

Oracle® Services for Microsoft Transaction Server 開発者ガイド 19c for Microsoft Windows

F21044-01(原本部品番号:E96617-01)

2019年5月

タイトルおよび著作権情報

Oracle Services for Microsoft Transaction Server開発者ガイド, 19c for Microsoft Windows
F21044-01

Copyright © 1996, 2019, Oracle and/or its affiliates. All rights reserved.

原著者: Maitreyee Chaliha

原協力著者: Janis Greenberg, Patricia Huey, Mark Kennedy, Roza Leyderman, Janelle Simmons, Alex Keh, Valarie Moore, Vivek Raja, Eric Wang, Yong Hu

このソフトウェアおよび関連ドキュメントの使用と開示は、ライセンス契約の制約条件に従うものとし、知的財産に関する法律により保護されています。ライセンス契約で明示的に許諾されている場合もしくは法律によって認められている場合を除き、形式、手段に関係なく、いかなる部分も使用、複写、複製、翻訳、放送、修正、ライセンス供与、送信、配布、発表、実行、公開または表示することはできません。このソフトウェアのリバース・エンジニアリング、逆アセンブル、逆コンパイルは互換性のために法律によって規定されている場合を除き、禁止されています。

ここに記載された情報は予告なしに変更される場合があります。また、誤りが無いことの保証はいたしかねます。誤りを見つけた場合は、オラクル社までご連絡ください。

このソフトウェアまたは関連ドキュメントを、米国政府機関もしくは米国政府機関に代わってこのソフトウェアまたは関連ドキュメントをライセンスされた者に提供する場合は、次の通知が適用されます。

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

このソフトウェアもしくはハードウェアは様々な情報管理アプリケーションでの一般的な使用のために開発されたものです。このソフトウェアもしくはハードウェアは、危険が伴うアプリケーション(人的傷害を発生させる可能性があるアプリケーションを含む)への用途を目的として開発されていません。このソフトウェアもしくはハードウェアを危険が伴うアプリケーションで使用する場合、安全に使用するために、適切な安全装置、バックアップ、冗長性(redundancy)、その他の対策を講じることは使用者の責任となります。このソフトウェアもしくはハードウェアを危険が伴うアプリケーションで使用したことに起因して損害が発生しても、オラクル社およびその関連会社は一切の責任を負いかねます。

OracleおよびJavaはOracle Corporationおよびその関連企業の登録商標です。その他の名称は、それぞれの所有者の商標または登録商標です。

Intel, Intel Xeonは、Intel Corporationの商標または登録商標です。すべてのSPARCの商標はライセンスをもとに使用し、SPARC International, Inc.の商標または登録商標です。AMD、Opteron、AMDロゴ、AMD Opteronロゴは、Advanced Micro Devices, Inc.の商標または登録商標です。UNIXは、The Open Groupの登録商標です。

このソフトウェアまたはハードウェア、そしてドキュメントは、第三者のコンテンツ、製品、サービスへのアクセス、あるいはそれらに関する情報を提供することがあります。お客様との間に適切な契約が定められている場合を除いて、オラクル社およびその関連会社は、第三者のコンテンツ、製品、サービスに関して一切の責任を負わず、いかなる保証もいたしません。お客様との間に適切な契約が定められている場合を除いて、オラクル社およびその関連会社は、第三者のコンテンツ、製品、サービスへのアクセスま

たは使用によって損失、費用、あるいは損害が発生しても一切の責任を負いかねます。

目次

- [図一覧](#)
- [表一覧](#)
- [タイトルおよび著作権情報](#)
- [はじめに](#)
 - [対象読者](#)
 - [ドキュメントのアクセシビリティについて](#)
 - [関連ドキュメント](#)
 - [表記規則](#)
- [Oracle Services for Microsoft Transaction Server開発者ガイドのこのリリースでの変更点](#)
 - [Oracle Services for Microsoft Transaction ServerのODAC 12cリリース3での変更点](#)
 - [新機能](#)
 - [Oracle Database 12cリリース1 \(12.1\)の変更内容](#)
 - [新機能](#)
- [1 Oracle DatabaseとMicrosoft Transaction Serverの連携](#)
 - [Microsoft Transaction Serverの概要](#)
 - [Microsoft Transaction ServerとOracleの統合の概要](#)
 - [Oracle Services for Microsoft Transaction ServerによるDTCのサポート](#)
 - [Real Application Clusters\(Oracle RAC\)の分散トランザクション](#)
 - [昇格可能ローカル・トランザクション](#)
 - [読取りコミット済トランザクションとシリアライズ可能トランザクション](#)
 - [Microsoft Transaction ServerおよびOracleに関する参照先](#)
- [2 Oracle Services for Microsoft Transaction Serverのインストール](#)
 - [インストールの概要](#)
 - [Microsoft Transaction Serverのインストール要件](#)
 - [Oracle製品](#)
 - [Oracle以外の製品](#)
 - [分散トランザクション](#)
 - [Microsoft Distributed Transaction Coordinatorの統合](#)
 - [ODP.NET, Managed Driverの設定](#)
 - [ODP.NET, Unmanaged Driverの設定](#)
 - [Oracle MTS Recovery Serviceの手動作成](#)
- [3 リカバリのシナリオ管理](#)
 - [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の自動トランザクション・リカバリのスケジューリング](#)
 - [アクセス制御リスト\(ACL\)の作成](#)
 - [自動トランザクション・リカバリの構成](#)
 - [データベースのジョブ・キュー・プロセスの設定および起動](#)
 - [自動トランザクション・リカバリの作成とスケジューリング](#)
 - [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](#)
 - [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](#)
- [4 Microsoft Transaction ServerおよびOracle Database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](#)
 - [Microsoft Transaction Serverアプリケーションの開発](#)
 - [Microsoft Transaction Serverコンポーネントの登録](#)
 - [登録コンポーネントのタイプ](#)
 - [コンポーネントの登録](#)
 - [Microsoft Transaction Serverにより調整されるコンポーネントのトランザクション](#)
 - [Microsoft DTCにより調整されるコンポーネントのトランザクション](#)
 - [Microsoft Transaction ServerとのOCI統合](#)
 - [COMコンポーネントの統合](#)
 - [MTS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](#)
 - [OCI接続プールで実行される、トランザクション以外のCOMコンポーネント](#)
 - [MS DTCおよびOCI接続プールを使用するCOMコンポーネント](#)
 - [MS DTCおよびプールされていないOCI接続を使用するCOMコンポーネント](#)
 - [OCI関数の使用方法](#)
 - [OraMTSSvcGet\(\)](#)
 - [OraMTSSvcRel\(\)](#)
 - [OraMTSSvcEnlist\(\)](#)
 - [OraMTSSvcEnlistEx\(\)](#)
 - [OraMTSEnCtxGet\(\)](#)
 - [OraMTSEnCtxRel\(\)](#)
 - [OraMTSJoinTxn\(\)](#)
 - [OraMTSTransTest\(\)](#)
 - [OraMTSOCIErrGet\(\)](#)
 - [ODBCとMicrosoft Transaction Serverの統合の概要](#)
 - [接続属性の設定](#)
 - [Oracle ODBCドライバの使用](#)
 - [Microsoft社製Oracle用ODBCドライバの使用](#)
- [5 Microsoft Transaction Serverのパフォーマンス・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](#)
 - [Microsoft Transaction Server接続の管理](#)
 - [接続プールのレジストリ・パラメータ](#)
 - [トランザクション・タイムアウト・パラメータ値の増加](#)
 - [初期化パラメータの設定の変更](#)
 - [追加のパラメータ](#)
 - [MSDTCの開始](#)
- [6 Oracle Microsoft Transaction Serverのトラブルシューティング](#)
 - [Oracle Services for Microsoft Transaction Serverパフォーマンスの追跡](#)
 - [接続プールに影響するOracle Netの変更の修正](#)
 - [複数のデータベースを使用するアプリケーションの設計](#)
 - [様々な種類の接続プールの使用](#)
 - [インダウト・トランザクションの使用](#)

