_NAME」→「Configuration and Migration Tools」→「Database Configuration Assistant」を選択します。
ファイル名およびディレクトリ名	ファイルおよびディレクトリ名には、大文字と小文字の区別がありません。＜、＞、：、"、/、 、および- の特殊文字は使用できません。特殊文字 ¥ は、引用符に囲まれている場合でも、要素の区切り文字として扱われます。ファイル名が ¥¥ で始まる場合、Windows では汎用命名規則を使用しているものと認識されます。	c:¥winnt"¥"system32 は、C:¥WINNT¥SYSTEM32 と同じです。
C:¥>	現行のハード・ディスク・ドライブの Windows コマンド・プロンプトを示します。コマンド・プロンプトのエスケープ文字は、カレット (^) です。プロンプトは、現在作業中のサブディレクトリを示しています。このマニュアルでは、コマンド・プロンプトと呼びます。	C:¥oracle¥oradata>
特殊文字	特殊文字の円記号 (¥) は、Windows コマンド・プロンプトで特殊文字の二重引用符 (") のエスケープ文字として必要な場合があります。カッコおよび特殊文字の一重引用符 (') は、エスケープ文字を必要としません。エスケープ文字および特殊文字の詳細は、Windows オペレーティング・システムのドキュメントを参照してください。	C:¥>exp scott/tiger TABLES=emp QUERY=¥"WHERE job='SALESMAN' and sal<1600¥" C:¥>imp SYSTEM/password FROMUSER=scott TABLES=(emp, dept)
HOME_NAME	Oracle ホーム名を示します。 ホーム名は、英数字 16 文字までです。ホーム名で利用できる特殊文字は、アンダースコアのみです。	C:¥> net start OracleHOME_NAME_TNSListener

規則	意味	例
ORACLE_HOME および ORACLE_BASE	<p>Oracle8 リリース 8.0 以下のリリースでは、Oracle コンポーネントをインストールすると、サブディレクトリはすべて、最上位の ORACLE_HOME ディレクトリ（デフォルトでは次のとおり）の下に置かれました。</p> <ul style="list-style-type: none">■ Windows NT の場合は C:¥orant■ Windows 98 の場合は C:¥orawin98 <p>あるいは、Oracle ホームと呼ばれるディレクトリの下に置かれました。</p> <p>今回のリリースは、Optimal Flexible Architecture (OFA) ガイドラインに準拠しています。すべてのサブディレクトリが最上位の ORACLE_HOME ディレクトリの下にあるわけではありません。ORACLE_BASE という最上位ディレクトリがあり、デフォルトは C:¥oracle です。コンピュータに Oracle9i リリース 2 (9.2) をインストールし、他の Oracle ソフトウェアをインストールしない場合、最初の Oracle ホーム・ディレクトリのデフォルト設定は、C:¥oracle¥orann です。nn は最新のリリース番号です。Oracle ホーム・ディレクトリは、ORACLE_BASE の直下に置かれます。</p> <p>このマニュアルでは、ディレクトリ・パスの例は、すべて OFA 表記規則に準拠しています。</p>	%ORACLE_HOME%¥rdbms¥admin ディレクトリに移動します。

Oracle Services for Microsoft Transaction Server の新機能

この製品は、Oracle9i リリース 2 (9.2) では新機能が追加されていません。多くの新機能は、Oracle9i リリース 1 (9.0.1) で追加されました。リリース 8.1 以下からアップグレードする場合、「[Oracle9i リリース 1 \(9.0.1\) の Oracle Services for Microsoft Transaction Server における新機能](#)」の新機能情報をお読みください。

次の項で、Oracle Services for Microsoft Transaction Server の新機能について説明します。

- [Oracle9i リリース 1 \(9.0.1\) の Oracle Services for Microsoft Transaction Server における新機能](#)

Oracle9i リリース 1 (9.0.1) の Oracle Services for Microsoft Transaction Server における新機能

リリース 8.1 以下のリリースの Oracle Services for Microsoft Transaction Server では、各 Oracle データベース・サーバー用に **Oracle Service for MTS** という Windows NT サービスを作成する必要がありました。これにより、データベースの **Microsoft Transaction Server** トランザクションへの登録が可能になりました。しかし、Oracle Service for MTS はデータベース・サーバーごとに 1 つしかサポートされませんでした。このリリースでは、Oracle Service for MTS は必要ありません。データベース・サーバーを Microsoft Transaction Server トランザクションに登録するためのコードおよびロジックはすべて、Microsoft Transaction Server アプリケーションの処理に埋め込みます。

その結果、このリリースでは次の機能が提供されます。

- パフォーマンスの向上

Microsoft Transaction Server アプリケーションと Oracle Service for MTS 間の通信が不要になります。

- 高可用性

データベース・サーバーが Oracle Service for MTS に依存しなくなります。これまでは、Oracle Service for MTS が停止している場合、データベース・サーバーを Microsoft Transaction Server トランザクションに登録できませんでした。

- 拡張性の改善

データベース・サーバーを Microsoft Transaction Server トランザクションに登録するためのコードは、Microsoft Transaction Server の各アプリケーションの処理に埋め込みます。

- 必要な構成の簡素化

Oracle MTS Recovery Service は、Oracle Services for Microsoft Transaction Server のインストール時に、中間層の各コンピュータで自動的に作成および構成されます。

関連項目： 1-4 ページの [図 1-1 「Microsoft Transaction Server とデータベース・サーバーの統合」](#)

- API の変更

Microsoft Transaction Server によって開始されたトランザクション内で Oracle 接続の登録を開始する Call Level Interface (CLI) にも変更が加えられました。OraMTSSvcEnlist() および OraMTSSvcEnlistEx() は、下位互換性を保つ目的でのみ提供されています。さらに、次の 3 つの新規 Application Program Interface (API) があります。

- OraMTSEnlCtxGet()
- OraMTSEnlCtxRel()
- OraMTSJJoinTxn()

関連項目： 5-9 ページの「[OCI と Microsoft Transaction Server の統合の概要](#)」

注意： [Microsoft 管理コンソール](#)用の [Oracle Manager for MTS Services スナップイン](#)はなくなりました。このスナップインは、Oracle8i リリース 8.1 以下のリリースで Oracle Service for MTS を作成するために必要でした。

Oracle と Microsoft Transaction Server の 連携

この章では、[Microsoft Transaction Server](#) と Oracle データベース・サーバーの統合について説明します。

この章の項目は次のとおりです。

- [Microsoft Transaction Server の概要](#)
- [Microsoft Transaction Server と Oracle の統合の概要](#)
- [Microsoft Transaction Server および Oracle に関する参照先](#)

Microsoft Transaction Server の概要

Microsoft Transaction Server は、インターネットまたはネットワーク・サーバー上で動作する、独自規格の **Component Object Model (COM)** トランザクション処理システムです。Microsoft Transaction Server は、アプリケーションおよびデータベース・トランザクションの要求をクライアント・コンピュータに代わって配置および管理します。Microsoft Transaction Server によって、次の機能が提供されます。

- 分散アプリケーション開発用の ActiveX/**Distributed Component Object Model (DCOM)** プログラム・モデル、および開発したアプリケーションを配置するランタイム環境
- トランザクション内のコンポーネントの**原子性、一貫性、独立性および永続性 (ACID)** プロパティ
- コンポーネント・キャッシュおよびデータベースの接続プーリングなどのパフォーマンス強化機能へのアクセス

Microsoft Transaction Server は、サーバー中心のアーキテクチャ・モデルによる 3 層のコンポーネントです。このため、アプリケーションのプレゼンテーション、ビジネス・ロジックおよびデータ要素を明確に分離して、ネットワークに接続された別々のコンピュータに配置できます。

関連資料： Microsoft Transaction Server の追加情報は、Microsoft のドキュメントを参照してください。

Microsoft Transaction Server と Oracle の統合の概要

特別な統合を行わなくても、リリース 8.0.6 以上の Oracle データベース・サーバーに接続された Microsoft Transaction Server に COM コンポーネントを配置できます。ただし、次のいずれかの機能を使用する場合は、**Oracle Services for Microsoft Transaction Server (MTS)** をインストールする必要があります。

- COM コンポーネントをトランザクションとして登録 (**Microsoft 管理コンソール**でコンポーネントの「プロパティ」ダイアログ・ボックスを使用) し、Microsoft Transaction Server でトランザクションを制御
- Microsoft Transaction Server でクライアント側の接続プーリングを使用

Oracle Services for Microsoft Transaction Server がインストールされた後、**Oracle MTS Recovery Service** も同じコンピュータに自動的にインストールされます。Oracle MTS Recovery Service は、そのコンピュータで開始された Microsoft Transaction Server 連携の Oracle トランザクションを調整します。COM コンポーネントが接続している各データベース・サーバーで、ユーザーは次のことを実行します。

- Microsoft Transaction Server 管理者ユーザー・アカウントの作成
- データベース・サーバー・レベルでのトランザクション・リカバリ・ジョブのスケジューリング

これにより、データベース・サーバーを Microsoft Transaction Server で開始されたトランザクションに登録することが可能になります。

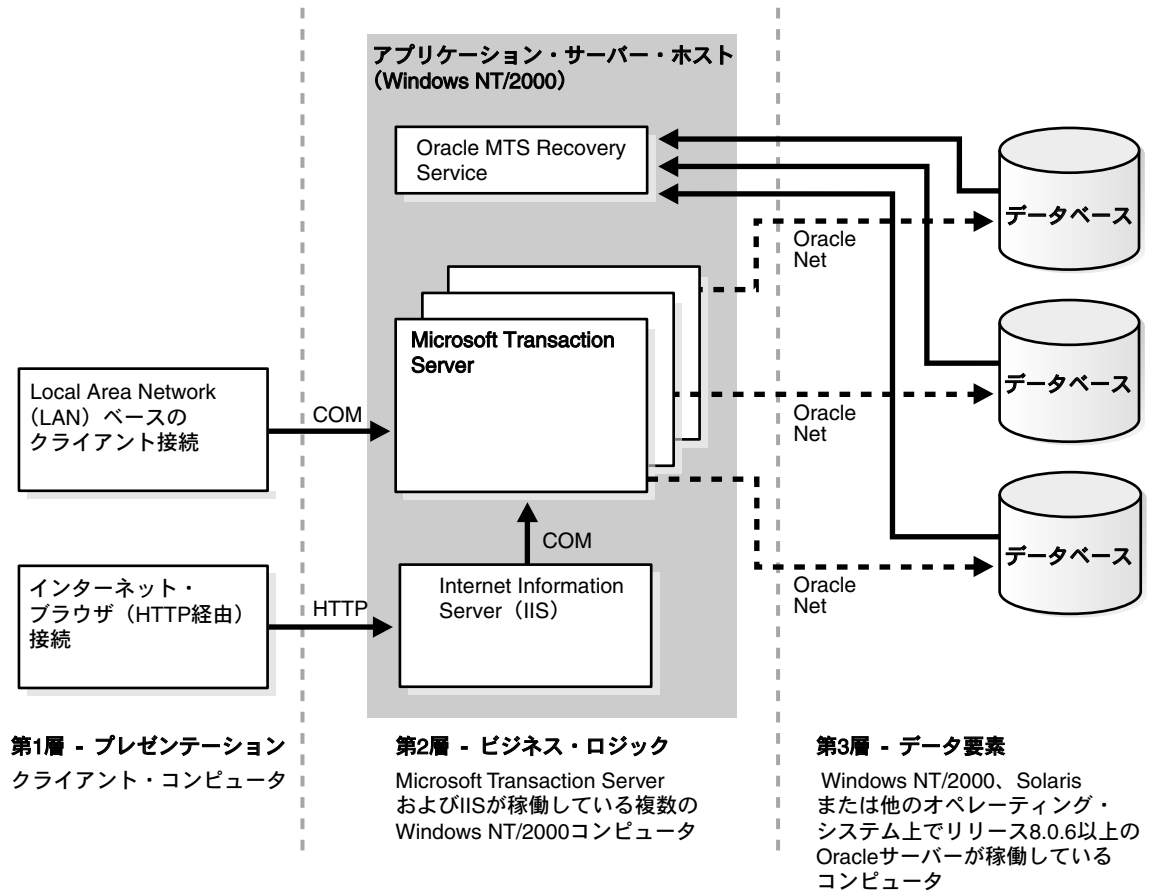
次のいずれかの Oracle 製品を使用して COM コンポーネントを作成します。

- **Oracle Call Interface (OCI)**
- **Oracle Objects for OLE (OO4O)**
- **Oracle ODBC ドライバ**
- **Oracle Provider for OLE DB**

これ以外の API も統合される予定です。

図 1-1 は、Microsoft Transaction Server およびデータベース・サーバーがどのようにしてサーバー中心の 3 層のモデルに統合されているかを示しています。

図 1-1 Microsoft Transaction Server とデータベース・サーバーの統合



Microsoft Transaction Server および Oracle に関する参照先

これで、データベース・サーバーで Microsoft Transaction Server を使用する準備が整いました。すぐ開始するには、表 1-1 に記載した各章を示した順序で読み進めてください。

表 1-1 Microsoft Transaction Server および Oracle に関する参照先

実行するタスク	参照先
<ul style="list-style-type: none"> Microsoft Transaction Server とデータベース・サーバーの統合に必要な Oracle 製品および Microsoft 製品のインストール 以前のリリースの Oracle Services for Microsoft Transaction Server からの移行 	第 2 章「Oracle 製品のインストールおよび移行」
<ul style="list-style-type: none"> Microsoft Transaction Server 管理者ユーザー・アカウントの作成 Microsoft Transaction Server トランザクション・リカバリ・ジョブのスケジューリング 	第 3 章「リカバリのシナリオ管理」
Microsoft アプリケーション・デモの実行	<p>次の処理を行うアプリケーション・デモの実行については、第 4 章「Microsoft アプリケーション・デモの実行」を参照してください。</p> <ul style="list-style-type: none"> Microsoft Transaction Server をホストとするトランザクション COM コンポーネントの使用 Microsoft Transaction Server によって制御されるトランザクション内のデータベース・サーバーへのアクセス
Microsoft Transaction Server をホストとする COM アプリケーションの作成	COM ベースのアプリケーションでの、OCI、OO4O、Oracle ODBC ドライバまたは Oracle Provider for OLE DB の使用方法については、第 5 章「Microsoft Transaction Server および Oracle データベース・サーバーを使用したプログラミング」を参照してください。
Microsoft Transaction Server を Windows NT で使用した場合と Windows 2000 で使用した場合の違いの確認	付録 A「Windows 2000 での Oracle Services for Microsoft Transaction Server の使用」

Oracle 製品のインストールおよび移行

この章では、**Microsoft Transaction Server** および Oracle データベース・サーバーの環境におけるインストールおよび移行の要件について説明します。

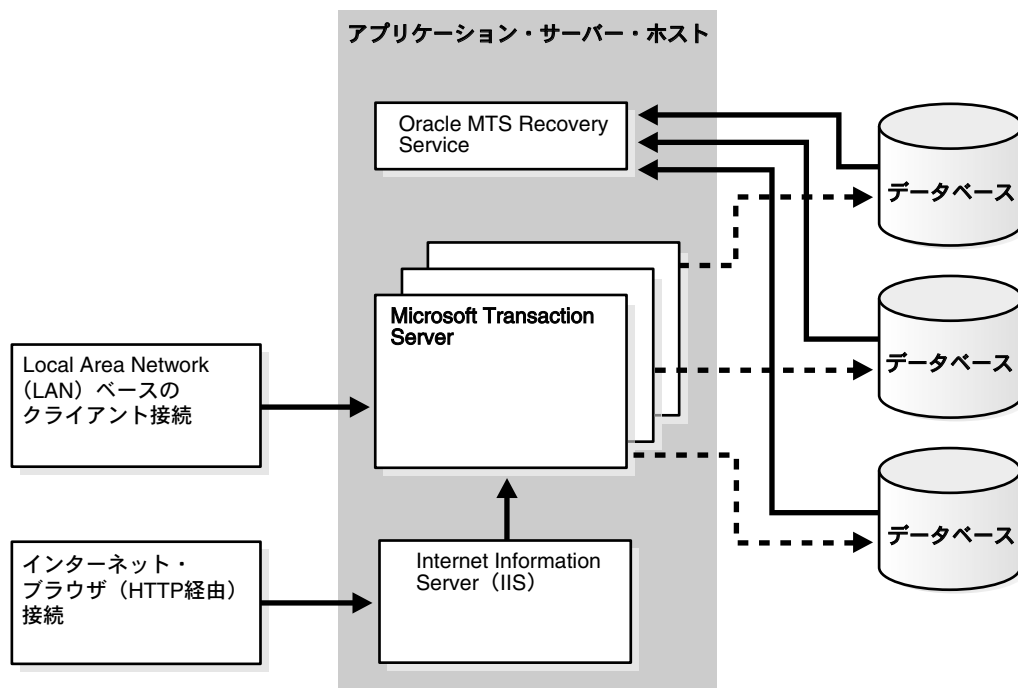
この章の項目は次のとおりです。

- **Microsoft Transaction Server および Oracle のインストールの概要**
- **Oracle Services for Microsoft Transaction Server のインストール要件**
- **Oracle Services for Microsoft Transaction Server の以前のリリースからの移行**
- **レジストリを使用した Oracle Services の手動削除**

Microsoft Transaction Server および Oracle のインストールの概要

図 2-1 は、Microsoft Transaction Server および複数のデータベース・サーバーの一般的なインストール・レイアウトを示しています。

図 2-1 Microsoft Transaction Server および複数のデータベース・サーバーのインストール・レイアウト



Oracle Services for Microsoft Transaction Server のインストール要件

表 2-1 は、インストールが必要な Oracle 製品および Oracle 以外の製品をまとめたものです。インストール要件を確認した後、必要な Oracle 製品をインストールする方法については、『Oracle9i Database for Windows インストレーション・ガイド』を参照してください。主なガイドラインは次のとおりです。

- データベース・サーバー、Microsoft Transaction Server および **Oracle Services for Microsoft Transaction Server (MTS)** を 1 台のコンピュータにインストールする場合は、最低 256MB の RAM が必要です。
- Oracle Services for Microsoft Transaction Server と同じ中間層のコンピュータに、**Oracle MTS Recovery Service** も自動的にインストールされます。1 台のコンピュータにインストール可能な Oracle MTS Recovery Service は 1 つのみです。
- **Oracle Fail Safe** クラスタ構成（または任意のフェイルオーバー構成）の場合、**Microsoft 分散トランザクション・コーディネータ (MS DTC)** コンポーネントが稼働しているノードに Microsoft Transaction Server をインストールします。これにより、フェイルオーバー中に Oracle MTS Recovery Service がクライアント・アプリケーションとともに移行されます。これは、リカバリ・トランザクションのスケジューリング時に構成できます。

注意： 現在、Microsoft Transaction Server とデータベース・サーバーの統合は、Windows 98 ではサポートされていません。

表 2-1 インストール要件

コンピュータのタイプ	Oracle 製品	Oracle 以外の製品
クライアント・コンピュータ	<ul style="list-style-type: none">■ 不要	<ul style="list-style-type: none">■ 正常に機能するネットワーク・プロトコル・ソフトウェア■ Web ブラウザ
Microsoft Transaction Server がインストールされている Windows NT コンピュータ	<ul style="list-style-type: none">■ Oracle Services for Microsoft Transaction Server¹■ クライアント用の Oracle Net Services²■ Oracle Objects for OLE (OO4O)³■ Oracle ODBC ドライバ³■ Oracle Provider for OLE DB³■ Oracle Call Interface (OCI)³■ Oracle Net Manager■ SQL*Plus	<ul style="list-style-type: none">■ Microsoft Internet Information Server (IIS)⁴■ Microsoft Transaction Server バージョン 2.0⁵■ Windows NT 4.0 Service Pack 5 以上⁶■ 正常に機能するネットワーク・プロトコル・ソフトウェア
データベース・サーバーがインストールされているコンピュータ	<ul style="list-style-type: none">■ Oracle サーバー (データベース・サーバー)⁷■ サーバー用の Oracle Net Services■ SQL*Plus	<ul style="list-style-type: none">■ 正常に機能するネットワーク・プロトコル・ソフトウェア

¹ インストール手順は、『Oracle9i Database for Windows インストレーション・ガイド』を参照してください。インストール中に、Oracle MTS Recovery Service がインダウト・トランザクションの解決のための要求をリスニングするポート番号を入力するよう求められます。

² Oracle Net Services は、Oracle Services for Microsoft Transaction Server とともに自動的にインストールされます。

³ リリース 9.2 の OO4O、Oracle ODBC ドライバ、Oracle Provider for OLE DB および OCI が必要なのは、これらの各製品が必要とされるコンポーネントを作成または使用する場合のみです。

⁴ IIS などのアプリケーション・サーバーは必須ではありませんが、Microsoft Transaction Server を使用する場合には役立ちます。アプリケーション・サーバーは、Microsoft Transaction Server と同じコンピュータにインストールします。拡張機能をサポートするアプリケーション・サーバーを選択する必要があります。

⁵ 現在、Microsoft Transaction Server は Microsoft Windows NT 4.0 Option Pack に組み込まれています。[Microsoft 管理コンソール](#)は、Microsoft Transaction Server とともに自動的にインストールされます。

⁶ Microsoft Transaction Server 2.0 をインストールした後で、Windows NT 4.0 Service Pack 5 以上をインストールすることをお勧めします。Microsoft Transaction Server 2.0 のインストール前にインストールした Service Pack は、再インストールする必要があります。

⁷ データベース・サーバーがリリース 8.0.5 以下の場合、このマニュアルで説明する統合は利用できません。リリース 8.0.5 以下のデータベース・サーバーで利用可能な機能の説明については、5-27 ページの「[Microsoft 社製 Oracle 用 ODBC ドライバの使用](#)」を参照してください。

Oracle Services for Microsoft Transaction Server の以前のリリースからの移行

Oracle9i リリース 1 (9.0.1) から、**Oracle Service for MTS** を作成する必要がなくなりました。ただし、Oracle Services for Microsoft Transaction Server をアンインストールする前に、Microsoft 管理コンソールの **Oracle Manager for MTS Services** スナップインを使用して、既存の Oracle Services を削除する必要があります。表 2-2 は、実行する手順を示しています。