- [Microsoft Transaction Server管理者ユーザー・アカウントの削除](#)
- [用語集](#)
- [索引](#)

図一覧

- [4-1 トランザクション内でのコンポーネント統合](#)
- [6-1 MTSアプリケーションからの分散DML文](#)

表一覧

- [2-1 分散トランザクションでサポートされるODP.NETのタイプと.NET Frameworkのバージョン](#)
- [4-1 MTSおよびOracle Databaseを統合するためのOCI関数のサマリー](#)
- [4-2 OraMTSSvcGet\(\)パラメータ](#)
- [4-3 OraMTSSvcRel\(\)パラメータ](#)
- [4-4 OraMTSSvcEnlist\(\)パラメータ](#)
- [4-5 OraMTSSvcEnlistEx\(\)パラメータ](#)
- [4-6 OraMTSEnlCtxGet\(\)パラメータ](#)
- [4-7 OraMTSEnlCtxRel\(\)パラメータ](#)
- [4-8 OraMTSJoinTxn\(\)パラメータ](#)
- [4-9 OraMTSOCIErrGet\(\)パラメータ](#)
- [6-1 ORAMTS_CP_TRACE_LEVELトレース・レジストリ・パラメータの値](#)

はじめに

このマニュアルでは、オペレーティング・システムに適用されるOracle Services for [Microsoft Transaction Server](#)をインストール、構成、使用および管理する方法について説明します。ここでは、Windows 2000、Windows XPおよびWindows Server 2003オペレーティング・システムに適用される、Oracle Databaseソフトウェアの機能について説明します。

内容は次のとおりです。

- [対象読者](#)
- [ドキュメントのアクセシビリティについて](#)
- [関連ドキュメント](#)
- [表記規則](#)

対象読者

このガイドは、次のタスクを実行する方を対象としています。

- Microsoft Transaction Serverとともに[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
- Oracle Services for Microsoft Transaction Serverとともに.NETアプリケーションを使用してOracle Databaseインスタンスにアクセス。

ドキュメントのアクセシビリティについて

Oracleのアクセシビリティについての詳細情報は、Oracle Accessibility ProgramのWebサイト (<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>)を参照してください。

Oracle Supportへのアクセス

サポートを購入したオラクル社のお客様は、My Oracle Supportを介して電子的なサポートにアクセスできます。詳細情報は (<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info>)か、聴覚に障害のあるお客様は (<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs>)を参照してください。

関連ドキュメント

詳細は、次のOracleドキュメントを参照してください。

- [『Oracle Databaseリファレンス』](#)
- [『Oracle Provider for OLE DB開発者ガイド for Microsoft Windows』](#)
- [Oracle Data Provider for .NET開発者ガイドfor Microsoft Windows](#)
- [『Oracle Database Net Services管理者ガイド』](#)
- [『Oracle Databaseプラットフォーム・ガイドfor Microsoft Windows』](#)

Oracleエラー・メッセージの情報は、[『Oracle Databaseエラー・メッセージ』](#)を参照してください。特定の範囲を表示したら、特

定のメッセージを検索できます。インターネットに接続している場合、Oracleオンライン・ドキュメントのエラー・メッセージ検索機能を使用して、特定のエラー・メッセージを検索できます。

このマニュアルに含まれる例の多くでは、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・スキーマを使用しています。これらのスキーマがどのように作成されているか、およびその使用方法については、『[Oracle Databaseサンプル・スキーマ](#)』を参照してください。

リリース・ノート、インストール関連ドキュメント、ホワイト・ペーパーまたはその他の関連ドキュメントは、OTN-J (Oracle Technology Network Japan)から、無償でダウンロードできます

<http://www.oracle.com/technetwork/index.html>

このマニュアルを含むOracleマニュアルの最新バージョンは、次のWebサイトを参照してください。

<http://www.oracle.com/technetwork/indexes/documentation/index.html>

表記規則

このマニュアルでは次の表記規則を使用します。

規則	意味
太字	太字は、操作に関連する Graphical User Interface 要素、または本文中で定義されている用語および用語集に記載されている用語を示します。
イタリック体	イタリックは、ユーザーが特定の値を指定するプレースホルダ変数を示します。
固定幅フォント	固定幅フォントは、段落内のコマンド、URL、サンプル内のコード、画面に表示されるテキスト、または入力するテキストを示します。

Oracle Services for Microsoft Transaction Server開発者ガイドのこのリリースでの変更点

ここでは、次の内容について説明します。

- [Oracle Services for Microsoft Transaction ServerのODAC 12cリリース3での変更点](#)
- [Oracle Database 12c リリース1 \(12.1\)での変更点](#)

Oracle Services for Microsoft Transaction ServerのODAC 12cリリース3での変更点

ODAC 12cリリース3のOracle Services for Microsoft Transaction Server開発者ガイドでは、次の点に変更されました。

新機能

このリリースの新機能は次のとおりです。

- Oracle.ManagedDataAccessDTC.dllのない分散トランザクション

Oracle.ManagedDataAccessDTC.dllアセンブリは、.NET Framework 4.5.2以上およびODP.NET管理対象ドライバで実行している分散トランザクション・アプリケーションで必要なくなりました。ODP.NETインストールにおいて、Oracle.ManagedDataAccessDTC.dllはグローバル・アセンブリ・キャッシュ(GAC)に含まれなくなりました。.NET Framework 4.5.1以前を使用するアプリケーションでは、Oracle.ManagedDataAccessDTC.dllはアプリケーション・ディレクトリまたはGACに置く必要があります。

Oracle Database 12cリリース1 (12.1)での変更点

Oracle Database 12cリリース1 (12.1)のOracle Services for Microsoft Transaction Server開発者ガイドでは、次の点に変更されました。

新機能

このリリースの新機能は次のとおりです。

- Oracleホーム・ユーザーとしてのOraMTSサービスの実行

OraMTSサービスは、指定したより権限の低いユーザー(OraMTSホーム・ユーザー)として実行します。インストーラによってではなく、サービスを手動で作成する必要がある場合のためのユーティリティが提供されています。現在、各Oracleホームには、独自のOraMTSサービスがあります。

関連項目:

[Oracle MTS Recovery Serviceの手動作成](#)

1 Oracle DatabaseとMicrosoft Transaction Serverの連携

これらのトピックでは、[Microsoft Transaction Server](#)とOracle Databaseの統合について説明します。

- [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では、クライアント・コンピュータにかわって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デプロイおよび管理します。Microsoft Transaction Serverによって提供される機能は次のとおりです。

- 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デプロイ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[Atomicity_ Consistency_ Isolation_ and Durability \(ACID\)](#)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。このモデルにより、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。Microsoft Transaction Server機能は、COM+およびEnterprise Servicesにも実装されています。Microsoft Transaction Server、COM+およびEnterprise Servicesは、Oracle Services for Microsoft Transaction Server([OraMTS](#))によりサポートされます。

関連項目:

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、Oracle Databaseに接続されたMicrosoft Transaction Serverで、Win32、Win64、COMまたは[Microsoft .NET](#)を使用して作成したアプリケーションをデプロイできます。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\(OraMTS\)](#)をインストールする必要があります。

- Win32、Win64、COM、または.NETアプリケーション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します。このためには、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・ボックスを使用します。
- 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Oracle Databaseインスタンスに残っているインダウト・トランザクションのリカバリに役立ちます。接続されているデータベースそれぞれについて、次の操作を行います。