表 2-2 移行要件

移行手順	操作	参照先
リリース 8.1.x のデータベース・サーバーを現行のリリースに移行し、リリース 9.2 を新たにインストール	1. Oracle Manager for MTS Services スナップインを使用して Oracle Services を削除します。 注意： データベース・サーバーをすでに削除している場合は、Oracle Manager for MTS Services スナップインを使用して Oracle Services を削除することはできません。かわりに、レジストリからサービスを削除します。これは推奨できる方法ではありません。	2-6 ページの「 Oracle Manager for MTS Services スナップインを使用した Oracle Services の削除」 データベース・サーバーをすでに削除している場合は、2-11 ページの「 レジストリを使用した Oracle Services の手動削除」
	2. 削除した Oracle Services に関連付けられているユーザーのロールおよび権限を削除します。	2-10 ページの「 Oracle Services の非アクティブ・ユーザーのロールおよび権限の削除」
	3. Microsoft Transaction Server がインストールされている Windows NT コンピュータから Oracle Services for Microsoft Transaction Server をアンインストールします。	『Oracle9i Database for Windows インストレーション・ガイド』
	4. リリース 9.2 の OO4O、Oracle Provider for OLE DB、Oracle ODBC ドライバまたは OCI を使用して Component Object Model (COM) コンポーネントを作成する予定の場合、その製品をインストールします。	
	5. Oracle Services for Microsoft Transaction Server リリース 9.2 を 1 つの Oracle ホームにインストールします。Oracle MTS Recovery Service も自動的にインストールされます。	
	6. Microsoft Transaction Server 管理者ユーザー・アカウントを作成します。	第 3 章「 リカバリのシナリオ管理 」
	7. Microsoft Transaction Server トランザクションに登録するすべてのデータベース・サーバーに対して、Microsoft Transaction Server トランザクション・リカバリ・ジョブをスケジューリングします。	

Oracle Manager for MTS Services スナップインを使用した Oracle Services の削除

Oracle Services を削除するには、Microsoft 管理コンソールの Oracle Manager for MTS Services スナップインを使用する必要があります。それ以外の方法で（たとえばキーボードの [Del] キーを押して）Oracle Services を削除すると、データベース・サーバーのデータの一貫性が損なわれます。データの一貫性が損なわれた場合、正常に完了またはリカバリしなかったトランザクションを、データベース管理者が手動でコミットするか異常終了する必要があります。Oracle Services を削除する前に、次のタスクを実行してすべてのトランザクションを必ず解決してください。

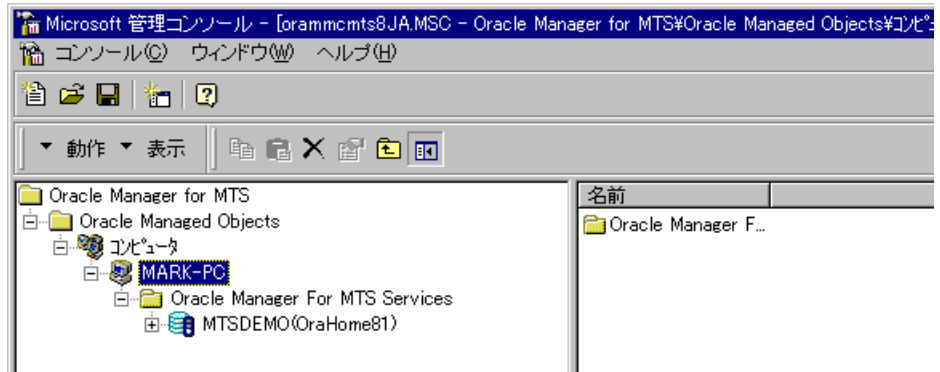
- [タスク 1: Oracle Services の停止](#)
- [タスク 2: データベース・サーバーの停止および再起動](#)
- [タスク 3: Oracle Services の再起動](#)
- [タスク 4: Oracle Services のトレース・ファイルの監視](#)
- [タスク 5: Oracle Services 表の削除](#)
- [タスク 6: Oracle Services の削除](#)

タスク 1: Oracle Services の停止

Oracle Services を停止するには、次のようにします。

1. Oracle Services を削除するコンピュータにログオンします。Oracle Services を削除する前に、操作する必要があります。Oracle Services は、このコンピュータで実行されていても、このコンピュータからアクセスできるリモート・コンピュータで実行されていてもかまいません。
2. 「スタート」 → 「プログラム」 → 「Oracle - HOME_NAME」 → 「Application Development」 → 「Oracle Manager for Microsoft Transaction Server」を選択します。
Microsoft 管理コンソールが表示されます。
3. 「エクスプローラ」ウィンドウで、削除する Oracle Services を見つけます。

- 削除する Oracle Services のアイコン（この例では「MTSDemo」）を右クリックします。



いくつかのオプションを含むメニューが表示されます。

- 「サービス停止」を選択します。

Oracle Services が停止したことを示すメッセージが表示されます。

- 「OK」をクリックします。

タスク 2: データベース・サーバーの停止および再起動

データベース・サーバーを停止してから再起動するには、次のようにします。

- データベース・サーバーが稼働しているコンピュータにログオンします。

- SQL*Plus を起動します。

```
C:\> sqlplus /NOLOG
```

- データベースに SYSDBA として接続します。

```
SQL> CONNECT / AS SYSDBA
```

- データベース・サーバーを停止します。

```
SQL> SHUTDOWN
```

- データベース・サーバーを再起動します。

```
SQL> STARTUP
```

- SQL*Plus を終了します。

```
SQL> EXIT
```

タスク 3: Oracle Services の再起動

Oracle Services を再起動するには、次のようにします。

1. Oracle Services を変更するコンピュータに戻ります。
2. 「スタート」 → 「プログラム」 → 「Oracle - HOME_NAME」 → 「Application Development」 → 「Oracle Manager for Microsoft Transaction Server」を選択します。
Microsoft 管理コンソールが表示されます。
3. 「エクスプローラ」ウィンドウで、開始する Oracle Services を見つけます。
4. Oracle Services のアイコンを右クリックします。
いくつかのオプションを含むメニューが表示されます。
5. 「サービス開始」を選択します。
Oracle Services が開始したことを示すメッセージが表示されます。
6. 「OK」をクリックします。

タスク 4: Oracle Services のトレース・ファイルの監視

Oracle Services のトレース・ファイルを監視するには、次のようにします。

1. 新しいトランザクションが Oracle Services を使用しないようにします。
2. リカバリが正常に完了したことを示すメッセージが、Oracle Services のトレース・ファイルに書き込まれるのを監視します。

2515156: [2096] OracleMTSService - Accepting new enlistment requests.

このファイルは %ORACLE_HOME%\%oramtss%\trace にあります。
3. このメッセージが表示されたら、Microsoft 管理コンソールで Oracle Services のアイコンを右クリックします。
4. 「サービス停止」を選択します。

Oracle Services が停止したことを示すメッセージが表示されます。
5. 「OK」をクリックします。

タスク 5: Oracle Services 表の削除

Oracle Services 表を削除するには、次のようにします。

1. データベース・サーバーが稼働しているコンピュータにログオンします。
2. SQL*Plus を起動します。

```
C:\> sqlplus /NOLOG
```
3. データベースに SYSDBA として接続します。

```
SQL> CONNECT / AS SYSDBA
```
4. 次のように表を削除します。

```
SQL> DROP TABLE mtsadmin_username.mts_proxy_info;
```

`mtsadmin_username` は Oracle Services のユーザー（`mtssys` など）です。

```
SQL> COMMIT;
```

タスク 6: Oracle Services の削除

Oracle Services を削除するには、次のようにします。

1. Oracle Services を削除するコンピュータにログオンします。Oracle Services は、このコンピュータで実行されていても、このコンピュータからアクセスできるリモート・コンピュータで実行されていてもかまいません。
2. 「スタート」→「プログラム」→「Oracle - HOME_NAME」→「Application Development」→「Oracle Manager for Microsoft Transaction Server」を選択します。
Microsoft 管理コンソールが表示されます。
3. 「エクスプローラ」ウィンドウで、削除する Oracle Services を見つけます。
4. Oracle Services のアイコンを右クリックします。
いくつかのオプションを含むメニューが表示されます。
5. 「削除」を選択します。
6. 表示されたメッセージに応じて、次の表に示された項に進んでください。

メッセージの内容	参照先
Oracle Services が正常に削除された。	2-10 ページの「 Oracle Services の非アクティブ・ユーザーのロールおよび権限の削除 」
Oracle Services が削除されなかった。	2-11 ページの「 レジストリを使用した Oracle Services の手動削除 」

Oracle Services の非アクティブ・ユーザーのロールおよび権限の削除

Oracle Services の不要になったユーザーや、サービスを削除したユーザーに割り当てられているロールおよび権限は、必ず削除します。

Oracle Services の非アクティブ・ユーザーのロールおよび権限を削除するには、次のようにします。

1. %ORACLE_HOME%\oramtss\admin に移動します。
2. ファイル revokeuser.sql をテキスト・エディタで開きます。
3. mts_user を、ロールおよび権限を無効にするユーザー名に置き換えます。

注意： このスクリプトでは、ユーザー名 mtssys およびパスワード mtssys を使用しています。パスワードを変更した場合や、mtssys 以外の Oracle Services ユーザー名を使用している場合は、正しいユーザー名およびパスワードに置き換える必要があります。

4. 変更内容を保存して revokeuser.sql を閉じます。
5. SQL*Plus を起動します。

```
C:¥> sqlplus /NOLOG
```

6. データベースに SYSDBA として接続します。

```
SQL> CONNECT / AS SYSDBA
```

7. 変更したスクリプトを実行します。

```
SQL> @%ORACLE_HOME%\oramtss\admin¥revokeuser.sql;
```

ユーザーのロールおよび権限が削除されます。

8. SQL*Plus を終了します。

```
SQL> EXIT
```

9. 最新リリースの Oracle Services for Microsoft Transaction Server をインストールする方法については、『Oracle9i Database for Windows インストレーション・ガイド』を参照してください。

レジストリを使用した Oracle Services の手動削除

Oracle Services を削除する前に、接続しているデータベース・サーバーへの連結を完全に解除する必要があります。この連結解除が失敗する場合があります。次の場合にのみ、この項の手順に従ってください。

- 2-6 ページの「[Oracle Manager for MTS Services](#) スナップインを使用した [Oracle Services の削除](#)」の手順が失敗した場合
- データベース・サーバーをすでに削除していて、Oracle Manager for MTS Services スナップインが使用できない場合

表 2-3 は、Microsoft 管理コンソールの Oracle Manager for MTS Services スナップインが Oracle Services の削除または変更 に失敗する 可能性のあるシナリオを説明したものです。

表 2-3 Oracle Services の削除または変更の失敗

シナリオ	対処
レジストリの情報を使用して、Oracle Manager for MTS Services スナップインからデータベース・サーバーに接続できない。	データベース・サーバーとそのリスナーが起動していることを確認します。SQL*Plus などのツールを使用して、データベース・サーバーが新しい接続を受け入れているかどうかを確認します。
データベース・サーバーの情報が、レジストリの情報と一致していない。	Oracle Manager for MTS Services スナップインが、Oracle Services の接続先とは異なるデータベース・サーバーに接続しています。Oracle Manager for MTS Services スナップインおよび Oracle Services が同じコンピュータで実行されている場合は、それらが異なる Oracle ホームにある tnsnames.ora ファイルを使用している可能性があります。異なるコンピュータで実行されている場合（たとえば、Oracle Manager for MTS Services スナップインでリモート・コンピュータ上のサービスを構成している場合）は、両者が使用している tnsnames.ora ファイルのエントリが別々のデータベースを指定しています。ローカルまたはリモートのどちらの問題であっても、これを解決するには、Oracle Manager for MTS Services スナップインおよび Oracle Services が使用する tnsnames.ora ファイルのエントリが、同じデータベース・インスタンスを指定するようにします。
データベース・サーバーに格納されたサービス情報を Oracle Manager for MTS Services スナップインから削除できない。	データベース・サーバーが不安定もしくは正常に動作していない可能性があります。データベース・プロセスの障害を示すデータベース・トレース・ファイルが作成されていないかどうか確認します。トレース・ファイルは %ORACLE_HOME%\oramtms¥trace にあります。

タスク 1: レジストリを使用した Oracle Services の手動削除

レジストリを使用して Oracle Services を手動で削除するには、次のようにします。

1. コマンド・プロンプトからレジストリ エディタを起動します。

```
C:¥> regedt32
```

「レジストリ エディタ」ウィンドウが表示されます。

2. HKEY_LOCAL_MACHINE を選択します。

System¥CurrentControlSet¥Services¥OracleMTSServicen に移動します。

n は Oracle Services の番号です。

ウィンドウの右側には、OracleMTSServicen に関連付けられた、次の表に示すような様々なパラメータおよび値が表示されます。

パラメータ	このパラメータの内容
ORAMTS_SUNAME	Oracle Services のユーザー名
ORAMTS_SUPWD	Oracle Services のユーザー名に対応するパスワード（レジストリでは暗号化される）
ORAMTS_ORADB	データベース・サーバーへの接続に使用する Oracle Services の ネット・サービス名

3. SQL*Plus を起動します。

```
C:¥> sqlplus /NOLOG
```

4. Oracle Services が接続に使用しているのと同じユーザー名およびネット・サービス名でデータベース・サーバーに接続します。

```
CONNECT as username/password@net_service_name
```

net_service_name は、データベースに接続するためのネット・サービス名です。パスワードは暗号化された形式でレジストリに格納されます。SQL*Plus で接続するときは、プレーン・テキストのパスワードを使用します。

5. 次のデータベース情報をチェックして、Oracle Services の接続先と同じデータベース・サーバーであることを確認します。

```
SQL> SELECT NAME, DBID FROM V$DATABASE;
```

次のような情報が表示されます。

```
NAME          DBID
-----
ORCL          12345678
```

6. これらの値がレジストリ値の ORAMTS_DBNAME（この例では ORCL）および ORAMTS_DBID（この例では 12345678）と一致するかどうかをチェックします。
7. サービス情報をチェックします。

```
SQL> SELECT rmguid FROM mts_proxy_info;
```

次のような情報が表示されます。

```
RMGUID
-----
2320b23e93e09fff02a231974
```

8. この情報がレジストリ値の ORAMTS_RMGUID と一致するかどうかをチェックします。
9. すべての値が一致する場合にのみ、次に進んでください。

一致しない値がある場合は、データベース・サーバーが Oracle Services の接続先と同じではありません。このまま次に進むと、データベース・サーバーにインストールされた Oracle Services で障害が発生します。これによって、データベース・サーバーの一貫性が損なわれた状態になり、データベース管理者による操作が必要になることがあります。SQL*Plus が Oracle Services の接続先と異なるデータベース・サーバーに接続したのは、tnsnames.ora ファイルが不一致なためです。

10. データベースに格納されているサービス情報を削除します。

```
SQL> DELETE FROM mts_proxy_info;
SQL> COMMIT;
```

11. SQL*Plus を終了します。

```
SQL> EXIT
```

タスク 2: OracleMTSServicen サービスの削除

OracleMTSServicen サービスを削除するには、次のようにします。

1. コンピュータを再起動します。
2. 「スタート」 → 「プログラム」 → 「Oracle - HOME_NAME」 → 「Application Development」 → 「Oracle Manager for Microsoft Transaction Server」を選択します。
Microsoft 管理コンソールが表示されます。
3. 「エクスプローラ」ウィンドウで、削除する Oracle Services を見つけます。
4. Oracle Services を右クリックします。
いくつかのオプションを含むメニューが表示されます。
5. 「削除」を選択します。
処理が正常に行われた場合は、Oracle Services が削除されたことを示すメッセージが表示されます。
6. 処理が正常に行われなかった場合は、Oracle Services が削除されなかったことを示すメッセージが表示されます。この場合、レジストリを使用してサービスのレジストリ・エントリを削除します。HKEY_LOCAL_MACHINE 以下で、次のキーを削除します。
`¥System¥CurrentControlSet¥Services¥OracleMTSServicen`
`n` は Oracle Services の番号です。
7. 2-10 ページの「[Oracle Services の非アクティブ・ユーザーのロールおよび権限の削除](#)」に進んでください。

リカバリのシナリオ管理

この章では、**Microsoft Transaction Server** 連携の Oracle トランザクション・リカバリを作成し、スケジューリングする方法について説明します。

この章の項目は次のとおりです。

- **Microsoft Transaction Server** の構成要件
- **Microsoft Transaction Server** トランザクション・リカバリの概要
- **Microsoft Transaction Server** 自動トランザクション・リカバリのスケジューリング
- **Microsoft Transaction Server** インダウト・トランザクションの表示
- **Oracle Fail Safe** 構成用のレジストリ値の変更

Microsoft Transaction Server の構成要件

Oracle Services for Microsoft Transaction Server (MTS) をインストールまたは移行した後に、Microsoft Transaction Server および Oracle データベース・サーバーの環境を構成する必要があります。表 3-1 で構成要件を確認してください。

表 3-1 構成要件

コンピュータのタイプ	Oracle の構成が必要かどうか	参照先
クライアント・コンピュータ	不要。	なし
Microsoft Transaction Server がインストールされている Windows NT コンピュータ	不要。	なし
データベース・サーバーがインストールされているコンピュータ	<p>必要。次の手順を実行します。</p> <ol style="list-style-type: none">データベース・サーバーに対して <code>oramtsadmin.sql</code> スクリプトを実行し、次のことを行います。 Microsoft Transaction Server 管理者ユーザー・アカウントを作成します（デフォルトのユーザー名は <code>mtssys</code> です）。 自動トランザクション・リカバリをスケジューリングします。Oracle Fail Safe の構成では、<code>oramtsadmin.sql</code> スクリプトの実行前または実行後にレジストリ値を変更します。	<p>3-3 ページの「Microsoft Transaction Server トランザクション・リカバリの概要」</p> <p>3-4 ページの「Microsoft Transaction Server 自動トランザクション・リカバリのスケジューリング」</p> <p>Oracle Fail Safe を使用している場合、3-11 ページの「Oracle Fail Safe 構成用のレジストリ値の変更」</p>

関連資料： Oracle 以外の製品（Microsoft Internet Information Server など）に必要な構成手順については、各製品のドキュメントを参照してください。

Microsoft Transaction Server トランザクション・リカバリの概要

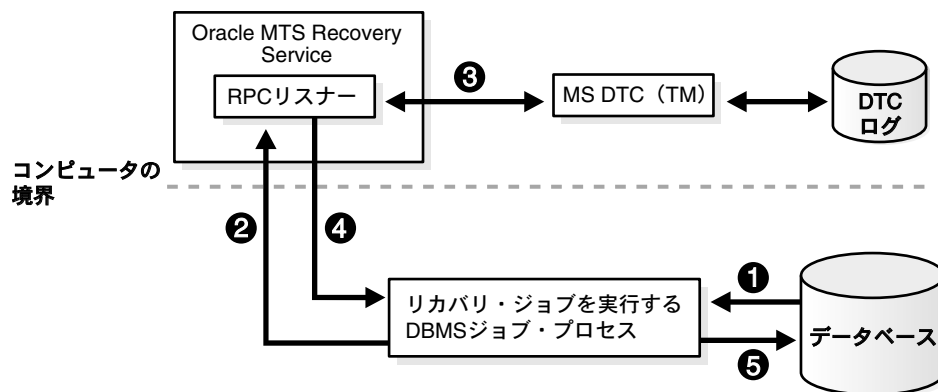
Oracle で Microsoft Transaction Server を使用するには、分散トランザクション機能が必要です。Microsoft Transaction Server 連携の Oracle トランザクションは、次のいずれかで障害が発生すると、インダウト・トランザクションになります。

- Microsoft Transaction Server アプリケーション
- ネットワーク
- **Microsoft 分散トランザクション・コーディネータ (MS DTC)**

Oracle MTS Recovery Service は、失敗したトランザクションが開始されたコンピュータで、インダウト・トランザクションを解決します。Oracle MTS Recovery Service は、Oracle Services for Microsoft Transaction Server とともに自動的にインストールされます。各コンピュータに Oracle MTS Recovery Service を 1 つのみインストールできます。Microsoft Transaction Server 対応の各データベース・サーバーに対してスケジューリングされたリカバリ・ジョブによって、Oracle MTS Recovery Service がインダウト・トランザクションを解決できるようになります。

図 3-1 は、インダウト・トランザクションの解決方法を説明したものです。

図 3-1 Microsoft Transaction Server のインダウト・トランザクションの解決方法



凡例

- 1 DBMSのリカバリ・ジョブがMTS連携のインダウト・トランザクションを検出します。
- 2 DBMSのリカバリ・ジョブがインダウト・トランザクションのXIDからRecovery Serviceのエンドポイント・アドレスを抽出し、MTS/MS DTCトランザクションの結果をRecovery Serviceに要求します。
- 3 Recovery Serviceがトランザクションの結果をMS DTCに要求します。
- 4 Recovery ServiceがDBMSのジョブ・プロセスにトランザクションの結果を報告します。
- 5 DBMSのリカバリ・ジョブがインダウト・トランザクションをコミットまたは異常終了します。

Microsoft Transaction Server 自動トランザクション・リカバリのスケジューリング

自動トランザクション・リカバリは、データベース・ジョブをスケジューリングすることで実行します。インダウト・トランザクションに対するデータベース・ジョブは、Microsoft Transaction Server トランザクションに登録しているデータベース・サーバーごとにスケジューリングする必要があります。

トランザクション・リカバリを構成するには、oramtsadmin.sql スクリプトを実行します。このスクリプトでは、スクリプト utl_oramts.sql および prvtoramts.plb を実行して、PL/SQL パッケージ utl_oramts を作成します。Microsoft Transaction Server トランザクションと連携したインダウト・トランザクションを示すデータベース・ビュー oramts_2pc_pending も作成されます。

oramtsadmin.sql スクリプトでは、次の処理が実行されます。

- Microsoft Transaction Server 管理者ユーザー・アカウントを作成
- トランザクション・リカバリのデータベース・ジョブを 1 分おきに自動的にスケジューリング

データベース・ジョブが実行されると、データベース・サーバーで Microsoft Transaction Server と連携した未解決のグローバル・トランザクションがないかどうかを確認されます。インダウト・トランザクションの **トランザクション識別子 (XID)** の情報により、トランザクションが開始されたコンピュータが示されます。そのコンピュータの Oracle MTS Recovery Service がトランザクションを解決します。

- リカバリ後のクリーンアップを 30 分おきにスケジューリング

次のタスクを実行し、データベース・サーバーでの自動トランザクション・リカバリをスケジューリングします。

- [タスク 1: SNP プロセスの設定および起動](#)
- [タスク 2: 自動トランザクション・リカバリの作成およびスケジューリング](#)

タスク 1: SNP プロセスの設定および起動

JOB_QUEUE_PROCESSES 初期化パラメータでは、インスタンスで起動されたジョブ・キュー (SNP) ・プロセスの数を指定します。

SNP プロセスを設定して起動するには、次のようにします。

1. SYSDBA 権限があることを確認します。
2. データベース・サーバーがインストールされているコンピュータにログオンします。
3. SQL*Plus を起動します。

```
C:\> sqlplus /NOLOG
```

4. データベース・サーバーに SYSDBA として接続します。

```
SQL> CONNECT / AS SYSDBA
```

5. 次の表で、データベース・サーバーのリリースに対応する初期化パラメータ・ファイルの場所を確認してください。

使用しているサーバー	移動先のディレクトリ
リリース 8.1.x 以上の データベース・サーバー	%ORACLE_HOME%\admin\¥DB_NAME\pfile¥init.ora
リリース 8.0.6 のデータ ベース・サーバー	%ORACLE_HOME%\¥database¥init.ora

6. 初期化パラメータ・ファイルを開きます。
7. 次の初期化パラメータに、ここに記載した値かそれ以上の値を設定します。

```
JOB_QUEUE_PROCESSES=1
```

このパラメータのデフォルト値は 0 です。メッセージの伝播先が多数ある場合は、このパラメータに 1 より大きい値を設定します。

8. データベース・サーバーを停止します。

```
SQL> SHUTDOWN
```

9. データベース・サーバーを再起動します。

```
SQL> STARTUP
```

10. SQL*Plus を終了します。

```
SQL> EXIT
```

タスク 2: 自動トランザクション・リカバリの作成およびスケジューリング

oramtsadmin.sql スクリプトでは、デフォルト・ユーザー名 mtssys で Microsoft Transaction Server 管理者ユーザー・アカウントを作成します。Microsoft Transaction Server トランザクション・リカバリ・ジョブは、管理者ユーザー・アカウントで実行されます。

oramtsadmin.sql スクリプトでは、utl_oramts.sql スクリプトを実行して、管理者ユーザー・アカウントに次の権限およびロールを付与します。

- CONNECT ロール
- SELECT_CATALOG_ROLE ロール
- FORCE_ANY_TRANSACTION 権限
- DBMS_JOBS パッケージで EXECUTE 権限を付与
- DBMS_TRANSACTION パッケージで EXECUTE 権限を付与

自動トランザクション・リカバリを作成してスケジューリングするには、次のようにします。

1. SYSDBA 権限があることを確認します。
2. データベース・サーバーがインストールされているコンピュータにログオンします。
3. SQL*Plus を起動します。

```
C:¥> sqlplus /NOLOG
```

4. データベース・サーバーに SYSDBA として接続します。

```
SQL> CONNECT / AS SYSDBA
```

5. oramtsadmin.sql スクリプトを実行します。

```
SQL> @%ORACLE_HOME%\oramts\admin\oramtsadmin.sql;
```

Microsoft Transaction Server の管理者ユーザー名およびパスワードをそれぞれ指定するよう求められます。デフォルトのユーザー名 mtssys およびパスワード mtssys を受け入れることも、変更することもできます。

6. 手順 5 でパスワードを変更しなかった場合、mtssys ユーザーのパスワードを後で変更してください。

```
SQL> ALTER USER mtssys IDENTIFIED BY new_password;
```

注意： ユーザー名を後で変更する場合、そのユーザーを削除し、oramtsadmin.sql スクリプトを再実行してから、プロンプトに対して別のユーザー名を指定します。

7. SQL*Plus を終了します。

```
SQL> EXIT
```

utl_oramts という単一の PL/SQL パッケージが Microsoft Transaction Server 管理者のスキーマに作成されます。utl_oramts では、次のパブリック・プロシージャを提供し、次のビューを作成します。

- utl_oramts.show_indoubt プロシージャ
- utl_oramts.recover_automatic プロシージャ
- utl_oramts.forget_RMs プロシージャ
- oramts_2pc_pending ビュー

utl_oramts.show_indoubt プロシージャ

タイプ

プロシージャ

引数

なし

戻り値

なし

説明

このプロシージャを使用して、データベース・サーバー内の Microsoft Transaction Server インダウト・トランザクションを表示します。このプロシージャでは、dbms_output パッケージを使用して結果を表示します。

このプロシージャを使用する場合、SERVEROUTPUT を ON に設定する必要があります。

```
SQL> SET SERVEROUTPUT ON
```

```
SQL> EXECUTE utl_oramts.show_indoubt;
```

次の情報が表示されます。

```
=====
currently indoubt transactions
=====
formatid   : 21255235
gtrid      : C2229A505904974D81FB7316B147325900000000
bqual      : 5BAB6A6B55CD294AA20335839110829C0100000000901944700050
local txid : 142.11.202
tx state   : prepared
protocol   : HTTP
endpoint   : middletier-1@foo.com:2030
=====
formatid   : 21255235
gtrid      : 259DF9C8DFC5574F8876F0DF4E15CCAD000000000
bqual      : 2C8DCED5B9816244BA2B73CC013EEB870100000000901944700050
local txid : 2.18.185
tx state   : prepared
protocol   : HTTP
endpoint   : middletier-2@foo.com:2030
```

utl_oramts.recover_automatic プロシージャ

タイプ

プロシージャ

引数

なし

戻り値

なし

説明

このプロシージャは、トランザクション・リカバリ・ジョブで実行されます。

utl_oramts.recover_automatic に対して自動データベース・ジョブがスケジューリングされます。ジョブが実行されると、データベース・サーバーで Microsoft Transaction Server と連携した未解決のグローバル・トランザクションがないかどうかを確認されます。インダウト・トランザクションの XID の情報により、トランザクションが開始されたコンピュータが示されます。Oracle MTS Recovery Service が通知を受け、トランザクションを解決します。

utl_oramts.forget_RMs プロシージャ

タイプ

プロシージャ

引数

なし

戻り値

なし

説明

このプロシージャを使用して、トランザクション・マネージャ（MS DTC）からの解決済トランザクションの消去を要求します。このプロシージャは、リカバリ後のクリーンアップ・ジョブで実行されます。

oramts_2pc_pending ビュー

ビュー oramts_2pc_pending は、oramtsadmin.sql を実行すると作成されます。oramts_2pc_pending は、データベース・サーバー内のインダウト・トランザクションを示します。このビューは、次の列で構成されます。

Formatid

これは、データベース・サーバー内のグローバル・トランザクションの formatid です。

global_transaction_id

これは、Microsoft Transaction Server トランザクションに対応する Oracle グローバル・トランザクションのトランザクション識別子です。実際には、Microsoft Transaction Server トランザクションの Global Unique Identifier（GUID）です。

branch_id

これは、Oracle トランザクションのブランチ識別子です。1 つの Microsoft Transaction Server トランザクションが複数の Oracle グローバル・トランザクションを持つ場合があります。これは、同じ Microsoft Transaction Server トランザクション内に存在する Microsoft Transaction Server/COM+ コンポーネントの数に依存します。これらのトランザクションはすべて、下位のグローバル・トランザクション識別子を持ちますが、異なるブランチ識別子を持ちます。

local_tx_id

ローカルの Oracle トランザクションは、各 Microsoft Transaction Server トランザクションに対応します。この列は、このローカル・トランザクションに対応する識別子を示します。

state

これは、ペンディング、ヒューリスティックにコミット、ヒューリスティックに異常終了などのトランザクションの状態を示します。

protocol

これは、データベース・サーバーのトランザクション・リカバリ・ジョブが Oracle MTS Recovery Service との通信に使用するプロトコルです。

endpoint

これは、Microsoft Transaction Server トランザクションが開始された Windows NT/2000 コンピュータのエンドポイントです。HTTP 接続の場合、これはホスト名およびポート番号に変換されます。

Microsoft Transaction Server インダウト・トランザクションの表示

データベース・サーバー内の Microsoft Transaction Server 連携のインダウト・トランザクションを表示するには、SQL*Plus を使用してビュー `oramts_2pc_pending` を問い合わせます。

Microsoft Transaction Server 連携のインダウト・トランザクションを表示するには、次のようにします。

1. Microsoft Transaction Server 管理者ユーザー・アカウントで SQL*Plus を起動します。

```
C:\> sqlplus mtsadmin_user/mtsadmin_password
```

2. 次のコマンドを入力します。

```
SQL> SELECT * FROM oramts_2pc_pending;
```

これにより、インダウト・トランザクションが開始されたコンピュータが表示されます。

Oracle Fail Safe 構成用のレジストリ値の変更

通常の構成では、MS DTC と Oracle MTS Recovery Service は同じコンピュータで稼働します。この場合、トランザクション・リカバリに必要な情報が Oracle と Microsoft Transaction Server の統合レイヤーで使用できます。

Microsoft Transaction Server アプリケーションが Windows NT クラスタの一部となるような構成（たとえば、アプリケーションがクラスタ内の別のノードまたはホストにフェイルオーバー可能な構成）では、MS DTC はクラスタ全体のリソースとして稼働します。すべてのクラスタ・ノードが、任意のクラスタ・ノードで稼働している MS DTC の 1 つのインスタンスを使用します。**Oracle Fail Safe** を使用すると、Oracle MTS Recovery Service もクラスタ全体のリソースにすることができます。

Oracle Fail Safe 構成を採用している場合、Microsoft Transaction Server トランザクションに登録しているクラスタ内の全ノードに次のレジストリ情報を必ずレプリケートしてください。

Oracle Fail Safe 構成用にレジストリ値を変更するには、次のようにします。

1. MS DTC および Oracle MTS Recovery Service がインストールされているコンピュータにログオンします。
2. コマンド・プロンプトからレジストリ エディタを起動します。

```
C:\> regedt32
```


「レジストリ エディタ」ウィンドウが表示されます。
3. `HKEY_LOCAL_MACHINE\Software\Oracle\OracleMTSRecoveryService` に移動します。
4. ここに指定されているレジストリ情報をクラスタ内の全ノードにコピーします。
5. キーを追加したコンピュータを再起動します。

Microsoft アプリケーション・デモの実行

この章では、**Microsoft Transaction Server** に組み込まれているサンプルの **Microsoft Component Object Model (COM)** アプリケーション・デモを使用する方法について説明します。

この章の項目は次のとおりです。

- **Microsoft アプリケーション・デモでの OCI の構成**
- **Microsoft アプリケーション・デモでの Oracle ODBC ドライバの構成**
- **Microsoft アプリケーション・デモでの Oracle Provider for OLE DB の構成**

注意： **Microsoft アプリケーション・デモ** は、Windows 2000 の Microsoft Transaction Server には組み込まれていません。

Microsoft アプリケーション・デモでの OCI の構成

Microsoft Transaction Server に付属している Sample Bank アプリケーション・デモで、[Oracle Call Interface \(OCI\)](#) を使用できます。多くの場合、OCI は Microsoft アプリケーション・デモに自動的に統合されます。[表 4-1](#) で、OCI と Microsoft アプリケーション・デモが統合されているかどうかを調べ、統合されていない場合の手順を確認してください。

表 4-1 OCI と Microsoft アプリケーション・デモの統合の確認

Microsoft Transaction Server の状態	動作
Oracle Services for Microsoft Transaction Server (MTS) のインストール時にすでにインストールされている	Oracle Universal Installer が、Sample Bank デモに含まれるいくつかの Visual C++ のファイルを自動的にバックアップし、それらを oci.dll ファイルと oramts.dll ファイルを統合するファイルに置き換える。これによって、Sample Bank デモで OCI を使用できるようになります。
Oracle Services for Microsoft Transaction Server のインストール時にインストールされていない	次の手順を実行する。 <ol style="list-style-type: none">1. Microsoft Transaction Server をインストールします。2. Windows NT コンピュータの <code>ROOTDRIVE:¥program files¥mts¥samples¥packages¥vcacct.dll</code> ファイルを別の場所にバックアップします。3. <code>%ORACLE_HOME¥oramts¥samples¥account.vc¥release</code> にある <code>vcacct.dll</code> ファイルを、手順 2 で示した Windows NT のディレクトリにコピーします。

Microsoft アプリケーション・デモの概要

Microsoft アプリケーション・デモは、`%ORACLE_HOME¥oramts¥samples¥account.vc` ディレクトリにインストールされます。このデモは、Microsoft Transaction Server に付属している Visual C++ Sample Bank パッケージの OCI 実装です。デモ・コンポーネントは、ユーザー・アカウント `scott` およびパスワード `tiger` を使用して、`mtsdemo` という [ネット・サービス名](#) を持つデータベースに接続します。この情報は、`oramisc.h` ファイルで変更できます。デモでは、次の 2 つの表も使用します。

- `account`
- `receipt`

これらの表は、インストール時に作成されるデフォルトの Oracle データベース・サーバーに含まれる、ユーザー `scott` のスキーマの一部です。詳細は、「[データベースに Microsoft アプリケーション・デモの適切な表が含まれていることの確認](#)」を参照してください。

データベースに Microsoft アプリケーション・デモの適切な表が含まれていることの確認

デフォルトのデータベース・サーバーを使用していない場合や、データベース・サーバーにユーザー `scott` が含まれていない場合は、Microsoft アプリケーション・デモの Sample Bank の実行に必要な表を、関連するユーザーのスキーマに作成します。

データベース・サーバーに適切な表が含まれていることを確認するには、次のようにします。

1. データベース・サーバーに適切な表が含まれているかどうかを次の表で確認してください。

データベース・サーバーの作成方法	状態
<ul style="list-style-type: none">■ Oracle Universal Installer で、「Enterprise Edition」、「Standard Edition」、「Personal Edition」のいずれかのインストール・タイプの「汎用目的」、「トランザクション処理」、「データ・ウェアハウス」のいずれかのデータベース構成タイプを選択。■ ユーザー定義の SQL スクリプトを使用して手動で作成し、次にデータベースに対して <code>scott.sql</code> スクリプトおよび <code>omtssamp.sql</code> スクリプトをこの順序で明示的に実行。¹■ Oracle Database Configuration Assistant で、次のデータベース構成タイプを選択。<div>General Purpose</div><div>Transaction Processing</div><div>Data Warehouse</div>	データベース・サーバーには適切な表が含まれている。4-5 ページの「 Microsoft アプリケーション・デモの実行 」に進んでください。
<ul style="list-style-type: none">■ ユーザー定義の SQL スクリプトを使用して手動で作成し、そのデータベースに対して <code>omtssamp.sql</code> を実行しない。■ SQL*Plus のコマンドライン・モードで CREATE DATABASE 構文を使用。■ Solaris やその他のオペレーティング・システムで実行可能な任意の方法。	データベース・サーバーには適切な表が含まれていない。4-5 ページの「 Microsoft アプリケーション・デモの実行 」に進む前に、この項の手順 1～6 を実行してください。

¹ `scott.sql` を実行する前に `omtssamp.sql` を実行した場合、多数のエラー・メッセージが表示されます。これらのメッセージは無視してください。製品は正常に機能しますが、この章で説明しているサンプルのアプリケーション・デモは動作しません。

データベース・サーバーに適切な表が含まれていない場合、それらを手動で作成する必要があります。

適切な表を手動で作成するには、次のようにします。

1. SQL*Plus を起動します。

```
C:¥> sqlplus /NOLOG
```

2. データベース・サーバーに接続します。

```
SQL> CONNECT username/password@net_service_name
```

`net_service_name` は、データベース・サーバーに接続するためのネット・サービス名です。scott 以外のユーザー名でデータベース・サーバーに接続した場合、`oramisc.h` のユーザー名およびパスワードを変更し、`vcacct.dll` をビルドしなおす必要があります。

3. ユーザー `mtsdemousr` を作成します。

```
SQL> CREATE USER mtsdemousr IDENTIFIED BY mtsdemousr;
```

これにより、Microsoft サンプル・アプリケーションを実行するユーザー `mtsdemousr` が作成されます。

4. ユーザー `mtsdemousr` に次のロールを割り当てます。

```
SQL> GRANT CONNECT, RESOURCE TO mtsdemousr;
```

5. ユーザー `mtsdemousr` で接続します。

```
SQL> CONNECT mtsdemousr/mtsdemousr@net_service_name
```

6. 次の SQL スクリプトを実行します。

```
SQL> @%ORACLE_HOME%\oramts¥samples¥sql¥omtssamp.sql;
```

これにより、ユーザー `mtsdemousr` のスキーマに `account` 表および `receipt` 表が作成されます。

Microsoft アプリケーション・デモの実行

この項では、Microsoft アプリケーション・デモの実行方法について説明します。

Microsoft アプリケーション・デモを実行するには、次のようにします。

注意： サンプル・プロジェクトは、Visual C++ 5.0 で作成されています。Visual C++ 6.0 を使用してプロジェクトをビルドするには、Visual C++ 6.0 に含まれる新しいリリースの MIDL コンパイラで Microsoft の .idl ファイルを処理する必要があります。

1. ディレクトリ %ORACLE_HOME%\oramtms\samples\account.vc に移動します。
2. コマンド・プロンプトで次のように入力します。

```
midl account.idl
```

Visual C++ 5.0 のプロジェクトを開くと、Visual C++ 6.0 によって自動的に新しいリリースに更新されます。

1. Microsoft Transaction Server に付属している Sample Bank クライアント・アプリケーションがインストールされていることを確認します。サンプル・コンポーネントの Dynamic Link Library (DLL) は、通常 `ROOTDRIVE:\program files\mtms\samples\packages` にインストールされます。また、次のようにコントロールパネルからインストールを確認することもできます。
 - a. 「スタート」→「設定」→「コントロールパネル」を選択します。
 - b. 「アプリケーションの追加と削除」をダブルクリックします。
「アプリケーションの追加と削除のプロパティ」ダイアログ・ボックスが表示されます。
 - c. 「Windows NT 4.0 Option Pack」を選択し、「追加と削除」をクリックします。
セットアップの初期化には、少し時間がかかります。初期化が終了すると、「Microsoft Windows NT 4.0 Option Pack セットアップ」ダイアログ・ボックスが表示されます。
 - d. 「次へ」をクリックします。
 - e. 「追加と削除」をクリックします。
 - f. 「Transaction Server」が表示されるまで一覧をスクロールします。
 - g. 「Transaction Server」をダブルクリックします。

- h. 「Transaction Server Development」サブコンポーネントの横のチェックボックスが選択されていることを確認します。選択されている場合は、デモはインストールされています。このサブコンポーネントが選択されていない場合は、サブコンポーネントを選択し、手順 **1i** ~ **1m** を実行します。それ以外の場合は、手順 **2** に進みます。
- i. 「OK」をクリックします。
- j. 「Microsoft Windows NT 4.0 Option Pack セットアップ」ダイアログ・ボックスで、「次へ」をクリックします。

インストールには少し時間がかかります。
- k. 「アプリケーションの追加と削除のプロパティ」ダイアログ・ボックスを終了します。
- l. 「コントロールパネル」を閉じます。
- m. 手順 **2** に進みます。
- 2. Microsoft Developer Studio を使用して、
%ORACLE_HOME%\oramts\samples\account.vc ディレクトリのプロジェクト
account.dsp を開きます。
- 3. vcacct.dll をビルドします。
- 4. ROOTDRIVE:\program files\mts\samples\packages にインストールされている
ファイル vcacct.dll をバックアップします。
- 5. **3** でビルドしたファイルで、vcacct.dll を上書きします。
- 6. 「スタート」→「プログラム」→「Windows NT 4.0 Option Pack」→「Microsoft
Transaction Server」→「Bank クライアント」を選択します。

これにより、Visual Basic の Sample Bank アプリケーションが起動し、コンポーネント
をテストする vbbank.exe が実行されます。
- 7. 「Language」フィールドの「Visual C++」ラジオ・ボタンをクリックします。
- 8. 「Component」フィールドの「Account」または「MoveMoney」ラジオ・ボタンをク
リックします。
- 9. 処理する口座番号および金額を入力します。
- 10. 「Submit」をクリックします。
- 11. SQL*Plus を起動します。

C:\> sqlplus /NOLOG
- 12. ユーザー名 scott およびパスワード tiger (変更していない場合) で接続します。

SQL> CONNECT scott/tiger

13. 処理が正常に終了したことを確認します。

```
SQL> SELECT * FROM account;  
SQL> SELECT * FROM receipt;
```

Microsoft アプリケーション・デモでの Oracle ODBC ドライバの構成

Microsoft サンプル・アプリケーションでリリース 9.2 の [Oracle ODBC ドライバ](#)を使用できます。Oracle ODBC ドライバをサンプル・アプリケーションに統合する方法については、5-25 ページの「[Oracle ODBC ドライバの使用](#)」を参照してください。

Microsoft アプリケーション・デモでの Oracle Provider for OLE DB の構成

Microsoft サンプル・アプリケーションでリリース 9.2 の [Oracle Provider for OLE DB](#)を使用できます。Microsoft Transaction Server での Oracle Provider for OLE DB の使用については、『Oracle Provider for OLE DB 開発者ガイド』を参照してください。

Microsoft Transaction Server および Oracle データベース・サーバーを使用した プログラミング

この章では、**Microsoft Transaction Server** および Oracle データベース・サーバーを使用したプログラミング方法について説明します。

この章の項目は次のとおりです。

- トランザクション内での COM コンポーネント統合の概要
- Microsoft Transaction Server アプリケーションの開発の概要
- OCI と Microsoft Transaction Server の統合の概要
- ODBC と Microsoft Transaction Server の統合の概要
- OO4O と Microsoft Transaction Server の統合の概要
- Oracle Provider for OLE DB と Microsoft Transaction Server の統合の概要
- その他の API と Microsoft Transaction Server の統合の概要

トランザクション内での COM コンポーネント統合の概要

トランザクション処理の中心となるのは、**Microsoft 分散トランザクション・コーディネータ (MS DTC)** と呼ばれる Microsoft Transaction Server のコンポーネントです。クライアント・コンピュータがトランザクション・コンポーネントでビジネス・メソッドを起動すると、Microsoft Transaction Server は MS DTC によって調整されるトランザクションを開始します。Oracle 接続をプールしているレイヤーにより、データベース・サーバーは MS DTC によって調整されるトランザクション内で**リソース・マネージャ (RM)** として機能します。**図 5-1** および**表 5-1** は、これらのコンポーネントやその他のコンポーネントがトランザクションを実行する方法の概要を示しています。