- Microsoft Transaction Server管理者ユーザー・アカウントを作成します。
- データベース・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を行います。

これにより、データベースを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 Data Provider for .NET\(ODP.NET\)](#)
- [Oracle ODBCドライバ\(Oracle Open Database Connectivity Driver\)](#)
- [Oracle Provider for OLE DB](#)
- [Oracle Call Interface\(OCI\)](#)

Oracle Services for Microsoft Transaction ServerによるDTCのサポート

Oracle Services for Microsoft Transaction Serverは、Windowsオペレーティング・システムの一部であるMicrosoft分散トランザクション・コーディネータ(DTC)と連携して動作します。DTCは、トランザクションに関与するすべてのデータ・リソースで、トランザクションの結果に一貫性があるかどうかを確認するために、2フェーズ・コミット・プロトコルを実装しています。

Real Application Clusters(Oracle RAC)の分散トランザクション

Oracle Databaseリリース11.1では、データベースは分散トランザクションのすべてのブランチを、単一のOracle RACインスタンスに自動的にリダイレクトします。以前は、開発者が個別にすべてのブランチを単一のOracle RACインスタンスにリダイレクトし、手動でこのプロセスを管理する必要がありました。

関連項目:

Real Application Clustersの分散トランザクションの詳細は、『*Oracle Real Application Clusters*管理およびデプロイメント・ガイド』を参照してください。

昇格可能ローカル・トランザクション

昇格可能ローカル・トランザクションでは、複数のデータベースがトランザクションに入れられるまで、すべてのトランザクションをローカルに保持できます。トランザクションに入れられた時点で、分散トランザクションに昇格します。

昇格可能トランザクション機能の柔軟性により、トランザクション・アプリケーションでリソースの使用効率を向上できます。分散トランザクションは、ローカル・トランザクションに比べて大幅なオーバーヘッドが必要です。そのため、1つのデータベースのみを使用している場合は、ローカル・トランザクションの方が優れています。設計時には、トランザクションがローカルになる場合と分散させる場合とが不明なことがあります。この機能以前は、ローカル・トランザクションがほとんどであるとしても、開発者は常に分散トランザクションを使用する必要があったため、不要なリソースの使用が生じていました。

この機能は、Oracle Database 11g リリース1以上でサポートされています。以前のバージョンのデータベースおよびその他のリソース・マネージャは、最初にOracle Database 11g リリース1以上のデータソースに接続されているかぎり、昇格可能トランザクションに入ることができます。

関連項目:

System. Transactionsサポートの詳細は、[『Oracle Data Provider for .NET開発者ガイド』\(Microsoft Windows用\)](#)を参照してください。

読取りコミット済トランザクションとシリアライズ可能トランザクション

Oracle Services for Microsoft Transaction Serverでは、シリアライズ可能または読取りコミット済の分離レベルに対して、分散トランザクションのセットがサポートされるようになりました。

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

これで、データベースでMicrosoft Transaction Serverを使用する準備が整いました。すぐに使用を開始するには、次のようになります。

1. Microsoft Transaction Serverとデータベースの統合に必要なOracle製品およびMicrosoft製品をインストールします。

[Oracle Services for Microsoft Transaction Serverのインストール](#)を参照してください。

2. Microsoft Transaction Server管理者ユーザー・アカウントを作成します。

[リカバリのシナリオ管理](#)を参照してください。

3. Microsoft Transaction Server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を行います。

[リカバリのシナリオ管理](#)を参照してください。

4. Microsoft Transaction Serverをホストとするアプリケーションを作成します。

COMベースのアプリケーションでの、OCI、Oracle ODBCドライバまたはOracle Provider for OLE DBの使用方法については、[Microsoft Transaction ServerおよびOracle Databaseを使用したプログラミング](#)を参照してください。

5. Microsoft Transaction Serverを異なるWindowsオペレーティング・システムで使用した場合の違いを確認します。

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

これらのトピックでは、[Microsoft Transaction Server](#)およびOracle Database環境のためのインストール要件について説明します。

- [インストールの概要](#)
- [Microsoft Transaction Serverのインストール要件](#)
- [Oracle Data Provider for .NET_ Managed DriverでのOracle Services for MTSの使用](#)
- [Oracle MTS Recovery Serviceの手動作成](#)

インストールの概要

[OraMTS](#)は、Oracle Database Clientの一部としてインストールできます。OraMTSサービスは、インストールしたOracleホームに対して作成されます。Oracle 12c以降、各Oracleホームには独自のOraMTS RecoveryのWindowsサービスがあります。また、Oracle 12c以降、OraMTSサービスはOracleホーム・ユーザーとして実行されます。

Oracleホーム・ユーザーは、Oracleホームから実行されるOracleサービスの所有者で、インストール後は変更できません。Windows組込みアカウントまたはWindowsユーザー・アカウントを指定できます。セキュリティを強化するため、Oracle Databaseのインストールで、Windowsの組込みアカウントではなく標準のWindowsユーザー・アカウントをOracleホーム・ユーザーとして選択することをお勧めします。Oracleホーム・ユーザーの主な目的は、Windowsユーザー・アカウントを使用してWindowsサービスを実行することです。ただし、このユーザー・アカウント(OraMTSホーム・ユーザー)は、データベース管理に使用することのない権限の低いユーザー・アカウントである必要があります。これによって、Oracleホーム・ユーザーで実行するOracle Databaseサービスが持つ権限が、Oracle製品を実行するときに必要な最低限の権限のみになります。Windowsユーザー・アカウントには、ローカル・ユーザー、ドメイン・ユーザーまたは管理対象サービス・アカウントを指定できます。

ソフトウェアのみのインストール、クローン・サイクルまたはデータベースでのAddNode操作などの一部の場合に、OraMTSサービスを手動で作成する必要がある場合があります。

このトピックでは、OraMTS用にインストールが必要なOracle製品およびOracle以外の製品について説明します。追加のインストール要件は次のとおりです。[Oracle Data Provider for .NET\(ODP.NET\)](#)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。

関連項目:

- [Oracle MTS Recovery Serviceの手動作成](#)
- [『Oracle Databaseプラットフォーム・ガイドfor Microsoft Windows』](#)

Microsoft Transaction Serverのインストール要件

Microsoft Transaction ServerがインストールされているWindowsコンピュータの製品要件は次のとおりです。

Oracle製品

- [OraMTS](#)
- OraMTSのインストールで自動的に含まれるOracle Database Client
- 次のようなOracle Services for MTSを使用するデータ・アクセス・ドライバ
 - [Oracle Data Provider for .NET\(ODP.NET\)](#)
 - [Oracle Open Database Connectivity\(ODBC\)ドライバ](#)
 - [Oracle Provider for OLE DB](#)
 - [Oracle Call Interface\(OCI\)](#)
- Oracle Database 11gリリース2以上のバージョンへのアクセス

注意:



- Oracle ODBC ドライバ、ODP.NET、Oracle Provider for OLE DB および OCI 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にのみ必要です。
- インストールによっては、Oracle MTS Recovery Service がインダウト・トランザクションを解決するためにリクエストをリスニングするポートの番号を入力するよう求められます。

Oracle以外の製品

- Windowsオペレーティング・システム

関連項目:

[『Oracle Databaseインストール・ガイド』](#)

- Microsoft Distributed Transaction Coordinator (Windowsオペレーティング・システムの一部)

分散トランザクション

分散トランザクションは1つ以上の文からなり、それらが個別に、またはグループとして、分散データベースの複数ノードのデータを更新します。

ODP.NET管理対象ドライバと管理対象外ドライバは、Microsoft Distributed Transaction Coordinator (MSDTC) およびOracleデータベースと統合されます。MSDTCは、同じSystem. Transactionsオブジェクトに入っているあらゆるリソース・マネージャと関係し、2フェーズ・コミットまたはロールバックをアトミックに実行します。これにより、Oracle分散トランザクションをネットワークで適切にコミットまたはロールバックできます。