図 5-1 トランザクション内でのコンポーネント統合

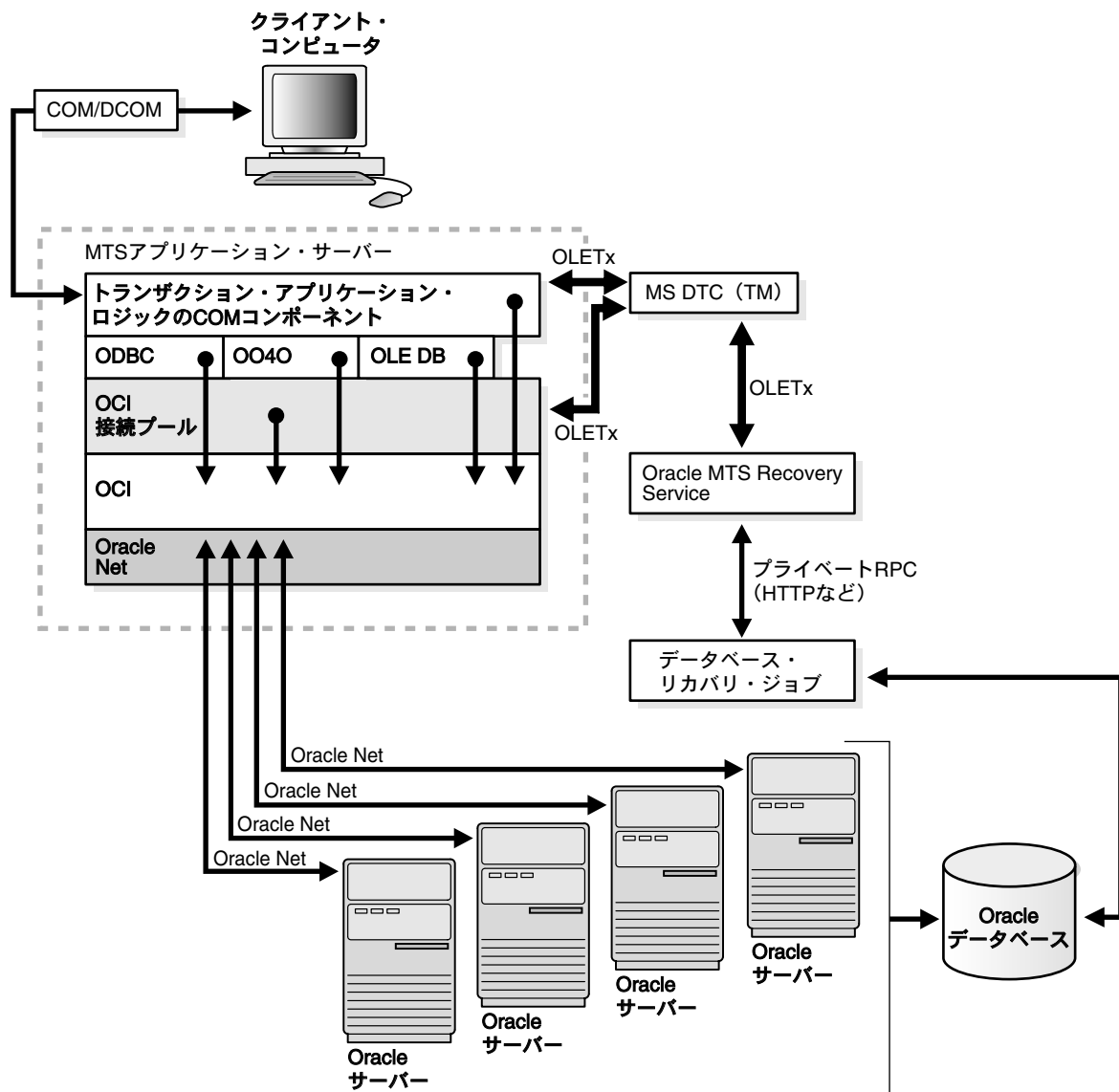


表 5-1 トランザクション内でのコンポーネント統合

コンポーネント	主な役割
クライアント・コンピュータ接続	<ul style="list-style-type: none"> Web ブラウザまたは Component Object Model (COM) / Distributed Component Object Model (DCOM) を使用してアプリケーション・コンポーネントを起動。
トランザクション・アプリケーション・ロジックの COM コンポーネント	<ul style="list-style-type: none"> ビジネス・ロジックの埋込み（コンポーネントがトランザクションの場合、Microsoft Transaction Server により、そのコンポーネントでメソッドが起動されるごとにトランザクションが開始される。）。 Oracle リソース・ディスペンサおよび Oracle Call Interface (OCI)、Oracle ODBC ドライバ、Oracle Provider for OLE DB または Oracle Objects for OLE (OO4O) を通じて、データベース・サーバーへのプール済の接続を取得。 変更をすべての RM にコミットするか異常終了するかを決定を Microsoft Transaction Server に通知することによって、処理の結果を決定。
Oracle ODBC ドライバ、OO4O、Oracle Provider for OLE DB および OCI	<ul style="list-style-type: none"> データベース・サーバーへのサービス・コンテキストを、OCI 接続プール・コンポーネントを通じて取得。 必要に応じて、プールしている接続リソースを（Oracle Provider for OLE DB または Oracle ODBC ドライバを通じて）提供。Oracle ODBC ドライバは、プール済の ODBC 接続を提供します。Oracle Provider for OLE DB は、プール済のデータソース・オブジェクトを提供します。OO4O は、OCI 接続プールを使用します。
OCI 接続プール	<p>トランザクション・コンポーネントのために次の処理を実行。</p> <ul style="list-style-type: none"> RM（データベース・サーバー）としてコンポーネントの Microsoft Transaction Server トランザクションに登録。 コンポーネントが属する Microsoft Transaction Server トランザクションに対応する Oracle グローバル・トランザクションを開始。 クライアント側の接続プーリングを実行するリソース・ディスペンサとして機能。
Oracle MTS Recovery Service	<ul style="list-style-type: none"> ホスト・コンピュータで開始された、Microsoft Transaction Server 連携のインダウト Oracle トランザクションをリカバリ。
MS DTC (Microsoft Transaction Server の一部)	<ul style="list-style-type: none"> 2 フェーズ・コミット・プロトコルを使用して、トランザクションをコミットまたは異常終了。 リカバリが必要なトランザクションを監視。1 つのトランザクションに複数の MS DTC を関与させることができます。コンピュータ A にある Microsoft Transaction Server のトランザクション・コンポーネントが、コンピュータ B にある Microsoft Transaction Server の別のトランザクション・コンポーネントを起動すると、コンピュータ A の MS DTC とコンピュータ B の MS DTC 間で接続がオープンされます。ルート MS DTC がトランザクションをコミットまたは異常終了するときには、関与するすべての MS DTC を通じて要求を送信します。このトランザクション要求は OCI 接続プール /Microsoft Transaction Server 統合に渡され、そこからデータベース・サーバーに送られます。
データベース・サーバー	<ul style="list-style-type: none"> Microsoft Transaction Server の RM として機能。このデータベース上で、クライアントのトランザクション要求が実行されます。

Microsoft Transaction Server アプリケーションの開発の概要

使用する API にかかわらず、トランザクションの調整にはほぼすべての場合に OCI 接続プールが利用されます。トランザクションを登録する方法、および OCI 接続プールでトランザクションを調整する方法については、次の各項を参照してください。

- [Microsoft Transaction Server コンポーネントの登録の概要](#)
- [Microsoft Transaction Server によって調整されるコンポーネント・トランザクションの概要](#)
- [MS DTC によって調整されるコンポーネント・トランザクションの概要](#)

関連資料： Microsoft Transaction Server のアプリケーション・コンポーネントを開発する方法については、Microsoft Transaction Server のヘルプ・ファイルにある『Microsoft Transaction Server Programmer's Guide』を参照してください。

Microsoft Transaction Server コンポーネントの登録の概要

Microsoft Transaction Server 環境で実行されるアプリケーション・コンポーネントは、DLL として作成されます。アプリケーション・コンポーネントは、Graphical User Interface (GUI) ツールの Microsoft Transaction Server エクスプローラを使用して Microsoft Transaction Server に登録されます。アプリケーション・コンポーネントを登録するときには、[表 5-2](#) で説明されているタイプのいずれかを指定します。

表 5-2 Microsoft Transaction Server コンポーネントの登録

タイプ	コンポーネントの実行方法
トランザクションが必要	トランザクション内で実行する必要がある。現在トランザクションが存在しない場合は、Microsoft Transaction Server により、コンポーネントでメソッドが起動されるごとに新しいトランザクションが自動的に作成されます。
新しいトランザクションが必要	それ自体のトランザクション内で実行する必要がある。Microsoft Transaction Server により、コンポーネントでメソッドが起動されるごとに新しいトランザクションが自動的に作成されます。
トランザクションをサポートする	クライアントのトランザクション内で実行できる。新しいコンポーネントが作成されると、そのコンテキストは起動元クライアントのコンテキストからトランザクションを継承します。クライアントにトランザクションがない場合は、新しいコンテキストもトランザクションなしで作成されます。
トランザクションをサポートしない	トランザクション内で実行されない。起動元のクライアントにトランザクションがあるかどうかにかかわらず、コンポーネントでの各メソッドの起動はそれを含むトランザクションなしで実行されます。

アプリケーション・コンポーネントの登録方法により、そのコンポーネントが Microsoft Transaction Server によって調整されるトランザクション内で実行されるかどうかが決まります。表 5-3 は、このプロセスを説明したものです。

表 5-3 Microsoft Transaction Server トランザクション内でのコンポーネントの実行

アプリケーション・コンポーネントの実行	状態
Microsoft Transaction Server によって調整されるトランザクション内で実行	<p>OCI 接続プールが常に使用され、Microsoft Transaction Server およびその MS DTC コンポーネントがトランザクションの作成、開始、管理およびコミットの各フェーズを調整する。Microsoft Transaction Server により、コンポーネントが行ったすべての変更は、トランザクションが成功した場合はコミットされ、失敗した場合は異常終了されます。</p> <p>関連項目： 5-7 ページの「Microsoft Transaction Server によって調整されるコンポーネント・トランザクションの概要」</p>
Microsoft Transaction Server によって調整されないトランザクション内で実行されない	<p>コンポーネントは Microsoft Transaction Server 環境で実行されるが、コンポーネントがアクセスするデータベース・サーバーは、MS DTC によって調整されるトランザクションに登録される場合とされない場合がある。MS DTC によって調整されないトランザクションの場合は、クライアント・アプリケーションがトランザクションの作成、開始、管理およびコミットを行う必要があります。データベース・サーバーにアクセスするインタフェース（Oracle Provider for OLE DB、Oracle ODBC ドライバ、OO4O、その他）によっては、OCI 接続プールが使用される場合があります。</p> <p>関連項目： 5-8 ページの「MS DTC によって調整されるコンポーネント・トランザクションの概要」</p>

Microsoft Transaction Server によって調整されるコンポーネント・トランザクションの概要

この項では、Microsoft Transaction Server によって調整されるトランザクション環境で、OCI 接続プール、Microsoft Transaction Server および MS DTC が、アプリケーション・コンポーネントでどのように機能するかを説明します。

1. 使用中のクライアント API (Oracle ODBC ドライバ、OCI、OO4O または Oracle Provider for OLE DB) が OCI 関数 `OraMTSSvcGet()` をコールして、OCI 接続プール・コンポーネントからサービス・コンテキストを取得します。
2. OCI 接続プール・コンポーネントがトランザクションを登録します。このトランザクションは、Microsoft Transaction Server の MS DTC コンポーネントによって調整されます。

これにより、OCI サービスおよび環境ハンドルがクライアント・アプリケーションに返されます。

3. クライアント・アプリケーションが次の処理を実行します。
 - a. データベース処理を実行します。
 - b. OCI 関数 `OraMTSSvcRel()` をコールして、トランザクションの開始時に取得した OCI 接続プールを解放します。
 - c. コンポーネントに関連付けられた Microsoft Transaction Server のコンテキスト・オブジェクトに対して、`SetComplete` (データベース処理をコミットする場合) または `SetAbort` (データベース処理を異常終了する場合) をコールします。
4. MS DTC は 2 フェーズ・コミット・プロトコルを実行し、トランザクションの準備とコミット (または異常終了) を行います。この通知が OCI 接続プール・コンポーネントに送られて、トランザクションが終了します。
5. OCI 接続プールは通知を受信し、フェーズ 1 (フェーズの準備) およびフェーズ 2 (フェーズのコミットまたは異常終了) の完了に必要な手順を実行します。

MS DTC によって調整されるコンポーネント・トランザクションの概要

この項では、Microsoft Transaction Server によって調整されるトランザクション内で実行されていないが MS DTC を使用しているアプリケーション・コンポーネントで、OCI 接続プール、Microsoft Transaction Server および MS DTC がどのように機能するかを説明します。

1. クライアント・アプリケーションは、MS DTC トランザクションを開始し、データベース・サーバーに接続します。これは次のいずれかです。
 - OCI ログオン・コール (`OCIServerAttach()` や `OCISessionBegin()` など) を通じて取得したプールされていない OCI 接続。このような接続の場合、アプリケーションは `OraMTSEnlCtxGet()` をコールして、OCI サービス・コンテキストを Microsoft Transaction Server の登録コンテキストに関連付けます。
 - `OraMTSSvcGet(..., ORAMTS_CFLG_NOIMPLICIT)` をコールすることで取得され、`OraMTSSvcRel()` によってまだ解放されていない接続プール。
2. プールされていない接続の場合、クライアント・アプリケーションは登録コンテキストを `OraMTSJoinTxn()` に渡します。
3. プール済の接続の場合、クライアント・アプリケーションは OCI サービス・コンテキストを `OraMTSSvcEnlist()` に渡します。
4. OCI 接続プール・コンポーネントが、接続（プール済またはプールされていない）を、Microsoft Transaction Server の MS DTC コンポーネントによって調整されるトランザクションに登録します。
5. クライアント・アプリケーションが次の処理を実行します。
 - a. データベース処理を実行します。
 - b. NULL トランザクション参照で `OraMTSSvcEnlist()` をコールし、MS DTC によって調整されるトランザクションから登録解除します。プールされていない接続の場合、NULL トランザクション参照で `OraMTSTxnJoin()` がコールされ、登録解除が実行されます。
 - c. `OraMTSSvcRel()` をコールし、プール済の接続を解放してプールに戻します。プールされていない接続の場合、クライアントは `OraMTSEnlCtxRel()` をコールして登録コンテキストを解放した後、データベースからログオフします。
 - d. MS DTC トランザクション・オブジェクトに対して、コミット・メソッドまたは異常終了メソッドをコールします (`pTransaction->Commit()` や `pTransaction->Abort()` など)。
6. MS DTC が 2 フェーズ・コミット・プロトコルを実行し、トランザクションをコミットします。
7. OCI 接続プールは通知を受信し、フェーズ 1（フェーズの準備）およびフェーズ 2（フェーズのコミットまたは異常終了）の完了に必要な手順を実行します。

OCI と Microsoft Transaction Server の統合の概要

OCI クライアント・アプリケーションと、Microsoft Transaction Server およびデータベース・サーバーを統合するには、次の OCI 関数を使用します。この統合については、次の各項を参照してください。

- [OCI および Microsoft Transaction Server の関数の概要](#)
- [OraMTSSvcGet\(\) 関数](#)
- [OraMTSSvcRel\(\) 関数](#)
- [OraMTSSvcEnlist\(\) 関数](#)
- [OraMTSSvcEnlistEx\(\) 関数](#)
- [OraMTSEnlCtxGet\(\) 関数](#)
- [OraMTSEnlCtxRel\(\) 関数](#)
- [OraMTSJoinTxn\(\) 関数](#)
- [OraMTSTransTest\(\) 関数](#)
- [OraMTSOCIErrGet\(\) 関数](#)

OCI および Microsoft Transaction Server の関数の概要

OCI リリース 8.1 以上を使用する必要があります。リリース 8.0 以下のリリースの OCI はサポートされません。

注意： 他の C++ の Microsoft Transaction Server コンポーネントと同様に、オブジェクト・コンテキストを取得して、コンポーネントの処理の状態に応じて `SetAbort()`、`SetComplete()`、`EnableCommit()` または `DisableCommit()` をコールしてください。OCITransCommit() や OCITransAbort() などの OCI トランザクション・コールは行わないでください。データが破損する危険性があります。

コードを変更するのは、OCI サービス・コンテキスト・ハンドルの取得および解放の部分のみです。OCI 関数 `OraMTSSvcGet()` でデータベースへのプール済の OCI 接続を取得するときに、OCI サービス・コンテキスト・ハンドルおよび環境ハンドルが取得されます。`oramts.h` ヘッダーをインクルードして、`oramts.lib` ライブラリとリンクします。それが終了したら、OCI 関数 `OraMTSSvcRel()` をコールして、サービス・コンテキスト・ハンドルおよび環境ハンドルを解放します。`OraMTSSvcGet()` を使用すると、接続プールおよび暗黙的なトランザクション・サポートが得られます（アプリケーション・コンポーネントを Microsoft Transaction Server トランザクション内で実行するように登録した場合）。詳細は、5-13 ページの「[OraMTSSvcGet\(\) 関数](#)」および 5-16 ページの「[OraMTSSvcRel\(\) 関数](#)」を参照してください。

リリース 8.1 以下のリリースでは、OraMTSSvcEnlist() および OraMTSSvcEnlistEx() により、プールされていない OCI 接続が Microsoft Transaction Server トランザクションに登録されました。これはサポートされなくなりました。MS DTC によって調整されるトランザクションにプール済の接続を登録する場合は、これらの関数を引き続き使用できます。Microsoft Transaction Server で開始され、MS DTC によって調整されるトランザクションにプールされていない OCI 接続を登録するには、クライアントが OraMTSJoinTxn() を使用する必要があります。詳細は、5-22 ページの「[OraMTSJoinTxn\(\) 関数](#)」を参照してください。

プロセスごとに、少なくとも 1 回 OCIInitialize をコールしてから、他の OCI コールを実行するようにしてください。これにより、OCI のプロセス環境が初期化されます。さらに、OCI_THREADED フラグを渡す必要があります。Microsoft の Internet Information Server (IIS) を使用していて、コンポーネントがインプロセス・ライブラリとしてコールされている場合は、すでに OCIInitialize が自動的にコールされています。

```
#include <oci.h>
#include <oramts.h>
#include <xolehlp.h>
// other MTS relevant includes ...

// prototype for the error handler.
BOOL Chekerr(sword swOCISat, OCIError *OCIErrh);

// MTS component method
HRESULT OCITestMethod()
{
    IObjectContext *pObjectContext = NULL;
    OCIEnv *myenvh = NULL;
    OCISvcCtx *mysvch = NULL;
    OCIError *myerrh = NULL;
    OCISnt *mystmh = NULL;
    DWORD dwStat;
    HRESULT hRes = S_OK;
    sword swOCISat;
    BOOL bCommit = FALSE;
    char *lpzStmt = "UPDATE EMP SET SAL = SAL + 1000";

    // Initialize the OCI environment first -- request OCI_THREADED
    OCIInitialize(OCI_THREADED, (dvoid*)NULL, NULL, NULL, NULL);
    // attempt to get a connection to the database through the resource dispenser
    OraMTSSvcGet(
        "scott", "tiger", "finprod_db", &mysvch, &myenvh, ORAMTS_CFLG_ALLDEFAULT);
    // validate return status
    if(dwStat != ORAMTS_ERR_NOERROR)
    {
        printf("error: failed to obtain a connection to the database - %ld",
            dwStat);
    }
}
```

```

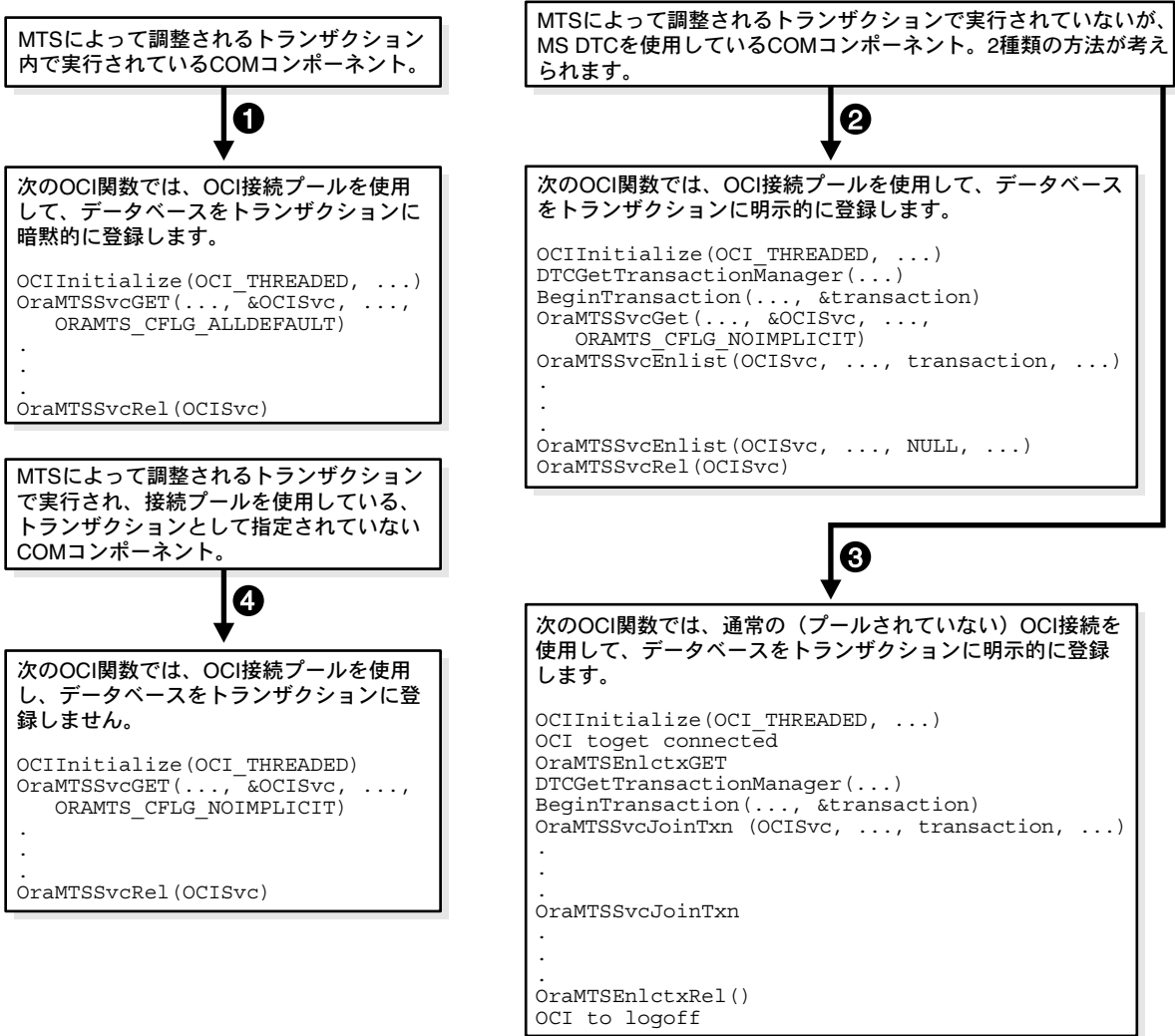
        goto cleanup;
    }
    // successful logon and enlistment in the MTS transaction. allocate statement
    // handles and other handles using the OCI environment handle myenvh ....
    swOCISat = OCIHandleAlloc(myenvh, (void *)&myerrh, OCI_HTYPE_ERROR, 0, NULL);
    if (Checkerr(swOCISat, myerrh)) goto cleanup;
    swOCISat = OCIHandleAlloc(myenvh, (dvoid *)&mystmh, OCI_HTYPE_STMT, 0, NULL);
    if (Checkerr(swOCISat, myerrh)) goto cleanup;
    // prepare a DML statement
    OCISatPrepare(mystmh, myerrh, lpzStmt, lstrlen(lpzStmt), OCI_NTV_SYNTAX,
OCI_DEFAULT)
    Checkerr(swOCISat, myerrh);
    // execute the statement -- ensure that AUTOCOMMIT is not requested.
    OCISatExecute(mysvch, mystmh, myerrh, 1, 0, NULL, NULL, OCI_DEFAULT);
    if (Checkerr(swOCISat, myerrh)) goto cleanup;
    // all's well so far choose to go for a commit
    bCommit = TRUE;
cleanup:
    if (mystmh) OCIHandleFree((void*)mystmh, OCI_HTYPE_STMT);
    if (myerrh) OCIHandleFree((void*)myerrh, OCI_HTYPE_ERROR);
    if (mysvch) OraMTSSvcRel(mysvch);
    if (bCommit)
        pObjectContext->SetComplete();
    else
        pObjectContext->Abort();
    return(bCommit ? S_OK : E_FAIL);
}

```

関連資料： コードのサンプルは、
%ORACLE_HOME%\oramts\samples\account.vc ディレクトリのファイル
を参照してください。

図 5-2 は、OCI 関数 OraMTSSvcGet ()、OraMTSSvcRel () および OraMTSJoinTxn の使用方法の概要を示しています。

図 5-2 OCI 関数の使用方法



Microsoft Transaction Server 環境をホストとしない COM アプリケーション（スタンドアロン・アプリケーション）でも、[図 5-2](#) の 2、3 および 4 の方法を使用できます。ただし、このタイプのアプリケーションでは（Microsoft Transaction Server エクスプローラの Microsoft 管理コンソール・スナップインを通じて）、宣言トランザクションは使用できません。

OraMTSSvcGet() 関数

機能

OraMTSSvcGet () は、OCI 接続プールからプール済の接続（OCI サービス・コンテキスト）を取得します。プール済の接続には、OCI サービス・コンテキスト・ハンドルおよび OCI 環境ハンドルが含まれます。

構文

```
DWORD OraMTSSvcGet (
    text*      lpUname,
    text*      lpPswd,
    text*      lpDbnam,
    OCISvcCtx** pOCISvc,
    OCIEnv**   pOCIEnv,
    ub4        dwConFlgs
);
```

パラメータ

[表 5-4](#) は、OraMTSSvcGet () のパラメータを説明したものです。

表 5-4 OraMTSSvcGet() のパラメータ

データ型	パラメータ	説明
text*	lpUname (IN)	データベース・サーバーに接続するためのユーザー名
text*	lpPswd (IN)	ユーザー名に対応するパスワード
text*	lpDbnam (IN)	データベースに接続するための ネット・サービス名 (Oracle Net Manager または Oracle Net Configuration Assistant で作成)
OCISvcCtx**	pOCISvc (OUT)	OCI サービス・コンテキスト・ハンドルへのポインタ
OCIEnv**	pOCIEnv (OUT)	OCI 環境ハンドルへのポインタ

表 5-4 OraMTSSvcGet() のパラメータ (続き)

データ型	パラメータ	説明
ub4	dwConFlgs (IN)	次の値をとる接続フラグ <ul style="list-style-type: none">■ ORAMTS_CFLG_ALLDEFAULT プール済の接続を取得し、Microsoft Transaction Server トランザクションが存在する場合は接続を登録します。コンポーネントが非トランザクションの場合は、登録要求を発行しません。■ ORAMTS_CFLG_NOIMPLICIT プール済の接続を取得しますが、そのコンポーネントがトランザクションの場合も、Microsoft Transaction Server トランザクションにリソースを登録しません。コンポーネントが OraMTSSvcEnlist () を使用して、後から接続を登録する場合は、このフラグを使用します。この方法で取得した接続を解放するときに、リソースが登録されている場合は、その前にクライアントがリソースの登録を解除する必要があります。■ ORAMTS_CFLG_UNIQUESEVR OCI サーバーごとに 1 つの OCI セッションを要求します。このリリースでは、多重化がサポートされていません。したがって、常にこのオプションが使用されます。■ ORAMTS_CFLG_SYSDBALOGN SYSDBA として接続する場合にこのフラグを使用します。■ ORAMTS_CFLG_SYSOPRLOGN SYSOPER として接続する場合にこのフラグを使用します。■ ORAMTS_CFLG_PRELIMAUTH ユーザー INTERNAL として Oracle8i 以下のデータベースに接続する場合にこのフラグを使用します。INTERNAL アカウントは、Oracle9i では無効になりました。かわりに、ORAMTS_CFLG_SYSOPRLOGN または ORAMTS_CFLG_SYSDBALOGN フラグを使用して、SYSDBA または SYSOPER アカウントでログオンしてください。

戻り値

プール済の OCI 接続（OCI サービス・コンテキスト）の取得に成功すると、ORAMTSERR_NOERROR が返されます。

備考

OraMTSSvcGet() はプール済の OCI 接続をコール側に返し、OCI を使用するデータベース・トランザクションを開始できるようにします。Microsoft Transaction Server によって調整されるトランザクションに OCI 接続を暗黙的に登録する場合は、OraMTSSvcGet() を使用します。このタイプのトランザクションでは、Microsoft Transaction Server が MS DTC コンポーネントを通じてトランザクションの作成、開始、管理、およびコミットの各フェーズを制御します。

OraMTSSvcGet() は、データベース・サーバーを Microsoft Transaction Server トランザクションに登録しない接続プールも提供します。そのためには、OraMTSSvcGet() を次のように設定します。

```
OraMTSSvcGet(...,ORAMTS_CFLG_NOIMPLICIT)
```

OraMTSSvcGet() を使用する場合は常に、OraMTSSvcRel() を使用して、終了時に接続を解放する必要があります。

注意： 接続プールは、COM コンポーネントをトランザクションに登録するかどうかにかかわらず使用されます。

SYSDBA および SYSOPER として接続するときには、それぞれ ORAMTS_CFLG_SYSDBALOGN フラグおよび ORAMTS_CFLG_SYSOPRLOGN フラグを使用します。

未登録の接続を scott/tiger アカウントで取得するには、OraMTSSvcGet() を次のようにコールします。

```
OraMTSSvcGet("SCOTT", "TIGER", "oracle", &OCISvc, &OCIEnv, ORAMTS_CFLG_ALLDEFAULT |
ORAMTS_CFLG_NOIMPLICIT);
```

OraMTSSvcGet() では、ユーザー名 (lpUsername)、パスワード (lpPsswd) およびネット・サービス名構文 (lpDbname) を一緒にユーザー名引数に指定することはできません (scott/tiger@prod_fin など)。かわりに、コール側は前述の構文例のように、lpUsername、lpPsswd および lpDbname を別々に指定する必要があります。ユーザー名およびパスワードに NULL 文字列を指定して OraMTSSvcGet() をコールすると、接続のために外部認証（オペレーティング・システム認証）が使用されます。

OraMTSSvcRel() 関数

機能

OraMTSSvcRel () は、プール済の OCI 接続（OCI サービス・コンテキスト）を解放して接続プールに戻します。OraMTSSvcGet () で取得した接続を解放するには、OraMTSSvcRel () を使用します。

構文

DWORD OraMTSSvcRel (OCISvcCtx* OCISvc);

パラメータ

表 5-5 は、OraMTSSvcRel () のパラメータを説明したものです。

表 5-5 OraMTSSvcRel() のパラメータ

データ型	パラメータ	説明
OCISvcCtx*	OCISvc (IN)	プール済の接続の OCI サービス・コンテキスト

戻り値

プール済の OCI 接続の解放に成功すると、ORAMTSERR_NOERROR が返されます。

備考

以前に OraMTSSvcGet () をコールして取得したプール済の OCI 接続が解放されて、接続プールに戻ります。解放されて接続プールに戻ると、OCI サービス・コンテキスト、その環境ハンドルおよびすべての子ハンドルは無効になります。

非トランザクションのクライアント・コンポーネントでは、OraMTSSvcGet (..., ..., ORAMTS_CFLG_ALLDEFAULT) を通じて取得した接続を解放してプールに戻す前に、OCITransCommit () または OCITransAbort () を明示的にコールする必要があります。そうしないと、そのセッションで行ったすべての変更がロールバックされます。トランザクション・コンポーネントは、Microsoft Transaction Server のオブジェクト・コンテキストに対して SetComplete メソッドまたは SetAbort メソッドを使用します。

リソースが登録されている場合、OraMTSSvcGet (..., ..., ORAMTS_CFLG_NOIMPLICIT) をコールして接続リソースを取得したコンポーネントは、まずリソースの登録を解除する必要があります。接続を明示的に登録した場合は、pTransaction->Commit () または pTransaction->Abort () をコールします。それ以外の場合は、OCITransCommit () または OCITransAbort () をコールしてから、接続を解放してプールに戻します。

OraMTSSvcEnlist() 関数

機能

OraMTSSvcEnlist() は、MS DTC によって調整されるトランザクションに OCI 接続を登録または登録解除します。このコールを使用して、プール済の接続を明示的に登録します。プールされていない接続は、OraMTSJoinTxn() を使用して登録する必要があります。

構文

```
DWORD OraMTSSvcEnlist(  
    OCISvcCtx*   OCISvc,  
    OCIError*    OCIErr,  
    void*        lpTrans,  
    unsigned     dwFlags  
);
```

パラメータ

表 5-6 は、OraMTSSvcEnlist() のパラメータを説明したものです。

表 5-6 OraMTSSvcEnlist() のパラメータ

データ型	パラメータ	説明
OCISvcCtx*	OCISvc (IN)	OraMTSSvcGet() をコールして取得したプール済の接続の OCI サービス・コンテキスト。
OCIError*	OCIErr (IN/OUT)	OCI エラー・ハンドル（無効）。
void*	lpTrans (IN)	MS DTC が制御する登録先のトランザクションへのポインタ。NULL の場合、OCI 接続は MS DTC が制御するトランザクションから登録解除されます。
unsigned	dwFlags (IN)	トランザクションへの登録に使用されるフラグ。ORAMTS_ENFLG_DEFAULT の値を使用します。登録の場合は、新しい Oracle グローバル・トランザクションを開始します。登録解除の場合は、Oracle グローバル・トランザクションから切り離し、OCI サービス・コンテキストがプールされていない接続を表す場合はコンテキスト・オブジェクトを削除します。

戻り値

成功すると、ORAMTSERR_NOERROR が返されます。

備考

このコールを使用して、プール済の接続を明示的に登録または登録解除します。プールされていない接続の登録および登録解除には、`OraMTSSvcRel()` を使用します。

`OraMTSSvcEnlist()` は、`ORAMTS_CFLG_NOIMPLICIT` フラグを指定した `OraMTSSvcGet()` を通じて以前に取得され、まだ `OraMTSSvcRel()` で解放されていないプール済の OCI 接続を登録（または登録解除）します。プール済の OCI 接続は、明示的に登録可能である必要があります。トランザクションが完了したら、次のようにトランザクション・ポインタとして `NULL` を渡し、`OraMTSSvcEnlist()` の登録を解除します。

```
OraMTSSvcEnlist (OCISvc, OCIErr, NULL, ORAMTS_ENFLG_DEFAULT)
```

`OraMTSSvcRel()` を使用して、終了時に接続を解放します。

コール側は、次の処理を実行する必要があります。

1. 接続を割り当てます。
2. 接続を登録します。
3. 処理を実行します。
4. 接続を登録解除します。
5. 接続を解放します。
6. コミットまたは異常終了します。

OraMTSSvcEnlistEx() 関数

機能

`OraMTSSvcEnlistEx()` は、OCI 接続またはサービス・コンテキストを MS DTC トランザクションに登録します。このコールは、プール済の接続の明示的な登録にのみ使用します。プールされていない接続は、`OraMTSJoinTxn()` を使用して登録する必要があります。

構文

```
DWORD OraMTSSvcEnlistEx(  
    OCISvcCtx* OCISvc,  
    OCIError*  OCIErr,  
    void*      lpTrans,  
    unsigned   dwFlags,  
    char*      lpDBName  
);
```

パラメータ

表 5-7 は、OraMTSSvcEnlistEx() のパラメータを説明したものです。

表 5-7 OraMTSSvcEnlistEx() のパラメータ

データ型	パラメータ	説明
OCISvcCtx*	OCISvc	OraMTSSvcGet() をコールして取得したプール済の接続の OCI サービス・コンテキスト。
OCIError*	OCIErr	OCI エラー・ハンドル（無効）。
void*	lpTrans	MS DTC が制御するトランザクションへのポインタ。NULL の場合、OCI 接続は MS DTC が制御するトランザクションから登録解除されます。
unsigned	dwFlags	登録フラグ。ORAMTS_ENFLG_DEFAULT の値をとります。登録の場合は、新しい Oracle グローバル・トランザクションを開始します。登録解除の場合は、Oracle グローバル・トランザクションを終了し、OCI サービス・コンテキストがプールのされていない接続を表す場合はコンテキスト・オブジェクトを削除します。
char*	lpDBName	データベースに接続するためのネット・サービス名 (Oracle Net Manager または Oracle Net Configuration Assistant で作成)。

戻り値

ORAMTSERR_ILLEGAL_OPER が返されます。

備考

プール済の接続には OraMTSSvcEnlistEx() を、プールされていない接続には OraMTSJoinTxn() を使用します。

OraMTSEnlCtxGet() 関数

機能

OraMTSEnlCtxGet () は、プールされていない OCI 接続の登録コンテキストを作成します。

構文

```
DWORD OraMTSEnlCtxGet (
    text*      lpUname,
    text*      lpPsswd,
    text*      lpDbnam,
    OCISvcCtx* pOCISvc,
    OCIError*  pOCIErr,
    ub4        dwFlags,
    void**     pCtxt
);
```

パラメータ

表 5-8 は、OraMTSEnlCtxGet () のパラメータを説明したものです。

表 5-8 OraMTSEnlCtxGet() のパラメータ

データ型	パラメータ	説明
text*	lpUname (IN)	データベース・サーバーに接続するためのユーザー名。
text*	lpPsswd (IN)	データベース・サーバーに接続するためのパスワード。
text*	lpDbnam (IN)	データベース・サーバーに接続するためのネット・サービス名。
OCISvcCtx*	pOCISvc (IN)	プールされていない接続の OCI サービス・コンテキスト。
OCIError*	pOCIErr (IN)	OCI エラー・ハンドル。
ub4	dwFlags (IN)	登録フラグ。現在指定可能な値は 0 のみです。
void**	pCtxt (OUT)	作成される登録コンテキスト。

戻り値

成功すると、ORAMTSERR_NOERROR が返されます。

備考

このコールでは、プールされていない接続の登録コンテキストが設定されます。このコールは、コール側がデータベースへの OCI 接続を確立した直後に開始する必要があります。作成されたコンテキストは、OraMTSJoinTxn() のコールに渡されます。OCI 接続を削除する前に、OraMTSEnlCtxRel() をコールして、登録コンテキストを削除します。

コール側は、次の処理を実行する必要があります。

- 1. OCI を通じてプールされていない接続を割り当てます。
- 2. OraMTSEnlCtxGet() をコールして、登録コンテキストを作成します。
- 3. OraMTSJoinTxn() をコールして、接続を確立します。
- 4. データベース処理を実行します。
- 5. NULL トランザクション・ポインタを指定した OraMTSJoinTxn() をコールして、接続を登録解除します。
- 6. 処理をコミットまたは異常終了します。
- 7. OraMTSEnlCtxRel() をコールして、登録コンテキストを解放します。
- 8. プールされていない OCI 接続を解放し、接続に関連付けられている OCI 環境ハンドルを削除します。

OraMTSEnlCtxRel() 関数

機能

OraMTSEnlCtxRel() は、以前に設定したプールされていない OCI 接続の登録コンテキストを破棄します。

構文

```
DWORD OraMTSEnlCtxRel(void* pCtxt);
```

パラメータ

表 5-9 は、OraMTSEnlCtxRel() のパラメータを説明したものです。

表 5-9 OraMTSEnlCtxRel() のパラメータ

データ型	パラメータ	説明
void*	pCtxt (IN)	破棄する登録コンテキスト

戻り値

成功すると、ORAMTSERR_NOERROR が返されます。

備考

プールされていない OCI 接続を解放する前に、クライアントが OraMTSEnlCtxRel () をコールして、その接続に対して作成した登録コンテキストをすべて破棄する必要があります。登録コンテキストでは、接続の OCI 環境ハンドルとは別に割り当てられた OCI ハンドルが保持されます。そのため、環境ハンドルが関連の登録コンテキストに対して削除されないことが必要になります。

OraMTSJoinTxn() 関数

機能

OraMTSJoinTxn () は、プールされていない OCI 接続を MS DTC トランザクションに登録します。

構文

```
DWORD OraMTSJoinTxn(void* pCtxt, void* pTrans);
```

パラメータ

表 5-10 は、OraMTSJoinTxn () のパラメータを説明したものです。

表 5-10 OraMTSJoinTxn() のパラメータ

データ型	パラメータ	説明
void*	pCtx (IN)	OCI 接続の登録コンテキスト
void*	pTrans (IN)	MS DTC トランザクション・オブジェクトの参照

戻り値

成功すると、ORAMTSERR_NOERROR が返されます。

備考

クライアントは、このコールをプールされていない OCI 接続で使用して、接続を MS DTC によって調整される トランザクションに登録します。クライアントは、MS DTC トランザクション・オブジェクトの参照とともに、OCI 接続を表す登録コンテキストの広域参照を渡します。pTrans が NULL の場合、OCI 接続は現在登録されている MS DTC トランザクションから登録解除されます。すでに登録済の OCI 接続を別の MS DTC トランザクションに登録できます。

OraMTSTransTest() 関数

機能

OraMTSTransTest() は、Microsoft Transaction Server によって開始されたトランザクション内で実行しているかどうかをテストします。

構文

```
BOOL OraMTSTransTest();
```

パラメータ

なし。

戻り値

Microsoft Transaction Server トランザクション内で実行している場合は、true が返されます。そうでない場合は、false が返されます。

備考

Microsoft Transaction Server のトランザクション・コンポーネントは、OraMTSTransTest() を使用して、あるコンポーネントが Microsoft Transaction Server トランザクションのコンテキスト内で実行されているかどうかを確認します。このコールでテストできるのは、Microsoft Transaction Server によって開始されたトランザクションのみであることに注意してください。MS DTC を直接コールして開始したトランザクションは検出されません。

OraMTSOCIErrGet() 関数

機能

OraMTSOCIErrGet() は、最後に実行した OraMTS 関数（通常は OraMTSSvcGet() または OraMTSJoinTxn()）の OCI エラー・コードおよびメッセージ・テキストがある場合に、それを取り出します。

構文

```
BOOL OraMTSOCIErrGet(DWORD* dwErr, LPCHAR lpErrMsg, DWORD* lpdLen);
```

パラメータ

表 5-11 は、OraMTSOCIErrGet () のパラメータを説明したものです。

表 5-11 OraMTSOCIErrGet() のパラメータ

データ型	パラメータ	説明
DWORD*	dwErr	エラー・コード
LPCCHAR	lpcEMsg	エラー・メッセージがある場合、そのエラーメッセージのバッファ
DWORD*	lpdLen	メッセージの実際のバイト数に設定

戻り値

OCI エラーが発生した場合は、true が返されます。そうでない場合は、false が返されます。true が返され、lpcEMsg および lpdLen が有効で、エラー・メッセージが準備されている場合、lpdLen のバイト数までが lpcEMsg にコピーされます。lpdLen は、メッセージの実際のバイト数に設定されます。

備考

OraMTSOCIErrGet () は、このスレッドで最後に実行した OraMTSSvc の OCI エラー・コードおよび OCI エラー・メッセージ・テキストがある場合に、それを取り出します。次に例を示します。

```
DWORD dwStat = OraMTSSvcGet("scott", "invalid_password","fin_prod",
db",&mysvch, &myenvh, ORAMTS_CFLG_ALLDEFAULT);
    if (dwStat != ORAMTS_ERR_NOERROR)
    {
        DWORD    dwOCIErr;
        char      errBuf[MAX_PATH];
        DWORD    errBufLen = sizeof(effBuf);

        if (OraMTSOCIErrGet(&dwOCIErr, &errBuf, &errBufLen))
            printf("OCIError %d: %s"%n);
    }
```

ODBC と Microsoft Transaction Server の統合の概要

この項では、Oracle ODBC ドライバを Microsoft Transaction Server およびデータベース・サーバーとともに使用する方法について説明します。次の項目について説明します。

- 接続属性の設定
- Oracle ODBC ドライバの使用 (推奨)
- Microsoft 社製 Oracle 用 ODBC ドライバの使用

OCI 接続プールは、5-5 ページの「[Microsoft Transaction Server アプリケーションの開発の概要](#)」で説明したとおりに動作します。この場合、ODBC を動作させるために OCI コードを変更する必要はありません。

接続属性の設定

Microsoft Transaction Server をリリース 9.2 の Oracle ODBC ドライバまたは Microsoft 社製 Oracle 用 ODBC ドライバと組み合わせて使用するには、接続属性を設定する必要があります。ODBC のコードで `SQLSetConnectAttr` 関数を使用して `SQL_ATTR_ENLIST_IN_DTC` パラメータをコールします。これにより、接続プールおよび暗黙的なトランザクション・サポートが得られます。詳細は、Microsoft Transaction Server オンライン・ヘルプの「MTS を Oracle にアクセスできるように設定する」を参照してください。

Oracle ODBC ドライバの使用

ODBC 3.0 とともに配布される ODBC ドライバ・マネージャは、接続プールをサポートするリソース・ディスペンサです。（詳細は、Microsoft Transaction Server の SDK を参照してください。）`SQLSetConnectAttr(..., SQL_ATTR_ENLIST_IN_DTC)` コールを使用して、リリース 9.2 の Oracle ODBC ドライバと ODBC 3.0 ドライバ・マネージャを統合することにより、MS DTC によって調整されるトランザクションに対して ODBC 接続の登録および登録解除を行うことができます。

リリース 9.2 の Oracle ODBC ドライバは、次のもので使用します。

- 開発するアプリケーション
- Microsoft Transaction Server に付属の Sample Bank アプリケーション（詳細は、[第 4 章「Microsoft アプリケーション・デモの実行」](#)を参照。）

リリース 9.2 の Oracle ODBC ドライバを構成するには、次のようにします。

1. 「スタート」→「設定」→「コントロール パネル」を選択します。
「コントロール パネル」ウィンドウが表示されます。
2. 「ODBC データソース」をダブルクリックします。
「ODBC データ ソース アドミニストレータ」ダイアログ・ボックスが表示されます。

- 3. 「ファイル DSN」タブを選択します。
- 4. Oracle ODBC ドライバを Microsoft の Sample Bank アプリケーション・デモで使用する場合は、次の手順 4a ～ 4d を実行します。それ以外の場合は、手順 5 に進みます。
 - a. Microsoft の mtssamples.dsn ファイルをバックアップします。このファイルは、`ROOTDRIVE:\program files\common files\odbc\data sources` にあります。
 - b. `mtssamples.dsn` を選択し、「削除」をクリックします。
 - c. 削除の確認を求めるメッセージが表示されたら、「はい」をクリックします。

これにより、Microsoft Transaction Server のサンプル・アプリケーション・デモで Microsoft 社製 ODBC ドライバを使用するための構成ファイルが削除されます。
 - d. 手順 5 に進みます。
- 5. 「追加」をクリックし、新しいファイル・データソース名 (DSN) を作成します。