ODP.NETでは、分散トランザクションにコミット読み取り分離レベルのみがサポートされています。



注意:

ODP.NET Core は分散トランザクションをサポートしていません。

Microsoft Distributed Transaction Coordinatorの統合

管理対象ODP.NETには、分散トランザクション登録、コミットおよびロールバックのためにMSDTCとの統合をサポートする、ネイティブで完全な管理対象実装が含まれます。

管理対象外ODP.NETの場合、Oracle Services for Microsoft Transaction (OraMTS)がMSDTCとの統合を提供し、クライアント・コンポーネントがOracleデータベース分散トランザクションに参加できるようになります。OraMTSは、MSDTCに対するOracleデータベースのプロキシとして動作し、Oracle分散データベース・トランザクションを残りの分散トランザクションとともにコミットまたはロールバックできるようにします。

管理対象外ODP.NETは、管理対象または管理対象外のOraMTS実装を使用できます。Oracle RACまたはData Guardからの高可用性を必要とするアプリケーションでは、管理対象OraMTSの使用をお勧めします。

分散トランザクションでネットワーク障害やサーバー・ハードウェア障害などの障害が発生した場合、インプロセス・トランザクションをインダウトにしておくことができます。管理対象および管理対象外のODP.NETはどちらも、OraMTSリカバリ・サービスを使用して、これらのインダウト・トランザクションを解決します。このリカバリ・サービスはWindowsサービスとして実行されます。管理者は、OraMTS Windowsリカバリ・サービスをインストールおよび構成して、管理対象と管理対象外のどちらのODP.NETを使用するかについてのリカバリ・シナリオを管理する必要があります。各マシンに必要なリカバリ・サービスは1つのみです。

ODP.NETが実行されMSDTCに参加しているすべてのクライアント・マシンに、OraMTSリカバリ・サービスをインストールする必要があります。マシンには複数のIPアドレスが存在する可能性があるため、管理対象ODP.NETアプリケーションの管理者は、実行中のリカバリ・サービスがあるホスト・マシン名またはIPアドレスをアプリケーションの.NET構成ファイルで指定できます。

ODP.NET, Unmanaged Driverは、リカバリ・サービスのIP/マシン名を自動的に解決します。

表2-1 分散トランザクションでサポートされるODP.NETのタイプと.NET Frameworkのバージョン

ODP.NETタイプ	.NET Frameworkバージョン	分散トランザクションのサポート
管理対象	4.5.2 以上	分散トランザクションに対する.NET Framework のネイティブ管理対象実装。
管理対象外	4.5.2 以上	OraMTS (デフォルト)または管理対象 OraMTS 実装。Oracle RAC または Data Guard による高可用性が必要な管理対象外 ODP.NET アプリケーションには、OraMTS を使用することが推奨されます。
管理対象外	3.5	OraMTS



注意:

.NET Framework 4 ファミリー内の.NET Framework 4.5.1 以下は、Microsoft および Oracle でサ

ポートされていません。

関連項目:

『[Oracle Services for Microsoft Transaction Server開発者ガイド for Microsoft Windows](#)』のOracle MTS Recovery Serviceの手動作成に関する項。

ODP.NET, Managed Driverの設定

この項では、ODP.NET, Managed Driverで分散トランザクションを使用するのに必要な設定および構成ステップを説明します。これらの.NET Frameworkバージョンで分散トランザクションを構成するには、次のステップに従います。

1. Oracle MTS Recovery Serviceを作成して設定するか、既存のリカバリ・サービスが実行中であることを確認します。
2. .NET構成でOMTSRECO_PORTの値を設定し、OraMTS Recovery Serviceが実行されているポート番号を指定します。

ODP.NET, Unmanaged Driverの設定

この項では、ODP.NET, Unmanaged Driverで分散トランザクションを使用するのに必要な設定および構成ステップを説明します。

.NET Framework 4.5.2以上の場合、ODP.NET管理対象外ドライバのアセンブリには管理対象OraMTSが含まれます。OraMTSは、アップグレード時に実装を継続できるよう、ODP.NET管理対象外ドライバのデフォルトのオプションです。ただし、Oracle Real Application ClustersまたはOracle Data Guardとともに高可用性FAN操作(HA Events = true)を使用する場合、管理対象OraMTSオプションを使用することをお勧めします。管理対象OraMTSオプションでは高可用性機能がサポートされますが、従来のOraMTSではサポートされません。

アプリケーションでは、.NET構成ファイルのUseOraMTSManagedパラメータを使用して、OraMTS (デフォルト)または管理対象OraMTSの使用を設定できます。

ODP.NET管理対象外ドライバに従来のOraMTS実装を使用するリカバリ・サービスを含めて、OraMTSをインストールして構成します。

管理対象OraMTSを設定するには、次のステップを実行します。

1. .NET構成ファイルで、UseOraMTSManagedをtrueに設定します。
2. Oracle MTS Recovery Serviceを作成して設定するか、既存のリカバリ・サービスが実行中であることを確認します。

関連項目:

.NET構成の設定の詳細は、[\[distributedTransactionセクション\]](#)を参照してください

Oracle MTS Recovery Serviceの手動作成

Oracle Database 12c以降、Windowsソフトウェア上のOracle Databaseには、OraMTS Recovery Serviceの手動による作成を有効にするためのoramtsctl.exeユーティリティが含まれています。ODP.NET, Managed DriverとUnmanaged Driverは両方とも、OraMTSが不要な完全に管理されたMS DTCソリューションを使用することを選択できます。ただし、いずれの場合にも、Oracle MTS Recovery Serviceは設定する必要があります。MS DTCを使用するOracleクライアント・マシンごとに、1つのOracle MTS Recovery Serviceを構成する必要があります。その1つのサービスを使用して、同じマシン上の複数のOracleアプリケーションを共有できます。実行時に、現在のOracleホームにサービスが作成され、Oracleホーム・ユーザーとして実行されます。

Oracle MTS Recovery Serviceを手動で作成する手順は次のとおりです。

1. Oracle Databaseをインストールします。
2. コマンド・プロンプトを開きます。
3. 次のコマンドを実行します。

```
oramtsctl -new
```

Oracle MTS Recovery Serviceを削除するには、次を実行します。

```
oramtsctl -new
```

```
oramtsctl -delete
```

使用方法

```
oramtsctl <-new|-delete|-start|-stop|-status|-config|-passwd|-trace|-trcdir>  
  [-host <hostname|IP_address> ... ]  
  [-port <number>]  
  [-ip <IP_address>]  
  [-dtchost <DTC_hostname>]  
  [-cluster <on|off>]  
  [-default]  
  
-new
```

新しいOracle1MTSRecoveryServiceサービスを作成して開始します。

```
-delete [Oracle_home_directory]
```

現在のホームまたは指定されたホームのOracle1MTSRecoveryServiceを削除します。

```
-start
```

Oracle1MTSRecoveryServiceサービスを開始します。

```
-stop
```

Oracle1MTSRecoveryServiceサービスを停止します。

```
-status Query Oracler1MTSRecoveryService
```

サービス・ステータス。

```
-config [-port <number>] [-ip <IP_address>] [-dtchost <DTC_hostname>]
        [-cluster <on|off>] [-default]
```

Oracle1MTSRecoveryServiceサービスの構成を表示または設定します。

-defaultを使用して、Oracle1MTSRecoveryServiceの構成をリセットします。

```
-passwd
```

Oracle1MTSRecoveryServiceサービスのユーザー・パスワードを更新します。

```
-trace [0-5]
```

トレースを表示またはトレース・レベルを設定します。トレースを無効にするにはレベル0を設定します。

```
-trcdir [trace_directory]
```