「データ ソースの新規作成」ウィザードが表示されます。
- 6. 「Oracle in HOME_NAME」を選択します。
- 7. 「詳細設定」をクリックします。
- 8. 「ドライバ固有のキーワードと値を入力」フィールドに、次の情報を追加します。

```
SERVER=database_alias
USERNAME=scott
PASSWORD=tiger
```

次の表は、キーワードおよび値を説明したものです。

項目名	入力する情報
SERVER	デモがデータベース・サーバーにアクセスするときに使用するデータベース別名 (mtsdemo)
USERNAME	scott (このアプリケーションのデータベース・サーバーのユーザー名)
PASSWORD	tiger (変更していないかぎり、ユーザー名 scott に対応するデータベース・サーバーのパスワード)

- 9. 「OK」をクリックします。
- 10. 「次へ」をクリックし、「データ ソースの新規作成」ウィザードの次の画面に進みます。

11. 次の表を確認し、この接続情報を保存するファイル DSN の名前を入力します。

Oracle の ODBC を使用するアプリケーション	入力する名前
Microsoft のサンプル・アプリケーション	mtssamples.dsn (Microsoft の ODBC の名前)。この名前は、手順 4b で削除した名前と完全に一致している必要があります。
開発するアプリケーション	任意の適切なアプリケーション名。

12. 「データ ソースの新規作成」ウィザードで残りの設定を行います。
13. 「OK」をクリックし、「ODBC データ ソース アドミニストレータ」ダイアログ・ボックスを終了します。
14. 「コントロール パネル」ウィンドウを閉じます。

Microsoft 社製 Oracle 用 ODBC ドライバの使用

データベース・サーバーのリリースが 8.0.5 以下の場合は、このマニュアルで説明している統合に関する情報を利用できません。ただし、Microsoft 社製 Oracle 用 ODBC ドライバを使用すると、統合機能が利用可能になります。他の API はサポートされません。

Windows NT 4.0 Option Pack に付属する Microsoft 社製 Oracle 用 ODBC ドライバを使用すると、アプリケーションと Microsoft Transaction Server およびデータベース・サーバーとの対話が可能になります。このドライバを使用する場合、この章の後半で説明している機能は使用できません。また、次の機能は提供されません。

- パフォーマンス上の利点
- Oracle 統合でのその他の API のサポート
- Oracle9i リリース 2 (9.2) クライアントのサポート

Microsoft 社製 Oracle 用 ODBC ドライバを有効にする方法は、Microsoft Transaction Server オンライン・ヘルプの「MTS を Oracle にアクセスできるように設定する」を参照してください。そこに説明されている手順を実行した後で、さらに次の手順を実行します。

Microsoft 社製 Oracle 用 ODBC ドライバを構成するには、次のようにします。

1. Microsoft 社製 Oracle 用 ODBC ドライバが動作しているコンピュータに、Oracle Required Support Files (RSF) リリース 7.3.4 および SQL*Net 2.3 をインストールします。
2. %ORACLE_HOME%\oramtssamples\sql\omtssamp.sql スクリプトを実行します。
3. SQL*Net Easy Config を使用して、データベース別名接続を設定します。これは、mtssamples.dsn ファイルが使用する別名になります。

4. Oracle Net がインストールされているホームに RSF リリース 7.3.4 をインストールした場合は、`¥HKEY_LOCAL_MACHINE¥SOFTWARE¥ORACLE` にある次のレジストリ・パラメータを必ず設定してください。

`ORAOCI = ORA73.DLL`

OO4O と Microsoft Transaction Server の統合の概要

OO4O を使用する上での特別な要件はありません。リリース 8.1 以上の OO4O を使用してください。Microsoft Transaction Server での OO4O の使用に関する追加情報は、OO4O のオンライン・ヘルプ・ファイルを参照してください。

関連項目： 接続プールの動作（OO4O コードを変更する必要がない場合）については、5-5 ページの「[Microsoft Transaction Server アプリケーションの開発の概要](#)」を参照してください。

Oracle Provider for OLE DB と Microsoft Transaction Server の統合の概要

Microsoft Transaction Server での Oracle Provider for OLE DB の使用については、『Oracle Provider for OLE DB 開発者ガイド』を参照してください。

その他の API と Microsoft Transaction Server の統合の概要

ActiveX Data Objects などのその他の API は、リリース 8.1 以上の Oracle ODBC ドライバを使用している場合を除き、現在サポートされていません。

Microsoft Transaction Server パフォーマンス・チューニング

この章では、**Microsoft Transaction Server** のパフォーマンス・チューニングについて説明します。

この章の項目は次のとおりです。

- **Microsoft Transaction Server** アプリケーションのパフォーマンス向上
- **Microsoft Transaction Server** 接続の管理
- Windows NT でのトランザクション・タイムアウト・パラメータ値の増加
- 初期化パラメータ・ファイルの設定の変更
- MS DTC の開始

Microsoft Transaction Server アプリケーションのパフォーマンス向上

プログラミング方法を最適化すると、パフォーマンスが向上します。特定のトランザクションのコードをすべて 1 つの **Component Object Model (COM)** コンポーネントに配置した場合、そのコンポーネントをトランザクションとして指定する必要がなくなります。これにより、Microsoft Transaction Server の処理によるオーバーヘッドがなくなります。

そして、Oracle のコミットやロールバックの機能を利用して、コンポーネント内のトランザクションを制御できます。**Oracle Call Interface (OCI)** を使用している場合は、`ORAMTSSvcGet()` を使用することも可能ですが、`ORAMTS_CFLG_NOIMPLICIT` フラグも使用できます。複数の Oracle データベース・サーバーにまたがって更新する場合は、データベース・リンクを使用して、COM コンポーネントから 1 つのデータベースに接続します。

関連項目： `ORAMTSSvcGet()` の使用の詳細は、5-9 ページの「[OCI と Microsoft Transaction Server の統合の概要](#)」を参照してください。

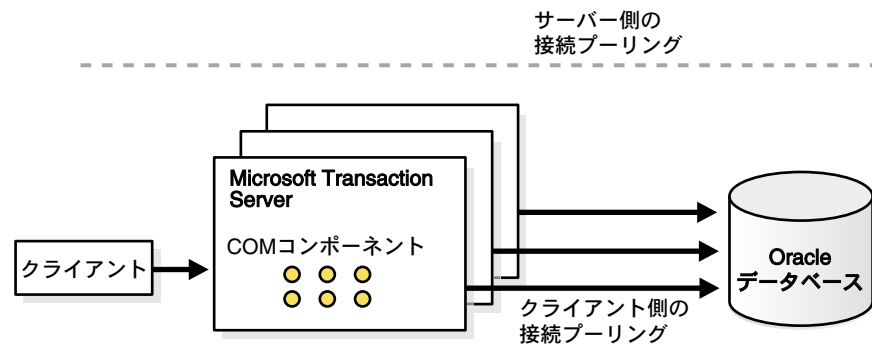
Microsoft Transaction Server 接続の管理

COM コンポーネントがデータベース・サーバーでのセッションを終了しても、デフォルトでは、接続はすぐには終了しません。かわりに、接続は接続プールでアイドル状態になり、他の COM コンポーネントがデータベース・サーバーに新たに接続するときに再利用されます。

アイドル期間中に接続の再利用が可能になることで、新規接続のオープン時にかかるリソース・コストが軽減されます。接続が接続プール内で再利用可能なアイドル状態で待機する時間の長さは、いくつかのレジストリ・パラメータの設定によって決まります。これらのパラメータは、クライアントの Microsoft Transaction Server コンポーネントがインストールされているコンピュータで変更できます。

図 6-1 は、接続プールの場所、およびクライアント側のプールに関連付けられているレジストリ・パラメータを示しています。

図 6-1 接続プールのレジストリ・パラメータ



クライアント側の接続プーリングのレジストリ・パラメータ:

ORAMTS_CONN_POOL_TIMEOUT
 ORAMTS_NET_CACHE_TIMEOUT
 ORAMTS_NET_CACHE_MAXFREE
 ORAMTS_OSCREDS_MATCH_LEVEL

クライアント側のパラメータは、
 %HKEY_LOCAL_MACHINE%SOFTWARE%ORACLE%HOME\ID
 に設定されています。

表 6-1 は、接続プールのレジストリ・パラメータについて説明したものです。

表 6-1 接続プールのレジストリ・パラメータ

クライアント側の パラメータ	説明	デフォルト値のエントリ
ORAMTS_CONN_POOL_TIMEOUT	このパラメータを使用すると、クライアント側の接続プール内で、接続が再利用可能なアイドル状態になってからタイムアウトまでの時間を設定できます。タイムアウトすると、接続は解放されます。	120 秒
ORAMTS_NET_CACHE_TIMEOUT	接続で ORAMTS_CONN_POOL_TIMEOUT に関連付けられた部分は、ユーザー名およびパスワード（つまり、ログオン・セッション）など、セッションに関する問題を処理します。接続で ORAMTS_NET_CACHE_TIMEOUT に関連付けられた部分は、データの送受信（つまり、実際の Oracle Net 接続）など、通信に関する問題を処理します。新しい Oracle Net 接続を確立するには、その接続を使用してログオン・セッションを確立する場合よりも多くのリソースが必要となります。この値は、ORAMTS_CONN_POOL_TIMEOUT に関連付けられたセッション・タイムアウト値よりも高く設定することをお勧めします。 ORAMTS_CONN_POOL_TIMEOUT で設定した時間が経過した後も、接続で ORAMTS_NET_CACHE_TIMEOUT に関連付けられた部分は少しの間利用可能です。これにより、新しいセッションを作成してサーバー接続を再利用できます。	120 秒 注意： この値は ORAMTS_CONN_POOL_TIMEOUT で設定した値に追加されます。たとえば、ORAMTS_CONN_POOL_TIMEOUT を 180 に設定し、ORAMTS_NET_CACHE_TIMEOUT を 60 に設定した場合、接続が完全に終了するまでの時間は 240 秒になります。
ORAMTS_NET_CACHE_MAXFREE	このパラメータを使用すると、常にクライアント側の接続プールで維持する最大空きサーバー接続数を設定できます。	5

表 6-1 接続プールのレジストリ・パラメータ (続き)

クライアント側の パラメータ	説明	デフォルト値のエントリ
ORAMTS_OSCREDS _MATCH_LEVEL	<p>このパラメータを使用すると、init.ora ファイルで OS_ROLES 初期化パラメータが true に設定されている場合に、接続に対して実行される Windows NT セキュリティ・チェックのレベルを設定できます。</p> <p>ユーザーがデータベース・サーバーに（たとえば CONNECT / コマンドを使用して）接続する場合、その Windows NT ユーザー名には特定のデータベース・ロールおよび権限が関連付けられます。ユーザーが切断すると接続はプールされ、利用可能なアイドル状態になります。別のユーザーが CONNECT / コマンドを入力したときは、両方のユーザーが持っている Windows NT の ID が一致する必要があります。一致しない場合には、2 番目のユーザーが最初のユーザーと同じデータベース・ロールおよび権限を取得することになります。2 番目のユーザーが CONNECT および RESOURCE というデータベース・ロールしか持っていないにもかかわらず、最初のユーザーに関連付けられた DBA データベース・ロールを偶然に取得してしまうと、セキュリティ上の問題が生じます。</p> <p>このような状況が起こらないように、パラメータを OS_AUTH_LOGIN に設定し、Windows NT のセキュリティ・チェックが必ず実行されるようにします。さらに、データベース・サーバーで OS_ROLES が true に設定されている場合は、CONNECT / と CONNECT username/password のどちらが実行されても、オペレーティング・システム・ユーザーのロールが接続に関連付けられます。この場合、Windows NT のセキュリティ・チェックを有効にするには、パラメータを ALWAYS に設定します。</p> <p>Windows NT のセキュリティ・チェックには、多くのリソース・コストがかかります。接続を再利用する前にオペレーティング・システムの資格証明を検証するため、Windows NT 関連のコストが必ず発生します。パフォーマンス上の理由から、このパラメータは NEVER に設定することをお勧めします。ただし、OS_ROLES を true に設定した場合、またはオペレーティング・システム認証による接続を使用する場合は、それに応じてこのパラメータを設定します。</p>	<p>次の 3 種類の値を設定可能</p> <ul style="list-style-type: none"> ■ ALWAYS Windows NT のセキュリティ・チェックが常に実行されます。2 番目のユーザーが、最初のユーザーのデータベース・ロールおよび権限を偶然に取得することはいできないため、この設定が最も安全です。 ■ OS_AUTH_LOGIN (デフォルト) ユーザー名およびパスワードが未入力の場合にのみ、Windows NT のセキュリティ・チェックが実行されます。これがデフォルト値です。 ■ NEVER Windows NT のセキュリティ・チェックは実行されません。この設定が、3 つの中で最もリソース・コストが低くなります。OS_ROLES を true に設定していない場合や、オペレーティング・システム認証による接続を使用していない場合は、この設定を使用してください。

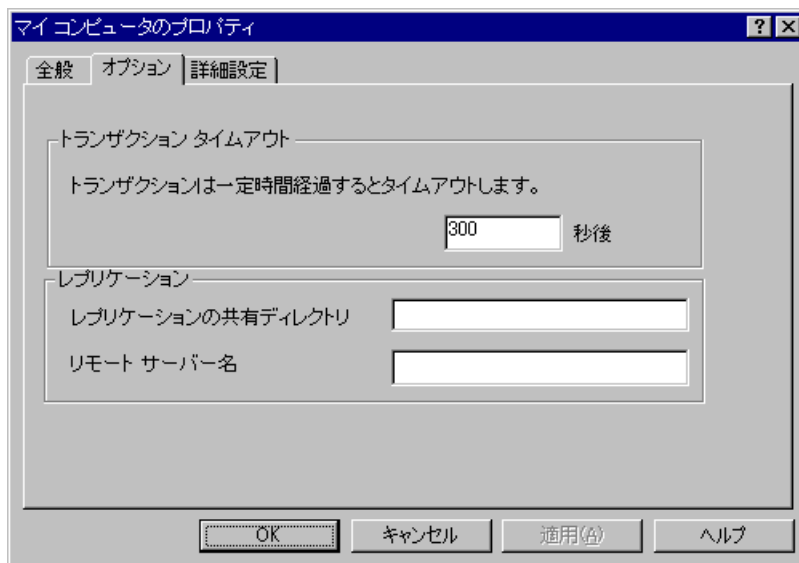
Windows NT でのトランザクション・タイムアウト・パラメータ値の増加

トランザクション要求が完了する前にタイムアウトしてしまう場合は、トランザクション・タイムアウト・パラメータの設定が低すぎる可能性があります。トランザクション・タイムアウト・パラメータ値を増やして、トランザクションの完了に十分な時間を設定します。

Windows NT でトランザクション・タイムアウト・パラメータ値を増やすには、次のようになります。

1. Microsoft Transaction Server がインストールされているコンピュータにログオンします。
2. 「スタート」 → 「プログラム」 → 「Windows NT 4.0 Option Pack」 → 「Microsoft Transaction Server」 → 「Transaction Server エクスプローラ」を選択します。
Microsoft 管理コンソールが表示されます。
3. 「Microsoft 管理コンソール」 ウィンドウで、「コンソールルート」をダブルクリックします。
4. 「Microsoft Transaction Server」をダブルクリックします。
5. 「コンピュータ」をダブルクリックします。
6. 「マイ コンピュータ」を右クリックします。
いくつかのオプションを含むメニューが表示されます。
7. 「プロパティ」を選択します。
「マイ コンピュータのプロパティ」ダイアログ・ボックスが表示されます。
8. 「オプション」タブを選択します。

9. 「トランザクション タイムアウト」フィールドに値を入力し、「OK」をクリックします。



トランザクション・タイムアウト値が増加します。通常環境では、60 秒に設定すれば十分です。ただし、トランザクションが多数の同時トランザクションと競合している場合は、この値では低すぎる場合があります。

関連項目： Windows 2000 でのトランザクション・タイムアウト・パラメータ値の増加については、[付録 A「Windows 2000 での Oracle Services for Microsoft Transaction Server の使用」](#)を参照してください。

初期化パラメータ・ファイルの設定の変更

データベース・サーバーを Microsoft Transaction Server と組み合わせて使用するには、いくつかの初期化パラメータを変更する必要がある場合があります。これらのパラメータに設定する値は、データベース処理負荷の環境によって異なります。

初期化パラメータ・ファイルの値を確認するには、次のようにします。

- 1. SYSDBA 権限があることを確認します。
- 2. データベース・サーバーがインストールされているコンピュータにログオンします。
- 3. SQL*Plus を起動します。
C:¥> sqlplus /NOLOG
- 4. データベースに SYSDBA として接続します。
SQL> CONNECT / AS SYSDBA
- 5. SESSIONS パラメータの値を確認します。
SQL> SHOW PARAMETER SESSIONS
- 6. PROCESSES パラメータの値を確認します。
SQL> SHOW PARAMETER PROCESSES

通常、Microsoft アプリケーション・デモの実行には、どちらのパラメータも現在の値が適しています。COM ベースのアプリケーションを作成して配置する場合、これらのパラメータに設定する値は、データベース環境の予想される処理負荷によって異なります。たとえば、データベース・サーバーに対して 100 の同時接続があると予想される場合は、どのようなシステム負荷にも対応できるように両方の値を 200 に設定します。これらの設定はリソースの使用量を大きく左右するため、パラメータ値の設定を高くしすぎないください。これらのパラメータについては、『Oracle9i データベース・リファレンス』を参照してください。

初期化パラメータを設定するには、次のようにします。

- 1. 次の表で、データベース・サーバーのリリースに対応する初期化パラメータ・ファイルの場所を確認してください。

使用している リリース	移動先のディレクトリ
リリース 8.1.x 以上の データベース	%ORACLE_HOME%¥admin¥DB_NAME¥pfile¥init.ora
リリース 8.0.6 の データベース	%ORACLE_HOME%¥database¥init%sid.ora

2. 初期化パラメータ・ファイルを開きます。
3. 次の初期化パラメータに、ここに記載した値以上を設定します。
 - SESSIONS = 200 (高い負荷が予想される場合はさらに大きい値)
 - PROCESSES = 200 (高い負荷が予想される場合はさらに大きい値)
4. データベース・サーバーを停止します。
`SQL> SHUTDOWN`
5. データベース・サーバーを再起動します。
`SQL> STARTUP`
6. SQL*Plus を終了します。
`SQL> EXIT`

MS DTC の開始

Oracle Services for Microsoft Transaction Server と通信するには、**Microsoft 分散トランザクション・コーディネータ (MS DTC)** が実行されている必要があります。

MS DTC を開始するには、次のようにします。

1. Microsoft Transaction Server がインストールされているコンピュータにログオンします。
2. 「スタート」 → 「プログラム」 → 「Windows NT 4.0 Option Pack」 → 「Microsoft Transaction Server」 → 「Transaction Server エクスプローラ」を選択します。
Microsoft 管理コンソールが表示されます。
3. 「Microsoft 管理コンソール」 ウィンドウで、「コンソールルート」をダブルクリックします。
4. 「Microsoft Transaction Server」をダブルクリックします。
5. 「コンピュータ」をダブルクリックします。
6. 「マイ コンピュータ」を右クリックします。
いくつかのオプションを含むメニューが表示されます。
7. 「MS DTC 開始」を選択します。

MS DTC が開始されます。

Oracle Services for Microsoft Transaction Server のトラブルシューティング

この章では、**Oracle Services for Microsoft Transaction Server (MTS)** のトラブルシューティングについて説明します。

この章の項目は次のとおりです。

- **Oracle Services for Microsoft Transaction Server** のパフォーマンスの追跡
- **Windows NT** エクスプローラの問題への対処
- 接続プールに影響する **Oracle Net** の変更の修正
- **Oracle Services for Microsoft Transaction Server** に関するよくある質問
- **Microsoft Transaction Server** 管理者ユーザー・アカウントの削除

Oracle Services for Microsoft Transaction Server のパフォーマンスの追跡

トレース・ファイルには、Oracle Services for Microsoft Transaction Server のパフォーマンスに関する情報が記録されます。この情報には、次のものが含まれます。

- エラー
- 登録要求とその結果
- 準備、コミットおよび異常終了の要求とその結果

oramts.dll 内のトレースを処理するレジストリ・パラメータが 2 つあります。
oramts.dll では、次の処理が実行されます。

- Oracle データベース・サーバーを Microsoft Transaction Server と統合するための API を実装します。
- リソース・ディスペンサとして機能し、プール済の **Oracle Call Interface (OCI)** 接続を提供します。
- プールされていない OCI 接続を持つクライアントを、**Microsoft 分散トランザクション・コーディネータ (MS DTC)** によって開始されたトランザクションに登録できるようにします。
- Oracle Services for Microsoft Transaction Server と通信して、MS DTC によって開始されたトランザクションにデータベース・サーバーを登録します。

表 7-1 は、トレースを処理するレジストリ・パラメータを説明したものです。どちらのパラメータも、まだ設定されていない場合は Oracle Services for Microsoft Transaction Server のインストール中に %HKEY_LOCAL_MACHINE%SOFTWARE%ORACLE%HOMEID に自動的に設定されます。これらのパラメータを環境変数として設定するのではなく、レジストリ・パラメータを使用してください。環境変数を設定した場合、変更を有効にするためにコンピュータを再起動する必要があります。

表 7-1 トレース・レジストリ・パラメータ

パラメータ	説明	データ型	デフォルト値
ORAMTS_CP_TRACE_LEVEL	oramts.dll 内のリソース・ディスペンサ・レイヤーをトレースします。トレース・ファイル名は、次のようになります。 oramtsccpid.trc pid はプロセスの識別番号です。新しいトレース情報は、常にファイルの末尾に追加されます。	REG_SZ	0
ORAMTS_CP_TRACE_DIR	トレース・ファイルが生成される出力ディレクトリを設定します。	REG_SZ	%ORACLE_HOME%\oramts\trace

表 7-2 は、ORAMTS_CP_TRACE_LEVEL トレース値の範囲を示しています。
ORAMTS_CP_TRACE_LEVEL は、トレースが必要な場合のみ、0 以外の値に設定します。

表 7-2 トレース・レジストリ・パラメータの値

レベル	説明
0	トレースを無効化 ¹
1	エラーのみトレース
2	エラーに加えて重要なイベントをトレース
4	関数の入力と出力、重要なイベント、およびエラーをトレース
5	参照カウント関数と、コンストラクタおよびデストラクタの入力と出力をトレース

¹ レジストリ・パラメータがレジストリにも、環境変数としても設定されていない場合は、トレースが無効（レベルを 0 に設定した場合と同様）になります。また、現時点ではレベル 3 がサポートされていないことにも注意してください。このパラメータを 3 に設定すると、かわりにレベル 2 のトレースが有効になります。

注意： Oracle MTS Recovery Service では、
%ORACLE_HOME%\oramts\trace ディレクトリにもトレース・ファイル出力を生成します。

Windows NT エクスプローラの問題への対処

Microsoft Transaction Server をデータベース・サーバーと組み合わせて使用していて、Windows NT エクスプローラがクラッシュしたり、その他の予期しない Windows NT の問題が発生する場合は、(Microsoft 社より提供される) Windows NT 4.0 Service Pack 5 以上をインストールしてください。

接続プールに影響する Oracle Net の変更の修正

ORAMTS レイヤー (つまり oramts.dll) が提供する接続プールでは、接続の **ネット・サービス名** を使用して、アプリケーションのプール済の接続を識別します。ネット・サービス名を変更したときに、現在プール済の接続が存在する場合は、接続プールを使用しているアプリケーションを停止して、再起動する必要があります。この変更には、tnsnames.ora ファイルでのネット・サービス名のホストまたはデータベース・システム識別子 (SID) の変更も含まれます。

この変更により、現在プール済の、旧ネット・サービス名に対応するすべての接続が破棄され、新たにプールされる接続は変更されたネット・サービス名を使用するようになります。この中には、**Microsoft Transaction Server** コンポーネントのホストになっているあらゆるアプリケーションが含まれます。

接続プールを空にするには、次のようにします。

次の表に示した手順を実行します。

アプリケーションのタイプ	操作
アウトプロセスの Microsoft Transaction Server コンポーネント (サーバー・パッケージ)	次のアプリケーションを実行します。 <code>C:¥> mtxsstop</code> これにより、接続プールが空になります。
インプロセスの Microsoft Transaction Server コンポーネント (ライブラリ・パッケージ)	アプリケーションを終了します。これにより、接続プールも空になります。

Oracle Services for Microsoft Transaction Server に関するよくある質問

この項では、一般的な質問とその回答を示します。

Q: 複数のデータベース・サーバーがある場合、アプリケーションをどのように設計すればよいですか。

A: Oracle クライアントは、2 種類の方法でデータベース・サーバーへの接続を確立できます。

- 通常の Oracle クライアントは、専用サーバー構成を使用してデータベース・サーバーへの接続を確立します。専用サーバー構成では、1 つのクライアントが 1 つの Oracle サーバー・プロセスに対応します。
- 高負荷の環境で拡張性を得るために、Oracle クライアントは共有サーバー構成を使用できます。共有サーバー構成では、1 つの Oracle サーバー・プロセスを複数のクライアント接続で共有することが可能です。

Microsoft Transaction Server は、分散トランザクションを通じてデータベース・サーバーと通信します。専用サーバー構成では、他のデータベース・サーバーからの分散更新（データベース・リンクをまたがる [データ操作言語（DML）](#) 文）は使用できません。ただし、データベース・サーバーへの元の接続が共有サーバー構成を使用して確立された場合は、他のデータベース・サーバーからの分散更新が可能です。

共有サーバー構成で DML 文を使用するには、tnsnames.ora ファイルに次のパラメータを設定します。

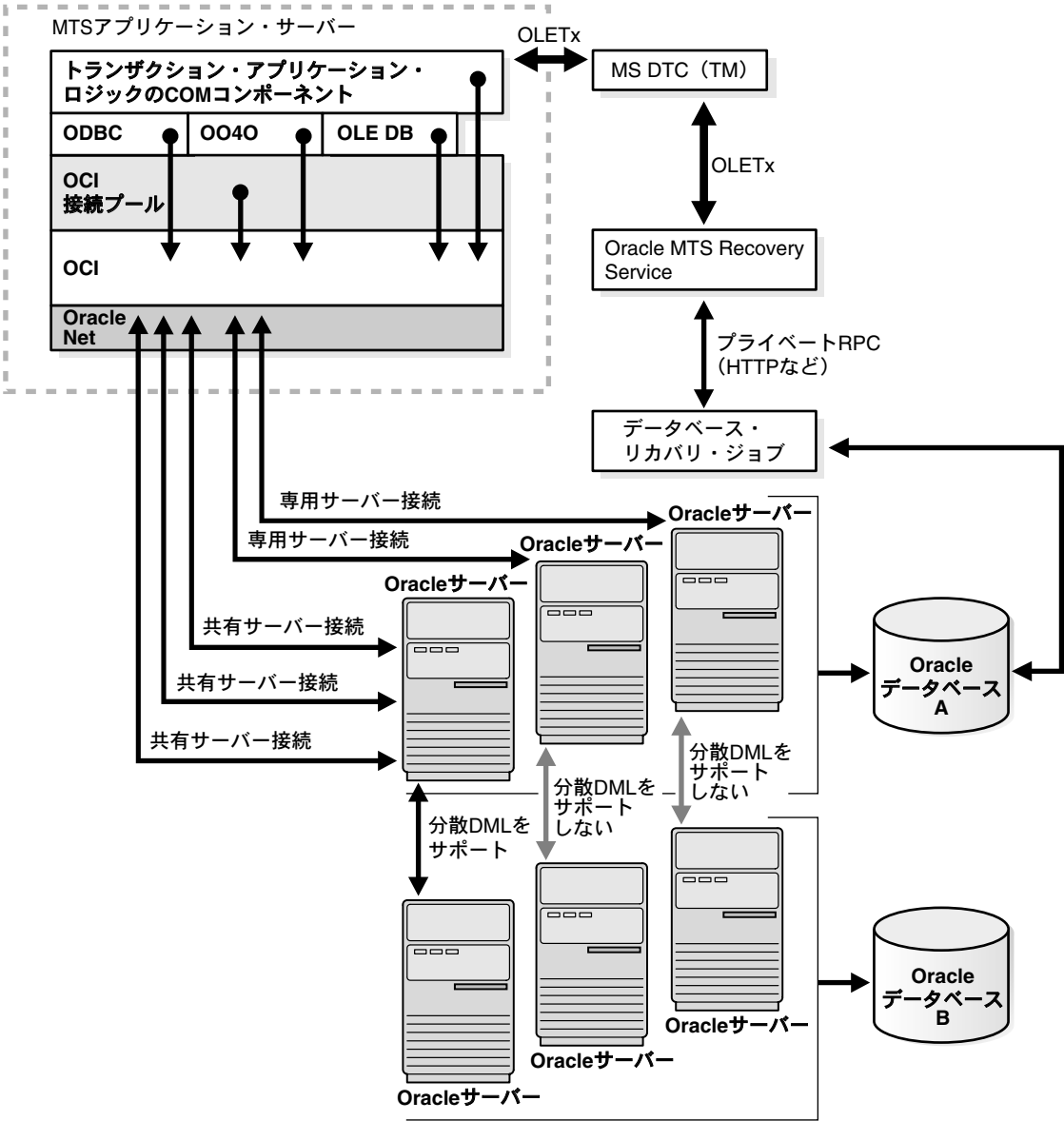
```
SERVER=dedicated
```

これにより、Oracle Net Listener が専用接続を提供します。

関連資料: 『Oracle9i Net Services 管理者ガイド』

図 7-1 は、このプロセスを示しています。

図 7-1 MTS アプリケーションからの分散 DML 文



Q: Oracle Net 接続プーリング、OCI 接続プーリング、および Microsoft Transaction Server 接続プーリングの違いは何でしょうか。

A: Oracle Net 接続プーリングはサーバー側の機能で、データベース・サーバーが共有サーバー・サポート用に構成されている場合にのみ実装されます。Oracle Net 接続プーリングを使用すると、共有サーバーに対する物理的なネットワーク接続数を最小限にすることができます。これは、ディスパッチャの接続を複数のクライアント・プロセス間で共有することによって実現されます。

Microsoft Transaction Server は、メモリーやデータベース接続などの特定のリソースのプールを可能にする、リソース・プーリング・インフラストラクチャを提供します。OCI 接続プーリング・レイヤーでは、Microsoft Transaction Server のリソース・プーリングと連携して、プール済の Oracle クライアント / サーバー・セッションを提供します。また、OCI 接続プーリング・レイヤーでは Oracle Net 接続のキャッシュも行い、クライアント / サーバー・セッションの設定時間を短縮します。

Q: インダウト・トランザクションとは何ですか。

A: Oracle では、次の構成で分散トランザクションを使用します。

- 分散データベース構成（データベース・リンクを使用した分散更新など）
- トランザクションの結果を調整するための外部トランザクション・マネージャ（Tuxedo、MS DTC など）

2 フェーズ・コミット・プロトコルにより、これらのトランザクションが完了します。フェーズ 1 では、トランザクション・マネージャ（TM）が、TM のトランザクションに関与する各種リソース・マネージャに、基礎となる分散トランザクションの準備を要求します。フェーズ 2 では、TM がトランザクションをコミットするか異常終了するかを決定し、リソース・マネージャに基礎となるトランザクションのコミットまたは異常終了を要求します。リソース・マネージャがフェーズ 2 の通知の受信に失敗した場合、基礎となる分散トランザクションはインダウトになります。

Oracle を Microsoft Transaction Server と統合するために、データベースで分散トランザクションが使用されます。分散トランザクションは、MS DTC によって調整されるトランザクションに対応します。トランザクションがコミットまたは異常終了できない（2 フェーズ・コミットのフェーズ 2）場合に、分散トランザクションがインダウトになる可能性があります。これは、Microsoft Transaction Server アプリケーション・サーバー・プロセス、データベース・サーバーまたはネットワークで障害が発生した場合に起こります。

関連項目： [第 3 章「リカバリのシナリオ管理」](#)

Microsoft Transaction Server 管理者ユーザー・アカウントの削除

Microsoft Transaction Server 管理者ユーザー・アカウントは、Oracle9i リリース 2 (9.2) で `oramtsadmin.sql` スクリプトを実行すると作成されます。Microsoft Transaction Server でトランザクションを調整しているデータベース・サーバーを後から変更した場合、旧データベース・サーバーから管理者ユーザー・アカウントのスキーマを削除できます。

Microsoft Transaction Server 管理者ユーザー・アカウントを削除するには、次のようにします。

1. SQL*Plus を起動します。

```
c:\¥> sqlplus /NOLOG
```

2. データベース・サーバーに SYSDBA として接続します。

```
SQL> CONNECT / AS SYSDBA
```

3. 次のコマンドを入力して、管理者ユーザー・アカウントのスキーマを削除します。

```
SQL> DROP USER mtsadmin_username CASCADE;
```

`mtsadmin_username` は Microsoft Transaction Server 管理者ユーザー・アカウントです (デフォルトは `mtssys`)。

4. 新規データベース・サーバーの Microsoft Transaction Server 管理者ユーザー・アカウントの作成については、[第 3 章「リカバリのシナリオ管理」](#)を参照してください。

Windows 2000 での Oracle Services for Microsoft Transaction Server の使用

この付録では、**Oracle Services for Microsoft Transaction Server (MTS)** を Windows NT で使用した場合と Windows 2000 で使用した場合の違いについて説明します。

この付録の項目は次のとおりです。

- **Windows NT と Windows 2000 での Microsoft Transaction Server の違い**
- **Windows 2000 でのトランザクション・タイムアウト・パラメータ値の増加**

注意： Oracle Services for Microsoft Transaction Server の構成は、Windows 2000 でも Windows NT でも同じです。

Windows NT と Windows 2000 での Microsoft Transaction Server の違い

表 A-1 は、**Microsoft Transaction Server** を Windows NT で使用した場合と Windows 2000 で使用した場合の違いについて説明したものです。

表 A-1 Windows NT と Windows 2000 での Microsoft Transaction Server の違い

機能	Windows NT の場合	Windows 2000 の場合
Microsoft Transaction Server	Microsoft Transaction Server はアドオンである。	Microsoft Transaction Server の機能は、COM+ サーバー (COM+ トランザクション) としてオペレーティング・システム内に統合されている。
Microsoft 管理コンソール	Microsoft 社から Microsoft 管理コンソールを入手する必要がある。	Microsoft 管理コンソールは標準で組み込まれている。
コンポーネント	COM。	COM+。
トランザクション・アプリケーションのインストールおよび構成	「スタート」→「プログラム」→「Windows NT 4.0 Option Pack」→「Microsoft Transaction Server」→「Transaction Server エクスプローラ」を選択する。	「スタート」→「プログラム」→「管理ツール」→「コンポーネント サービス」を選択する。
トランザクション・コーディネータ	Microsoft 分散トランザクション・コーディネータ (MS DTC)。	分散トランザクション・コーディネータ (DTC)。
Microsoft アプリケーション・デモ	Microsoft アプリケーション・デモの Visual C/C++ Sample Bank アプリケーションが Microsoft Transaction Server に付属している。	Microsoft アプリケーション・デモは付属していない。

Windows 2000 でのトランザクション・タイムアウト・パラメータ値の増加

トランザクション要求が完了する前にタイムアウトしてしまう場合は、トランザクション・タイムアウト・パラメータの設定が低すぎる可能性があります。トランザクション・タイムアウト・パラメータ値を増やして、トランザクションの完了に十分な時間を設定します。

Windows 2000 でトランザクション・タイムアウト・パラメータ値を増やすには、次のようにします。

1. Microsoft Transaction Server がインストールされている Windows 2000 コンピュータにログオンします。
2. 「スタート」→「プログラム」→「管理ツール」→「コンポーネント サービス」を選択します。

Microsoft 管理コンソールが表示されます。

3. 「Microsoft 管理コンソール」ウィンドウで、「コンソールルート」をダブルクリックします。
4. 「コンポーネント サービス」をダブルクリックします。
5. 「コンピュータ」をダブルクリックします。
6. 「マイ コンピュータ」を右クリックします。
いくつかのオプションを含むメニューが表示されます。
7. 「プロパティ」を選択します。
「マイ コンピュータのプロパティ」ダイアログ・ボックスが表示されます。
8. 「オプション」タブを選択します。
9. 「トランザクション タイムアウト」フィールドに値を入力し、「OK」をクリックします。

トランザクション・タイムアウト値が増加します。通常環境では、60 秒に設定すれば十分です。ただし、トランザクションが多数の同時トランザクションと競合している場合は、この値では低すぎる場合があります。

用語集

Component Object Model (COM)

オブジェクト間通信を、個々のオブジェクトの記述に使用されるプログラム言語の種類に関係なく実現するためのバイナリ標準。

Distributed Component Object Model (DCOM)

オブジェクトがネットワークを介して他のオブジェクトと対話できるようにするための COM 拡張。

JOB_QUEUE_PROCESSES

この初期化パラメータでは、インスタンスで起動されるジョブ・キュー・プロセスの数を指定する。ジョブ・キュー・プロセスを実行するには、このパラメータを 1 以上に設定する必要がある。

listener.ora

リスナーについての次の項目を示すリスナー構成ファイル。

- 一意な名前
- 接続要求を受け入れるプロトコル・アドレス
- リスニング対象のサービス

Microsoft Transaction Server

インターネットまたはネットワーク・サーバー上で動作する、COM ベースのトランザクション処理システム。

Microsoft アプリケーション・デモ (Microsoft application demo)

Windows NT 版の Microsoft Transaction Server に付属している Visual C++ Sample Bank パッケージの Oracle Call Interface (OCI) 実装。

Microsoft 管理コンソール (Microsoft Management Console)

スナップインと呼ばれる管理ツールのホストとして機能するアプリケーション。Microsoft 管理コンソール自体は、機能を提供しない。

Microsoft 分散トランザクション・コーディネータ (Microsoft Distributed Transaction Coordinator: MS DTC)

トランザクション処理の中心となるのは、Microsoft 分散トランザクション・コーディネータと呼ばれる Microsoft Transaction Server のコンポーネント。

mtssys

デフォルトの Microsoft Transaction Server 管理者ユーザー名。リリース 8.1 以下の Oracle8i では、これは Oracle Service for MTS のユーザー名だった。

Optimal Flexible Architecture (OFA)

Oracle ソフトウェアおよびデータベースのファイルのネーミングおよび配置に関する一連のガイドライン。

Oracle Call Interface (OCI)

データベース・サーバー内のデータおよびスキーマの操作を可能にする API。OCI プログラムは、データベース以外のアプリケーションの場合と同じ方法でコンパイルおよびリンクする。前処理やプリコンパイルを別途行う必要はない。

Oracle Fail Safe

あるクラスタ・ノードで障害が発生した場合に、そのノードで実行されているデータベース・サーバーおよびアプリケーションを、稼働しているノードへ自動的に素早くフェイルオーバー（移動）する。

Oracle Manager for MTS Services スナップイン (Oracle Manager for MTS Services snap-in)

Oracle9i リリース 1 (9.0.1) 以降より、この Microsoft 管理コンソール・スナップインは、Oracle Services for Microsoft Transaction Server で不要となったため、使用できない。

Oracle MTS Recovery Service

Oracle MTS Recovery Service は、失敗したトランザクションが開始されたコンピュータで、インダウト・トランザクションを解決する。Microsoft Transaction Server 対応の各データベース・サーバーに対してスケジューリングされたリカバリ・ジョブによって、Oracle MTS Recovery Service がインダウト・トランザクションを解決できるようになる。

Oracle Objects for OLE (OO4O)

Object Linking and Embedding (OLE) インプロセス・サーバーに結合されたカスタム・コントロール (OCX または ActiveX) で、Oracle データベースのネイティブ機能を Windows アプリケーションにプラグインできるようにする。

Oracle ODBC ドライバ (Oracle Open Database Connectivity Driver)

Oracle ODBC ドライバは、1 つのアプリケーションから多数の異なるデータ・ソースへのアクセスを可能にする標準インタフェースを提供する。アプリケーションのソース・コードをデータ・ソースごとにコンパイルしなおす必要はない。データベース・ドライバにより、アプリケーションが特定のデータ・ソースにリンクされる。データベース・ドライバは、アプリケーションが特定のデータ・ソースにアクセスするために必要に応じて起動できる DLL である。したがって、アプリケーションはデータベース・ドライバが存在するすべてのデータ・ソースにアクセスできる。

Oracle Provider for OLE DB

アプリケーション、コンパイラおよびその他のデータベース・コンポーネントから Oracle データへの高性能な効率の良いアクセスを提供するインタフェース。

Oracle Service for MTS

この Windows NT サービスは、Oracle9i リリース 1 (9.0.1) から、Oracle Services for Microsoft Transaction Server では不要であり、使用できない。

Oracle Services for Microsoft Transaction Server (MTS)

リリース 8.0.6 以上の Oracle データベースを Microsoft Transaction Server に完全に統合するコンポーネント。このコンポーネントにより、Microsoft Transaction Server を使用して COM ベースのアプリケーションを開発および配置することが可能になる。

ORAMTS_ORADB

Oracle Service for MTS を Oracle データベースに接続するためのデータベース別名 (Oracle9i リリース 1 (9.0.1) から、Oracle Services for Microsoft Transaction Server ではサポートされない)。

SYSDBA

ADMIN OPTION および SYSOPER システム権限とすべてのシステム権限を持つ特別なデータベース管理ロール。SYSDBA では、CREATE DATABASE アクションおよび時間ベースのリカバリも許可されている。

SYSOPER

データベース管理者が、STARTUP、SHUTDOWN、ALTER DATABASE OPEN/MOUNT、ALTER DATABASE BACKUP、ARCHIVE LOG および RECOVER を実行することを許可する、特別なデータベース管理ロール。RESTRICTED SESSION 権限も含まれる。

tnsnames.ora

ネット・サービス名にマップされる接続記述子を含むファイル。すべてのクライアントまたは各クライアントで使用するために、このファイルを集中してメンテナンスすることも、ローカルでメンテナンスすることもできる。

原子性、一貫性、独立性および永続性 (Atomicity, Consistency, Isolation, and Durability: ACID)

ACID は、トランザクション・マネージャ（トランザクション・モニターとも呼ぶ）によってすべてのトランザクションに提供される 4 つの基本的な属性からなる。

データ操作言語 (data manipulation language)

データベースのデータの問合せおよび更新を行う SQL 文のカテゴリ。一般的な DML 文には、SELECT、INSERT、UPDATE および DELETE がある。

トランザクション識別子 (transaction identifier: XID)

トランザクションが開始されたクライアント・コンピュータを示す。

ネット・サービス名 (net service name)

Oracle Net 接続に対する Oracle Net サーバー、および特定のシステム識別子 (SID) またはデータベースを示すためにクライアントが使用する名前。ネット・サービス名は、ポート番号とプロトコルにマップされる。ネット・サービス名は、接続文字列、データベース別名、ホスト文字列またはサービス名とも呼ばれる。

Oracle Net サーバーのみではなく、接続先の特定の SID またはデータベース・サーバーも示す。

リソース・マネージャ (resource manager: RM)

Microsoft Transaction Server は、データベース・サーバーがリソース・マネージャ (RM) として機能できるようにトランザクション・プロセスに登録する。

索引

数字

2 フェーズ・コミット・プロトコル, 7-7

3 層アーキテクチャ, 1-2, 1-4, 5-2

A

account.idl ファイル, 4-5

account 表

Microsoft アプリケーション・デモでの使用, 4-2

C

COM, 「Component Object Model (COM)」を参照

Component Object Model (COM)