トレース・ディレクトリを表示または設定します。

```
-host <hostname|IP_address> ...
```

名前またはIPアドレスで特定されるホストで操作を実行します。

オプションが指定されていない場合は、ローカル・ホストが使用されます。

```
-port <number>
```

サービス・ポートを指定するには、このオプションで-newまたは-configを実行します。

オプションが指定されない場合、サービス・ポートは自動的に構成されます。

```
-ip <IP_address>
```

IPアドレスを指定するには、このオプションで-newまたは-configを実行します。

```
-default
```

デフォルト構成を設定するには、このオプションで-newまたは-configを実行します。

```
oramtsctl -new -host host1 host2.domain.com host3 -port 2033
```

例:

自動構成でサービスをインストールします。

```
oramtsctl -new
```

サービスをポート2032にインストールします。

```
oramtsctl -new -port 2032
```

ポート2033を使用するようにサービスを構成します。

```
oramtsctl -config -port 2033
```

サービスのユーザー・パスワードを更新します。

```
oramtsctl -passwd
```

複数のホストにサービスをインストールし、統一ポート2033を使用します。

```
oramtsctl -new -host host1 host2.domain.com host3 -port 2033
```

3 リカバリのシナリオ管理

これらのトピックでは、[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。

- [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の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\(OraMTS\)](#)のインストールまたは移行後に、Microsoft Transaction ServerおよびOracle Database環境を構成する必要があります。

コンピュータにMicrosoft Transaction Serverがインストールされている場合、Windowsコンピュータでは構成の必要はありません。

Microsoft Transaction Serverを構成するには、Oracle Databaseがインストールされているコンピュータで次のタスクを実行します。

1. データベースに対してoramtsadmin.sqlスクリプトを実行し、Microsoft Transaction Server管理者ユーザー・アカウント(デフォルト・ユーザー名はmtssys)を作成します。
2. 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。
[「Microsoft Transaction Serverの自動トランザクション・リカバリのスケジューリング」](#)を参照してください。
3. Oracle Fail Safe構成を持っている場合は、oramtsadmin.sqlスクリプトの実行前または実行後にレジストリの値を変更します。
[「Oracle Fail Safe構成用のレジストリ値の変更」](#)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 Database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のアプリケーション
- 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で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。

Oracle MTS Recovery Serviceにより、Microsoft Transaction Serverのインダウト・トランザクションが次の順序で解決されます。

1. DBMSリカバリ・ジョブにより、MTS連携のインダウト・トランザクションが検出されます。
2. DBMSリカバリ・ジョブでは、インダウト・トランザクションのXIDから、リカバリ・サービスのエンドポイントのアドレスが抽出され、MTS/MS DTCトランザクションの結果に対するリカバリ・サービスが要求されます。
3. リカバリ・サービスにより、トランザクションの結果に対するMS DTCが要求されます。
4. リカバリ・サービスでは、DBMSジョブ・プロセスに対してトランザクションの結果が報告されます。
5. DBMSリカバリ・ジョブにより、インダウト・トランザクションがコミットまたは終了されます。

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

OraMTSは、サーバーベースのリカバリを使用して、MSDTCによって開始されたインダウト・トランザクションを解決します。これを行うには、OraMTS管理者は、UTL_HTTPを通じてWindows中間層ノードにアクセス可能である必要があります。[「自動トランザクション・リカバリの構成」](#)に示すように、oramtsadmin.sqlにより、UTL_HTTPに対する実行権限がOraMTS管理者に付与されます

注意:



Oracle バージョン 11g 以上では、[「アクセス制御リスト\(ACL\)の作成」](#)で説明するとおり、DBA はアクセス制御リスト(ACL)を作成する必要があります。

アクセス制御リスト(ACL)の作成

Oracle Databaseバージョン11g 以上では、DBAは、アウトバウンドHTTP接続を作成する権限をOraMTS管理者に付与するアクセス制御リスト(ACL)を作成する必要があります。[例3-1](#)で、これを説明しています。

例3-1 ACLリストの作成とOraMTS管理者の追加

```
BEGIN
-- Create the new ACL, naming it "OraMTSadmin.xml", with a description.
-- This provides the OraMTS administrative user e.g. MTSADMIN user FOO
-- the privilege to connect
DBMS_NETWORK_ACL_ADMIN.CREATE_ACL('OraMTSadmin.xml',
                                   'Allow usage to the UTL network packages',
                                   'FOO', TRUE, 'connect');
-- Now grant privilege to resolve DNS names to the OraMTS administrative user
DBMS_NETWORK_ACL_ADMIN.ADD_PRIVILEGE('OraMTSadmin.xml',
                                       'FOO', TRUE, 'resolve');
-- Specify which hosts this ACL applies to, in this case we are allowing
-- access to all hosts. if one knew the list of all Windows middle-tier,
-- these could be added one by one.
DBMS_NETWORK_ACL_ADMIN.ASSIGN_ACL('OraMTSadmin.xml', '*');
END;
```

自動トランザクション・リカバリの構成

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・リカバリは、utl_oramts.sqlおよびprvtoramts.plbスクリプトをトリガーしてPL/SQLパッケージutl_oramtsを作成するoramtsadmin.sqlスクリプトを実行することで構成されます。また、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表示するデータベース・ビューoramts_2pc_pendingが作成されます。

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
- 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、データベースでの自動トランザクション・リカバ리를スケジューリングします。

- [データベースのジョブ・キュー・プロセスの設定および起動](#)
- [自動トランザクション・リカバリの作成とスケジューリング](#)

データベースのジョブ・キュー・プロセスの設定および起動

初期化パラメータJOB_QUEUE_PROCESSESでは、1インスタンスで起動されるジョブ・スレーブの最大数を指定します。

ジョブ・キュー・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. Oracle Database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. 初期化パラメータJOB_QUEUE_PROCESSES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。

6. Oracle Databaseを停止します。

```
SQL> SHUTDOWN
```

7. Oracle Databaseを再起動します。

```
SQL> STARTUP
```


8. SQL*Plusを終了します。

```
SQL> EXIT
```

自動トランザクション・リカバリの作成とスケジューリング

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。

- CREATE SESSION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. Oracle Database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. oramtsadmin.sqlスクリプトを実行します。

```
SQL> @ORACLE_BASE¥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. ステップ5でパスワードを変更しなかった場合は、次のスクリプトを使用して、変更できます。

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

このタスクの完了後にユーザー名を変更するには、そのユーザーを削除してoramtsadmin.sqlスクリプトを再実行し、要求されたら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プロセス](#)
- [oramts_2pc_pendingビュー](#)

utl_oramts.show_indoubtプロセス

このプロセスを使用して、データベース内のMicrosoft Transaction Serverインダウト・トランザクションを表示します。このプロセスでは、dbms_outputパッケージを使用して結果を表示します。

説明

このプロセスは、SERVEROUTPUTがONに設定されている必要があります。

```
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          : G2229A505904974D81FB7316B147325900000000
bqual         : 5BAB6A6B55CD294AA20335839110829C0100000000901944700050
local txid     : 142.11.202
tx state       : prepared
protocol       : HTTP
endpoint       : middletier-1@foo.com:2030
=====
formatid       : 21255235
gtrid          : 259DF9C8DFC5574F8876F0DF4E15CCAD000000000
bqual         : 2C8DCE5B9816244BA2B73CC013EEB870100000000901944700050
local txid     : 2.18.185
tx state       : prepared
protocol       : HTTP
endpoint       : middletier-2@foo.com:2030
=====
```

utl_oramts.recover_automaticプロセス

このプロセスは、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プロセス

このプロセスを使用して、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。このプロセスは、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

oramts_2pc_pendingビューは、oramtsadmin.sqlを実行することで作成されます。oramts_2pc_pendingは、データベース内のインダウト・トランザクションを表示します。このビューは次の列で構成されています。

Formatid

これは、データベース内のグローバル・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

これには、Oracleトランザクションのブランチ識別子が示されます。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

これには、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終了などのトランザクションの状態が示されます。

protocol

これは、データベース内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oramts_2pc_pendingビューを問い合わせます。

1. Microsoft Transaction Server管理者ユーザー・アカウントを使用して、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。これにより、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るようになります。

Microsoft Transaction ServerアプリケーションがWindows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。

4 Microsoft Transaction ServerおよびOracle Databaseを使用したプログラミング

これらのトピックでは、[Microsoft Transaction Server](#)およびOracle Databaseを使用したプログラミング方法について説明します。

- [トランザクション内でのCOMコンポーネント統合](#)
- [Microsoft Transaction Serverアプリケーションの開発](#)
- [Microsoft Transaction ServerとのOCI統合](#)
- [ODBCとMicrosoft Transaction Serverの統合の概要](#)

OraMTSでは、Oracle Provider for OLE DBおよびOracle Data Provider for .NETとの統合も可能になります。

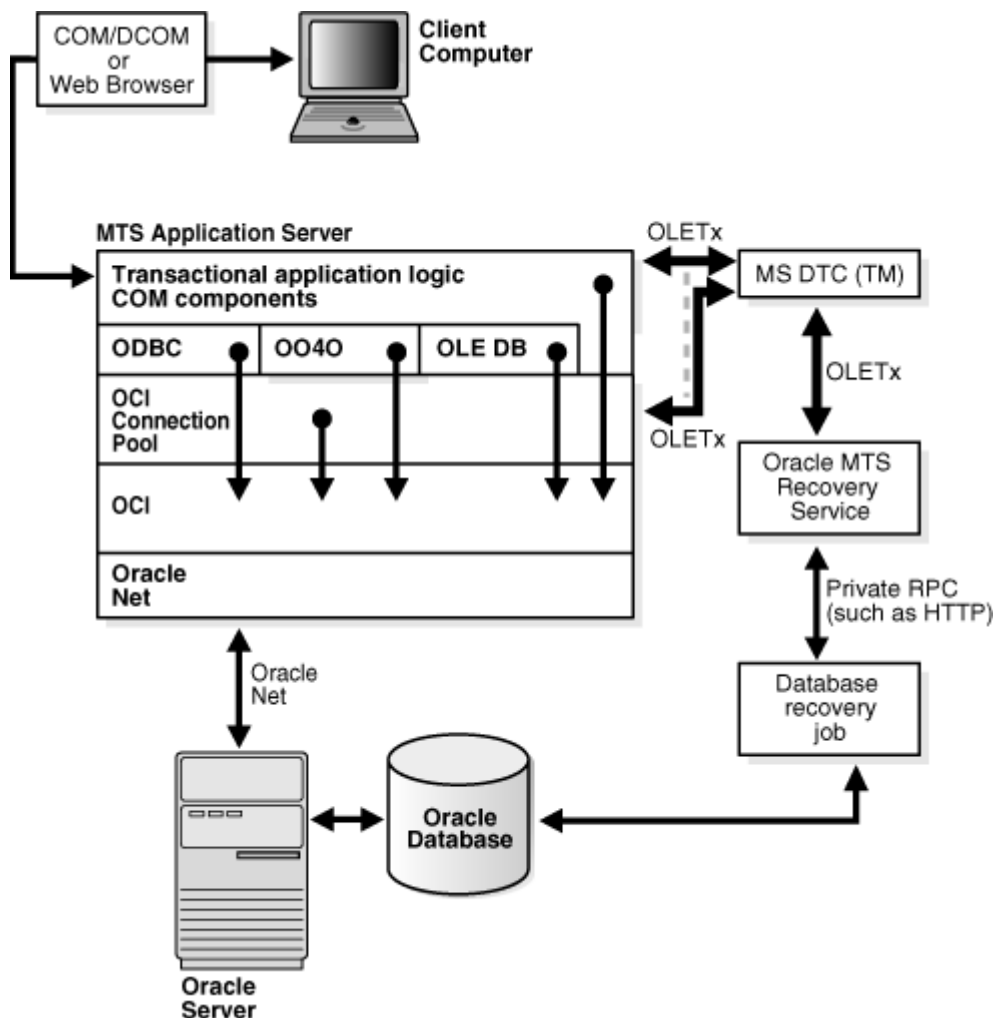
関連項目:

- MTSでのOracle Provider for OLE DBの使用方法的詳細は、『[Oracle Provider for OLE DB開発者ガイド for Microsoft Windows](#)』を参照してください。
- MTSでのOracle Data Provider for .NETの使用方法的詳細は、『[Oracle Data Provider for .NET開発者ガイド](#)』(Microsoft Windows用)を参照してください。

トランザクション内でのCOMコンポーネント統合

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4-1](#)はこのトランザクション・モデルを表しています。

図4-1 トランザクション内でのコンポーネント統合



クライアント・コンピュータ

クライアント・コンピュータは、Webブラウザや[Component Object Model \(COM\)/Distributed Component Object Model \(DCOM\)](#)を通じて[MTSアプリケーション・サーバー](#)上のアプリケーション・コンポーネントをアクティブにします。

MTSアプリケーション・サーバー

MTSアプリケーション・サーバーは、クライアント・コンピュータがWebブラウザを通じて間接的に、または[Component Object Model \(COM\)/Distributed Component Object Model \(DCOM\)](#)を通じて直接的にアクティブ化できるトランザクション・アプリケーション・コンポーネントをホストするためにWindowsオペレーティング・サービスによって提供されるサービスで構成されます。このアプリケーション・サーバーは、クライアントのリクエストに応じてCOMコンポーネントを起動します。起動は、必要に応じてトランザクションの範囲内で実行されます。

トランザクション・アプリケーション・ロジックCOMコンポーネント

主に次の3つの役割を果たします。

- 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たびにトランザクションが開始されます。
- Oracleリソース・ディスパッチと、[Oracle Call Interface \(OCI\)](#)、[Oracle Open Database Connectivity \(ODBC\) Driver](#)または[Oracle Provider for OLE DB](#)を通じて、Oracle Databaseへのプール済の接続を取得。
- 変更をすべてのRMにコミットするか終了するかを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、Oracle Provider for OLE DB、およびOCI

主に次の2つの役割を果たします。

- Oracle Database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。

OCI接続プール

主に次の3つの役割を果たします。

- RM(Oracle Database)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 Net

異機種分散コンピュータ環境で接続を提供します。

Oracle MTS Recovery Service

ホスト・コンピュータで開始され、Microsoft Transaction Serverと連携するインダウトOracleトランザクションをリカバリします。

データベース・リカバリ・ジョブ

インダウトDTCトランザクションを検出します。このジョブは、インダウト・トランザクションのXIDから、リカバリ・サービスのエンドポイントのアドレスを抽出し、リカバリ・サービスからMicrosoft DTCトランザクションの結果を要求します。最終的に、トランザクションの結果を受け取ると、このジョブはインダウト・トランザクションをコミットまたは終了します。

Microsoft DTC

Microsoft分散トランザクション・コーディネータは、Microsoft Transaction Serverの一部であり、次に示す2つの主要な役割があります。

- 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に送られます。

Oracleデータベース

Microsoft Transaction ServerのRMとして機能します。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

OCI接続プールは、ほとんどすべてのアプリケーション・プログラミング・インタフェースで、トランザクションの調整に利用されます。このトピックでは、トランザクションの登録方法、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説明します。

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

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、Dynamic Link Library(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。

登録コンポーネントのタイプ

アプリケーション・コンポーネントを登録するときには、次のいずれかのタイプを指定します。

- **トランザクションが必要:** コンポーネントはトランザクションで実行する必要があります。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たび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:** コンポーネントは、クライアントのトランザクション内で実行できます。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。
- **新しいトランザクションが必要:** コンポーネントは、専用のトランザクション内で実行する必要があります。Microsoft Transaction Serverにより、コンポーネントでメソッドが起動されるたび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しない:** コンポーネントはトランザクション内で実行されません。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。

コンポーネントの登録

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。

- アプリケーション・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で**実行**される場合、常にOCI接続プールが使用され、Microsoft Transaction ServerおよびそのMS DTCコンポーネントにより、トランザクションの作成、開始、管理およびコミットが調整されます。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、失敗した場合は終了されます。
- アプリケーション・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で**実行されない**場合、このコンポーネントはMicrosoft Transaction Server環境で実行されますが、コンポーネントがアクセスするデータベースは、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ります。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。データベースにアクセスするインターフェイス(Oracle Provider for OLE DB、Oracle ODBCドライバ、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。

Microsoft Transaction Serverにより調整されるコンポーネントのトランザクション

このトピックでは、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. クライアントAPI (Oracle ODBCドライバ、OCI、ODP.NETまたはOracle Provider for OLE DBのいずれか)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。

OCIサービスおよび環境ハンドルはクライアント・アプリケーションに戻されます。

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。

- データベース処理を実行します。
- 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。
- 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終了する場合)をコールします。

4. 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準備およびコミットするか、または終了します。これにより、OCI接続プール・コンポーネントに通知が送られ、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終了)の完了に必要なステップを実行します。

Microsoft DTCにより調整されるコンポーネントのトランザクション

このトピックでは、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、Oracle Databaseに接続します。接続プロトコルは、次のいずれかのシナリオのようになります。
 - プールされていないOCI接続は、`OCIserverAttach ()`や`OCISessionBegin ()`など、OCIログオン・コールを通じて取得されます。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。
 - 接続プールは、`OraMTSSvcGet (... , ORAMTS_CFLG_NOIMPLICIT)`をコールすることで取得されます。
2. クライアントは、次のいずれかのシナリオでコンテキストを処理します。
 - 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。
 - 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。
3. 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。
4. その後、クライアント・アプリケーションは次の処理を実行します。
 - データベース処理を実行します。
 - 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。
 - `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。

- MS DTCトランザクション・オブジェクトに対して、`pTransaction->Commit()` や `pTransaction->Abort()` のような、コミット・メソッドまたは終了メソッドをコールします。

5. 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。

6. 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終了)の完了に必要なステップを実行します。

Microsoft Transaction ServerとのOCI統合

[例4-1](#)は、OCIとMTSサーバーを統合する方法を表しています。コードを変更しなければならないのは、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。`OraMTSSvcGet()` をコールして、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の両方が取得されます。`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。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、`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・サポートが得られるようになります。

各プロセスで、他のOCIコールを実行する前に、`OCIInitialize`を1回以上コールするようにしてください。これにより、OCIプロセス環境が初期化されます。また、`OCI_THREADED`フラグをそれに渡す必要があります。Microsoft Internet Information Server(IIS)を使用している場合、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れば、`OCIInitialize`はすでにコールされています。レジストリ・キー`ORAMTS_OCI_OBJ_MODE`は追加済です。この値を1に設定し、OCIをオブジェクト・モードで初期化します(そうしないと、OCIはスレッド・モードで初期化されます)。

例4-1 MTSとOCIの統合

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

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

// MTS component method
HRESULT OCITestMethod()
{
    IObjectContext *pObjectContext = NULL;
    OCIEnv      *myenvh = NULL;
    OCISvcCtx  *mysvch = NULL;
    OCIError   *myerrh = NULL;
    OCIStnt   *mystmh = NULL;
    DWORD      dwStat;
    HRESULT    hRes = S_OK;
    sword      swOCIStat;
    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(
        "hr", "hr_password", "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 ...
swOCIStat = OCIHandleAlloc(myenvh, (void *)&myerrh, OCI_HTYPE_ERROR, 0, NULL);
if (Checkerr(swOCIStat, myerrh)) goto cleanup;
swOCIStat = OCIHandleAlloc(myenvh, (void *)&mystmh, OCI_HTYPE_STMT, 0, NULL);
if (Checkerr(swOCIStat, myerrh)) goto cleanup;
// prepare a DML statement
OCIStmtPrepare(mystmh, myerrh, lpzStmt, lstrlen(lpzStmt), OCI_NTV_SYNTAX,
OCI_DEFAULT)
Checkerr(swOCIStat, myerrh);
// execute the statement -- ensure that AUTOCOMMIT is not requested.
OCIStmtExecute(mysvch, mystmh, myerrh, 1, 0, NULL, NULL, OCI_DEFAULT);
if (Checkerr(swOCIStat, 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);
}