Microsoft Transaction Server 環境への登録, 5-5

Microsoft Transaction Server 内での実行

調整されたトランザクション, 5-7

Microsoft 社製 Oracle 用 ODBC ドライバおよび

Microsoft Transaction Server との使用, 5-27

MS DTC によって調整されるトランザクション内で

の実行, 5-8

ODBC, 2-4

ODBC および Microsoft Transaction Server を使用
したプログラミング, 5-25

Oracle Call Interface および Microsoft Transaction
Server を使用したプログラミング, 5-9

Oracle Objects for OLE, 2-4

Oracle Objects for OLE および Microsoft
Transaction Server を使用したプログラミング,
5-28

Oracle ODBC ドライバおよび Microsoft Transaction
Server との使用, 5-25

Oracle Provider for OLE DB および Microsoft
Transaction Server を使用したプログラミング,
5-28

Windows 2000, A-2

コンポーネントのトランザクション化, 1-2

トランザクション内での統合, 5-2

CONNECT ロール, 3-6

D

DBMS_JOBS パッケージ, 3-6

DBMS_TRANSACTION パッケージ, 3-6

DCOM, 「Distributed Component Object Model
(DCOM)」を参照

Distributed Component Object Model (DCOM), 1-2

DML, 「データ操作言語 (DML)」を参照

F

FORCE_ANY_TRANSACTION 権限, 3-6

J

JOB_QUEUE_PROCESSES 初期化パラメータ, 3-5

M

Microsoft Developer Studio

Microsoft アプリケーション・デモでの使用, 4-6

Microsoft Internet Information Server

インストール要件, 2-4

Microsoft Transaction Server

3 層アーキテクチャ・モデル, 1-2

COM コンポーネントの登録, 5-5

- Microsoft アプリケーション・デモでの Oracle Call Interface の使用, 4-2
- Microsoft 社製 Oracle 用 ODBC ドライバの使用, 5-27
- Microsoft 社製 Oracle 用 ODBC ドライバを使用したプログラミング, 5-25
- MS DTC によって調整されるトランザクション内で実行されるコンポーネント, 5-8
- MS DTC の開始, 6-9
- MS DTC の役割, 5-4
- Oracle Call Interface の役割, 5-4
- Oracle Call Interface を使用したプログラミング, 5-9
- Oracle Objects for OLE の役割, 5-4
- Oracle Objects for OLE を使用したプログラミング, 5-28
- Oracle ODBC ドライバの使用, 5-25
- Oracle ODBC ドライバの役割, 5-4
- Oracle ODBC ドライバを使用したプログラミング, 5-25
- Oracle Provider for OLE DB の役割, 5-4
- Oracle Provider for OLE DB を使用したプログラミング, 5-28
- Oracle データベース・サーバーとの統合, 1-2
- Oracle データベース・サーバーの役割, 5-4
- Oracle データベースに関する参照先, 1-5
- Windows 2000 でのタイムアウト・パラメータ値の増加, A-3
- Windows NT でのタイムアウト・パラメータ値の増加, 6-6
- 以前のリリースからの移行, 2-5
- インストール要件, 2-3, 2-4
- 管理者ユーザー・アカウントの作成, 3-6
- 管理者ユーザー名の変更, 3-6
- クライアント・コンピュータの役割, 5-4
- 接続プールの役割, 5-4
- 定義, 1-2
- データベース・リンク, 7-5
- トランザクション内で実行される COM コンポーネント, 5-7
- トランザクション・リカバリのスケジューリング, 3-4
- 複数のデータベースを使用したアプリケーションの設計, 7-5
- 利点, 1-2
- Microsoft Transaction Server がインストールされているコンピュータ
 - インストール要件, 2-4
 - 構成要件, 3-2
- Microsoft Transaction Server のデモ
 - Microsoft アプリケーション・デモでの Oracle ODBC ドライバの使用, 4-7
 - Microsoft アプリケーション・デモでの Oracle Provider for OLE DB の使用, 4-7
 - Oracle Call Interface の使用, 4-2
- Microsoft アプリケーション・デモ
 - Microsoft Developer Studio の使用, 4-6
 - mtsdemousr ユーザー名の使用, 4-4
 - omtssamp.sql スクリプトの実行, 4-4
 - Oracle Call Interface の使用, 4-2
 - Oracle ODBC ドライバの使用, 4-7
 - Oracle Provider for OLE DB の使用, 4-7
 - Oracle データベースに適切な表が含まれていることの確認, 4-3
 - Visual C++ の使用, 4-5
 - Windows 2000, A-2
 - インストール先, 4-2
 - インストールの確認, 4-5
 - 実行, 4-5
 - データベース内容の概要, 4-2
- Microsoft 管理コンソール
 - Windows 2000, A-2
 - インストール方法, 2-4
- Microsoft 分散トランザクション・コーディネータ (MS DTC)
 - MS DTC によって調整されるトランザクション内で実行される COM コンポーネント, 5-8
 - Oracle Service for MTS での使用, 2-3
 - Windows 2000, A-2
 - 開始, 6-9
 - クラスタ内, 3-11
 - 登録, 5-18
 - 役割, 5-2, 5-4
- MS DTC, 「Microsoft 分散トランザクション・コーディネータ (MS DTC)」を参照
- mtsdemousr ユーザー名
 - Microsoft アプリケーション・デモの使用, 4-4
- mtsdemo ユーザー名, 4-2
 - account 表および receipt 表の使用, 4-2
- Microsoft アプリケーション・デモでの使用, 4-2
- mtssamples.dsn ファイル
 - Oracle ODBC ドライバの使用, 5-26
- mtssys ユーザー名
 - デフォルトの管理者ユーザー・アカウント, 3-6

パスワードの変更, 3-6
mtxstop.exe ファイル
実行, 7-4

O

OCI, 「Oracle Call Interface (OCI)」を参照
OCI_THREADED フラグ
渡す, 5-10
OCIInitialize 関数
コール, 5-10
ODBC, 「Open Database Connectivity (ODBC)」を参照
OLE DB, 「Oracle Provider for OLE DB」を参照
omtssamp.sql スクリプト, 4-4, 5-27
OO4O, 「Oracle Objects for OLE (OO4O)」を参照
Open Database Connectivity (ODBC)
Microsoft Transaction Server での Microsoft 社製
Oracle 用 ODBC ドライバの構成, 5-27
Microsoft Transaction Server での Microsoft 社製
Oracle 用 ODBC ドライバの使用, 5-27
Microsoft Transaction Server での Oracle ODBC ド
ライバ, 5-4
Microsoft Transaction Server での Oracle ODBC ド
ライバの構成, 5-25
Microsoft Transaction Server での Oracle ODBC ド
ライバの使用, 5-25
Microsoft Transaction Server を使用したプログラミ
ング, 5-25
Microsoft アプリケーション・デモでの Oracle
ODBC ドライバの使用, 4-7
Oracle ODBC ドライバでの mtssamples.dsn ファイ
ルの使用, 5-26
Oracle ODBC ドライバのインストール要件, 2-4
Oracle ODBC ドライバの役割, 5-4
SQL_ATTR_ENLIST_IN_DTC パラメータの使用,
5-25
SQLSetConnectAttr 関数の使用, 5-25
接続属性の設定, 5-25
Oracle Call Interface (OCI)
Microsoft Transaction Server を使用したプログラミ
ング, 5-9
Microsoft アプリケーション・デモでの使用, 4-2
MS DTC によって調整されるトランザクションの登
録, 5-17, 5-18
MS DTC によって調整されるトランザクションの登
録解除, 5-17

OCITransCommit() および OCITransAbort() の使用
に関する注意, 5-9

Oracle Call Interface、Oracle Net および Microsoft
Transaction Server の各接続プーリングの違い,
7-7

OraMTSEnlCtxGet() 関数, 5-20
OraMTSEnlCtxGet() 関数のパラメータ, 5-20
OraMTSJoinTxn() 関数, 5-22
OraMTSJoinTxn() 関数のパラメータ, 5-22
OraMTSOCIErrGet() 関数, 5-23
OraMTSOCIErrGet() 関数のパラメータ, 5-24
OraMTSSvcEnlist() 関数, 5-17
OraMTSSvcEnlist() 関数のパラメータ, 5-17
OraMTSSvcEnlistEx() 関数, 5-18
OraMTSSvcEnlistEx() 関数のパラメータ, 5-19
OraMTSSvcGet() 関数, 5-13
OraMTSSvcGet() 関数のパラメータ, 5-13
OraMTSSvcRel() 関数, 5-16
OraMTSSvcRel() 関数のパラメータ, 5-16
OraMTSTransTest() 関数, 5-23
OraMTSTransTest() 関数のパラメータ, 5-23
サンプル・ファイルの場所, 5-11
プール済の Oracle Call Interface 接続の解放, 5-16
プール済の Oracle Call Interface 接続の取得, 5-15
プール済または標準の Oracle Call Interface 接続の
取得, 5-18, 5-20
役割, 5-4
Oracle Fail Safe
インストール要件, 2-3
レジストリ・パラメータの変更, 3-11
Oracle Manager for MTS Services スナップイン
不要, xv
Oracle MTS Recovery Service
インストール, 2-3
インダウト・トランザクションの解決, 3-3
トレース・ファイル出力, 7-3
Oracle Net Manager
インストール要件, 2-4
Oracle Objects for OLE (OO4O)
Microsoft Transaction Server, 5-4
Microsoft Transaction Server を使用したプログラミ
ング, 5-28
インストール要件, 2-4
役割, 5-4
Oracle ODBC ドライバ, 「Open Database Connectivity
(ODBC)」を参照
Oracle Provider for OLE DB

- Microsoft Transaction Server との統合, 5-4
- Microsoft Transaction Server を使用したプログラミング, 5-28
- Microsoft アプリケーション・デモでの使用, 4-7
- 役割, 5-4
- Oracle Service for MTS
 - MS DTC での使用, 2-3
 - Oracle9i で廃止, xiv
 - 削除, 2-6
 - 削除の準備, 2-6
 - 非アクティブ・ユーザーのロールおよび権限の削除, 2-10
 - レジストリでの Oracle Service for MTS の変更, 2-11
- Oracle Services for Microsoft Transaction Server
 - インストール可能な製品, 3-2
 - インストール方法, 2-4
 - インストール要件, 2-4
- Oracle データベースがインストールされているコンピュータ
 - インストール要件, 2-4
 - 構成要件, 3-2
- Oracle データベース・サーバー
 - init.ora ファイルのパラメータ設定の変更, 6-8
 - Microsoft Transaction Server との統合, 1-2
 - インストール要件, 2-4
 - 役割, 5-4
 - リリース 8.0.5 以下のリリースとの Microsoft Transaction Server の統合不可, 2-4
- oramts_2pc_pending
 - ビュー, 3-9
- ORAMTS_CFLG_ALLDEFAULT フラグ
 - 説明, 5-14
- ORAMTS_CFLG_NOIMPLICIT フラグ
 - 説明, 5-14
- ORAMTS_CFLG_PRELIMAUTH フラグ
 - 説明, 5-14
- ORAMTS_CFLG_SYSDBALOGN フラグ
 - 使用, 5-15
 - 説明, 5-14
- ORAMTS_CFLG_SYSOPRLOGN フラグ
 - 使用, 5-15
 - 説明, 5-14
- ORAMTS_CFLG_UNIQUESVR フラグ
 - 説明, 5-14
- ORAMTS_CONN_POOL_TIMEOUT レジストリ・パラメータ, 6-4

- ORAMTS_CP_TRACE_DIR レジストリ・パラメータ, 7-2
- ORAMTS_CP_TRACE_LEVEL レジストリ・パラメータ, 7-2
- ORAMTS_ENFLG_DEFAULT フラグ
 - 説明, 5-17, 5-19
- ORAMTS_NET_CACHE_MAXFREE レジストリ・パラメータ, 6-4
- ORAMTS_NET_CACHE_TIMEOUT レジストリ・パラメータ, 6-4
- ORAMTS_ORADB レジストリ・パラメータ, 2-12
- ORAMTS_OSCREDS_MATCH_LEVEL レジストリ・パラメータ, 6-5
- ORAMTS_SUNAME レジストリ・パラメータ, 2-12
- ORAMTS_SUPWD レジストリ・パラメータ, 2-12
- oramtsadmin.sql スクリプト
 - Microsoft Transaction Server 管理者ユーザー・アカウントの作成, 3-6
 - PL/SQL パッケージの作成, 3-6
- oramts.dll ファイル
 - 定義, 7-2
- OraMTSEnlCtxGet() 関数
 - Oracle Call Interface 関数, 5-20
 - プール済または標準の Oracle Call Interface 接続の登録, 5-20
- OraMTSEnlCtxRel() 関数
 - ORAMTSERR_NOERROR を返す, 5-22
 - 以前に設定した登録コンテキストの破棄, 5-21
 - 構文, 5-21
 - パラメータ, 5-21
- ORAMTSERR_ILLEGAL_OPER
 - 接続取得時の戻り値, 5-19
- ORAMTSERR_NOERROR
 - 接続解放時の戻り値, 5-16
 - 接続取得時の戻り値, 5-15, 5-17, 5-22
- OraMTSJoinTxn() 関数, 5-10
 - Oracle Call Interface 関数, 5-22
 - 構文, 5-22
 - 接続取得時に ORAMTSERR_NOERROR を返す, 5-22
 - パラメータ, 5-22
 - プールされていない Oracle Call Interface 接続の登録, 5-22
- OraMTSOCIErrGet() 関数, 5-23
 - Oracle Call Interface エラー・コードの取出し, 5-23
 - Oracle Call Interface 関数, 5-23
 - 構文, 5-23

パラメータ, 5-24
OraMTSSvcEnlist() 関数
 Oracle Call Interface 関数, 5-17
 ORAMTS_ENFLG_DEFAULT フラグ, 5-17
 構文, 5-17
 使用の制限, 5-17
 接続取得時に ORAMTSERR_NOERROR を返す,
 5-17
 パラメータ, 5-17
 プール済または標準の Oracle Call Interface 接続の
 登録, 5-18
OraMTSSvcEnlistEx() 関数
 Oracle Call Interface 関数, 5-18
 OraMTSJoinTxn() 関数との違い, 5-19
 構文, 5-18
 使用の制限, 5-18
 接続取得時に ORAMTSERR_ILLEGAL_OPER を返
 す, 5-19
 パラメータ, 5-19
OraMTSSvcGet() 関数
 Oracle Call Interface 関数, 5-13
 ORAMTS_CFLG_ALLDEFAULT フラグ, 5-14
 ORAMTS_CFLG_NOIMPLICIT フラグ, 5-14
 ORAMTS_CFLG_PRELIMAUTH フラグ, 5-14
 ORAMTS_CFLG_SYSDBALOGN フラグ, 5-14
 ORAMTS_CFLG_SYSOPRLOGN フラグ, 5-14
 ORAMTS_CFLG_UNIQUESVR フラグ, 5-14
 概要, 5-9, 5-13
 構文, 5-13
 接続取得時に ORAMTSERR_NOERROR を返す,
 5-15
 パラメータ, 5-13
 プール済の接続を返す, 5-15
 役割, 5-13
OraMTSSvcRel() 関数
 Oracle Call Interface 関数, 5-16
 概要, 5-9
 構文, 5-16
 接続解放時に ORAMTSERR_NOERROR を返す,
 5-16
 パラメータ, 5-16
 プール済の接続の解放, 5-16
OraMTSTransTest() 関数
 Oracle Call Interface 関数, 5-23
 構文, 5-23
 パラメータ, 5-23
ORAOCI レジストリ・パラメータ

設定, 5-28
OS_ROLES 初期化パラメータ, 6-5

P

PROCESSES 初期化パラメータ
 値の変更, 6-8
prvtoramts.plb ファイル, 3-4

R

receipt 表
 Microsoft アプリケーション・デモでの使用, 4-2
recover_automatic
 パブリック・プロシージャ, 3-8
revokeuser.sql スクリプト
 実行, 2-10
RM, 「リソース・マネージャ (RM)」を参照

S

SELECT_CATALOG_ROLE ロール, 3-6
Service Pack 5.0 以上
 Windows NT エクスプローラのクラッシュへの対
 処, 7-4
 インストール要件, 2-4
SESSIONS 初期化パラメータ
 値の変更, 6-8
show_indoubt
 パブリック・プロシージャ, 3-7
SNP プロセス
 起動, 3-4
SQL*Plus
 インストール要件, 2-4

T

tnsnames.ora ファイル
 エントリが正しいデータベースを指定していること
 を確認, 2-11
 共有サーバー構成用の設定, 7-5

U

- utl_oramts PL/SQL パッケージ
 - パブリック・プロシージャの提供, 3-7
- utl_oramts.forget_RMs
 - パブリック・プロシージャ, 3-9
- utl_oramts.sql スクリプト, 3-4
 - 権限およびロールの付与, 3-6

V

- vcacct.dll ファイル, 4-2
 - ビルド, 4-6
- Visual C++
 - Microsoft アプリケーション・デモでの使用, 4-5

W

- Windows 2000
 - Microsoft アプリケーション・デモの組込みなし, 4-1
 - Windows NT で Microsoft Transaction Server を使用した場合との違い, A-2
- Windows NT
 - Windows 2000 で Microsoft Transaction Server を使用した場合との違い, A-2
- Windows NT エクスプローラ
 - クラッシュへの対処, 7-4

い

- 移行
 - Oracle Services for Microsoft Transaction Server の
以前のリリースから, 2-5
 - 要件, 2-5
- インストール
 - Oracle MTS Recovery Service, 2-3
 - 同一コンピュータ上の Microsoft Transaction Server
および Oracle データベース, 2-2
 - 別々のコンピュータ上の Microsoft Transaction
Server および Oracle データベース, 2-2
- インストール要件
 - Microsoft Internet Information Server, 2-4
 - Microsoft Transaction Server, 2-3, 2-4
 - Microsoft Transaction Server がインストールされて
いるコンピュータ, 2-4
 - Oracle Fail Safe が稼働中のコンピュータ, 2-3

- Oracle Net Manager, 2-4
- Oracle Objects for OLE, 2-4
- Oracle ODBC ドライバ, 2-4
- Oracle Services for Microsoft Transaction Server,
2-4
- Oracle データベースがインストールされているコン
ピュータ, 2-4
- Oracle データベース・サーバー, 2-4
- Service Pack 5.0 以上, 2-4
- SQL*Plus, 2-4
- クライアント・コンピュータ, 2-4
- クライアント用の Oracle Net Services, 2-4
- サーバー用の Oracle Net Services, 2-4
- 必要な RAM, 2-3
- インダウト・トランザクション
- JOB_QUEUE_PROCESSES 初期化パラメータ, 3-5
- SNP プロセスの起動, 3-4
- 解決, 3-3
- 自動リカバリのスケジューリング, 3-4
- 定義, 7-7
- 表示, 3-10

か

- 環境ハンドル, 5-13
- 管理者ユーザー名
 - Microsoft Transaction Server, 3-6
 - 削除, 7-8

き

- 機能, 新しい, xiv
- 共有サーバー構成, 7-5

く

- クライアント, 2-4
 - 構成要件, 3-2
- クライアント用の Oracle Net Services
 - インストール要件, 2-4

け

権限

- Oracle Service for MTS の非アクティブ・ユーザーの権限の削除, 2-10
- utl_oramts.sql スクリプト, 3-6
- 管理者ユーザー・アカウント, 3-6

こ

構成要件

- Microsoft Transaction Server がインストールされているコンピュータ, 3-2
- Oracle Fail Safe 用の変更, 3-11
- Oracle データベースがインストールされているコンピュータ, 3-2
- クライアント・コンピュータ, 3-2

さ

サーバー用の Oracle Net Services

- インストール要件, 2-4

サービス

- 削除, 2-6

サービス・ハンドル, 5-13

参照先

- Microsoft Transaction Server および Oracle データベース, 1-5

サンプル

- Microsoft アプリケーション・デモ, 4-2, 4-7

し

初期化パラメータ

- JOB_QUEUE_PROCESSES, 3-5
- OS_ROLES, 6-5
- PROCESSES, 6-8
- SESSIONS, 6-8

新機能, xiv

せ

接続

- 接続プールの管理, 6-3

接続属性

- ODBC での設定, 5-25

接続プール

Microsoft Transaction Server との統合, 5-4

Oracle Call Interface、Microsoft Transaction Server および Oracle Net の各接続プーリングの違い, 7-7

OraMTSSvcGet() 関数の使用方法, 5-13

環境ハンドルの取得, 5-13

クライアント側のレジストリ・パラメータ, 6-3

サービス・ハンドルの取得, 5-13

接続の解放, 5-16

接続の管理, 6-3

接続プールを空にする, 7-4

役割, 5-4

た

タイムアウト・パラメータ

- Microsoft Transaction Server での増加, 6-6

ち

チューニング

init.ora パラメータ SESSIONS および PROCESSES の変更, 6-8

Windows 2000 でのトランザクション・タイムアウト・パラメータ値の増加, A-3

接続プールの管理, 6-3

トランザクション・タイムアウト・パラメータ値の増加, 6-6

パフォーマンスの向上, 6-2

て

データ操作言語 (DML)

- 共有サーバー構成での使用, 7-5

データベース・リンク

- Microsoft Transaction Server, 7-5

デモ

Microsoft アプリケーション・デモ, 4-2, 4-7

Microsoft アプリケーション・デモのインストールの確認, 4-5

Microsoft アプリケーション・デモの実行, 4-5

mtsdemour ユーザー名の使用, 4-4

omtssamp.sql スクリプトの実行, 4-4

Oracle データベースに適切な表が含まれていることの確認, 4-3

Visual C++ の使用, 4-5

インストール先のディレクトリ, 4-2

と

- トラブルシューティング
 - MS DTC の開始, 6-9
 - Windows NT エクスプローラのクラッシュへの対処, 7-4
 - 管理者ユーザー・アカウントの削除, 7-8
 - 接続プールに影響する Oracle Net の変更の修正, 7-4
 - トランザクション・タイムアウト・パラメータ値の増加, 6-6
 - トランザクション・リカバリ, 3-10
 - トレース・ファイルの使用, 7-2
- トランザクション
 - COM コンポーネントの統合の概要, 5-2
 - トランザクション・タイムアウト・パラメータ値の増加, 6-6
- トランザクション・アプリケーション
 - Windows 2000, A-2
- トランザクション・リカバリ
 - JOB_QUEUE_PROCESSES 初期化パラメータ, 3-5
 - Oracle Fail Safe 環境, 3-3
 - SNP プロセスの起動, 3-4
 - 概要, 3-3
 - スケジューリング, 3-4
 - トラブルシューティング, 3-10
- トレース・ファイル
 - Oracle MTS Recovery Service, 7-3
 - oramts.dll, 7-2
 - 使用, 7-2
 - ファイル名の規則, 7-2
 - リカバリ成功メッセージの監視, 2-8
 - レジストリの設定, 7-2

ね

- ネット・サービス名
 - 接続プールに影響する変更, 7-4

は

- パスワード
 - mtssys ユーザー名に対する変更, 3-6
- パフォーマンス
 - 向上, 6-2
- パブリック・プロシージャ
 - recover_automatic, 3-8

- show_indoubt, 3-7
- utl_oramts.forget_RMs, 3-9
- 提供, 3-7

ひ

- ビュー
 - oramts_2pc_pending, 3-9

ふ

- プールされていない Oracle Call Interface 接続
 - OraMTSJoinTxn 関数, 5-22
- プール済の接続
 - 解放, 5-16
- プログラミング
 - Oracle Provider for OLE DB および Microsoft Transaction Server, 5-28
 - プログラミング方法
 - パフォーマンス向上のための最適化, 6-2
 - 分散トランザクション
 - インダウト, 7-7

ゆ

- ユーザー名
 - Microsoft アプリケーション・デモでの使用, 4-6

り

- リカバリ
 - インダウト・トランザクション, 3-3
- リソース・マネージャ (RM)
 - 役割, 5-2, 5-4

れ

- レジストリ
 - Oracle Fail Safe 構成用の値の変更, 3-11
 - Oracle Service for MTS のデータベース接続の変更, 2-11
 - トレース・ファイルの設定, 7-2
- レジストリ・パラメータ
 - Oracle Fail Safe 用の変更, 3-11
 - ORAMTS_CONN_POOL_TIMEOUT, 6-4
 - ORAMTS_CP_TRACE_DIR, 7-2
 - ORAMTS_CP_TRACE_LEVEL, 7-2

ORAMTS_NET_CACHE_MAXFREE, 6-4
ORAMTS_NET_CACHE_TIMEOUT, 6-4
ORAMTS_ORADB, 2-12
ORAMTS_OSCREDS_MATCH_LEVEL, 6-5
ORAMTS_SUNAME, 2-12
ORAMTS_SUPWD, 2-12

ろ

ロール

Oracle Service for MTS の非アクティブ・ユーザー
のロールの削除, 2-10
管理者ユーザー・アカウント, 3-6