```

COMコンポーネントの統合

COMコンポーネントの統合には、いくつかのシナリオがあります。Microsoft Transaction Server環境でホストされていないCOMアプリケーション(スタンドアロン・アプリケーション)は、Microsoft Transaction ServerエクスプローラのMicrosoft管理コンソールを通じて宣言トランザクションを使用することはできませんが、次に示すシナリオのうち、最後の3つを使用できます。

MTS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

MTSによって調整されるトランザクションで実行されるCOMコンポーネントは、OCI接続プールを使用して、トランザクションにデータベースを暗黙的に登録します。次の擬似コード・リストは、OCI関数の使用方法を示しています。

```

OCIInitialize(OCI_THREADED, ...)
OraMTSSvcGet(..., &OCISvc, ..., ORAMTS_CFLAG_ALLDEFAULT)
...
OraMTSSvcRel (OCISvc)

```

OCI接続プールで実行される、トランザクション以外のCOMコンポーネント

トランザクションではないと指定され、かつMTSによって調整されるトランザクションで実行されるCOMコンポーネントは、OCI接続プールを使用しますが、トランザクションにデータベースを**登録しません**。次の擬似コード・リストは、OCI関数の使用方法を示しています。

```

OCIInitialize(OCI_THREADED, ...)
OraMTSSvcGet(..., &OCISvc, ..., ORAMTS_CFLAG_NOIMPLICIT)
...
OraMTSSvcRel (OCISvc)

```

MS DTCおよびOCI接続プールを使用するCOMコンポーネント

MTSによって調整されるトランザクションで実行されていないCOMコンポーネントは、OCI接続プールとともにMS DTCを使用して、トランザクションにデータベースを明示的に登録します。次の擬似コード・リストは、OCI関数の使用方法を示しています。

```
OCIInitialize(OCI_THREADED, ...)  
DTCGetTransactionManager(...)  
BeginTransaction(..., &transaction)  
OraMTSSvcGet(..., &OCISvc, ..., ORAMTS_CFLAG_NOIMPLICIT)  
OraMTSSvcEnlist(OCISvc, ..., transaction, ...)  
...  
OraMTSSvcEnlist(OCISvc, ..., NULL, ...)  
OraMTSSvcRel(OCISvc)
```

MS DTCおよびプールされていないOCI接続を使用するCOMコンポーネント

MTSによって調整されるトランザクションで実行されていないCOMコンポーネントは、プールされていないOCI接続とともにMS DTCを使用して、トランザクションにデータベースを明示的に登録します。次の擬似コード・リストは、OCI関数の使用方法を示しています。

```
OCIInitialize(OCI_THREADED, ...)  
OCI to get connected  
OraMTSEnlCtxGET  
DTCGetTransactionManager(...)  
BeginTransaction(..., &transaction)  
OraMTSJoinTxn(OCISvc, ..., transaction, ...)  
...  
OraMTSJoinTxn  
...  
OraMTSEnlCtxRel()  
OCI to logoff
```

OCI関数の使用方法

このトピックでは、前述のOCI関数について詳しく説明します。[表4-1](#)ではこれらの関数をまとめています。

表4-1 MTSおよびOracle Databaseを統合するためのOCI関数のサマリー

OCI関数	概要
OraMTSSvcGet()	OCI 接続プールから、プール済接続を取得します。
OraMTSSvcRel()	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。
OraMTSSvcEnlist()	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。
OraMTSSvcEnlistEx()	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。
OraMTSEnlCtxGet()	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。
OraMTSEnlCtxRel()	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。

OCI関数	概要
OraMTSJoinTxn()	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。
OraMTSTransTest()	Microsoft Transaction Server で開始されたトランザクションの内部で実行されているかどうかをテストします。
OraMTSOCIErrGet()	OCI エラー・コードとメッセージ・テキストを取り出します。

OraMTSSvcGet()

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4-2 OraMTSSvcGet()のパラメータ

パラメータ	IN/OUT	説明
lpUname	IN	Oracle Database に接続するためのユーザー名。
lpPswd	IN	ユーザー名に対応するパスワード。
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)。
pOCISvc	OUT	OCI サービス・コンテキスト・ハンドルへのポインタ。
pOCIEnv	OUT	OCI 環境ハンドルへのポインタ。
dwConFlgs	IN	接続フラグ。使用可能な値は次のとおりです。

- 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に登録します。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_UNIQUESRVR

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 Oracle9i 以前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。

使用上の注意

- `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・トランザクションが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、Microsoft Transaction ServerトランザクションにOracle Databaseを登録せずに、接続プールを提供することもできます。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。
- 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hr/hr_password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 ("hr", "hr_password", "oracle", &OCISvc, &OCIEnv, ORAMTS_CFLG_ALLDEFAULT |
ORAMTS_CFLG_NOIMPLICIT);
```

- OraMTSSvcGet ()では、ユーザー名(lpUname)、パスワード(lpPsswd)およびネット・サービス名構文(lpDbname)と一緒にユーザー名引数に指定することはできません(hr/hr_password@prod_finなど)。かわりに、コール側は前述の構文例のように、lpUname、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()

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この関数を使用します。

構文

```
DWORD OraMTSSvcRel (OCISvcCtx* OCISvc);
```

パラメータ

表4-3 OraMTSSvcRel()のパラメータ

パラメータ	IN/OUT	説明
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。

使用上の注意

- 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()

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4-4 OraMTSSvcEnlist()のパラメータ

パラメータ	IN/OUT	説明
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。
OCIErr	IN/OUT	OCI エラー・ハンドル(無効)。
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。
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。

使用上の注意

- 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_GFLG_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。

- コール側は接続の割当て、接続の登録、処理の実行、接続の登録解除、接続の解放を行った後に、処理をコミットまたは終了する必要があります。

OraMTSSvcEnlistEx()

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4-5 OraMTSSvcEnlistEx()のパラメータ

パラメータ	IN/OUT	説明
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。
OCIErr	IN/OUT	OCI エラー・ハンドル(無効)。
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。
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。
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。

使用上の注意

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()

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*      lpPswd,
    text*      lpDbnam,
    OCIsvcCtx* pOCIsvc,
    OCIError*  pOCIErr,
    ub4        dwFlags,
    void**     pCtx);

```

パラメータ

表4-6 OraMTSEnlCtxGet()のパラメータ

パラメータ	IN/OUT	説明
lpUsername	IN	Oracle Database に接続するためのユーザー名。
lpPswd	IN	Oracle Database に接続するためのパスワード。
lpDbnam	IN	データベースに接続するためのネット・サービス名。
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。
pOCIErr	IN	OCI エラー・ハンドル
dwFlags	IN	登録フラグ。現在指定可能な値は 0 のみです。
pCtx	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。

使用上の注意

- 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。
 - OCIを通じてプールされていない接続を割り当てます。
 - OraMTSEnlCtxGet() をコールして、登録コンテキストを作成します。
 - OraMTSJoinTxn() をコールして、接続を確立します。
 - データベース処理を実行します。
 - 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。
 - 処理をコミットまたは終了します。
 - OraMTSEnlCtxRel() をコールして、登録コンテキストを解放します。
 - 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()

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* pCtx);
```

パラメータ

表4-7 OraMTSEnlCtxRel()のパラメータ

パラメータ	IN/OUT	説明
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。

使用上の注意

- 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()

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4-8 OraMTSJoinTxn()のパラメータ

パラメータ	IN	説明
pCtxt	IN	OCI 接続の登録コンテキスト
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。

使用上の注意

- 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()

Microsoft Transaction Serverで開始されたトランザクションの内部で実行されているかどうかをテストします。

構文

```
BOOL OraMTSTransTest();
```

戻り値

Microsoft Transaction Server transaction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。

使用上の注意

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()

最後に実行したOraMTS関数(通常はOraMTSSvcGet ()またはOraMTSJoinTxn ())のOCIE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 lpcEMsg,  
DWORD* lpdLen);
```

パラメータ

表4-9 OraMTSOCIErrGet()のパラメータ

パラメータ	IN/OUT	説明
dwErr	-	エラー・コード
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
lpdLen	-	メッセージの実際のバイト数に設定

戻り値

OCIEラーが発生すると、trueが戻されます。それ以外の場合はfalseが返されます。trueが戻された場合、lpcEMsgおよびlpdLenが有効で、スタッシュ・エラー・メッセージがあるときは、最大でlpdLenバイトがlpcEMsgにコピーされます。lpdLenにはメッセージの実際のバイト数が設定されます。

使用上の注意

[例4-2](#)は、OraMTSOCIErrGet ()が、このスレッドで最後に実行したOraMTSSvc ()のOCIEラー・コードおよびOCIEラー・メッセージ・テキストがある場合に、それを取り出す方法を示しています。

例4-2 OCIEラー・コードおよびメッセージ・テキストの取出し

```
DWORD dwStat = OraMTSSvcGet ("hr",  
"invalid_password",  
"fin_prod",  
"db",  
&mysvch,  
&myenvh,  
ORAMTS_CFLG_ALLDEFAULT);  
  
if (dwStat != ORAMTS_ERR_NOERROR)  
{  
DWORD dwOCIErr;
```

```
char    errBuf[MAX_PATH];
DWORD  errBufLen = sizeof(errBuf);

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

ODBCとMicrosoft Transaction Serverの統合の概要

このトピックでは、Oracle ODBCドライバをMicrosoft Transaction ServerおよびOracle Databaseとともに使用する方法について説明します。ODBCを正常に動作させるためにOCIコードを変更する必要はありません。

接続属性の設定

Oracle ODBC Driver 11.1またはMicrosoft Oracle ODBCドライバとともにMicrosoft Transaction Serverを使用するには、SQLSetConnectAttr関数を使用して接続属性を設定し、ODBCコードで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。

Oracle ODBC Driver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。SQLSetConnectAttr(..., SQL_ATTR_ENLIST_IN_DTC)コールを使用して、リリース11.1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。

リリースのOracle ODBCドライバ11.1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。

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、次のステップを実行します。それ以外の場合は、このステップはスキップします。

- 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。
- mtssamples.dsnを選択し、「削除」をクリックします。
- 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。

デモを使用する予定がない場合は、「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。

5. 「HOME_NAME」で「Oracle」を選択します。
6. 「詳細」をクリックします。
7. 「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=hr  
PASSWORD=hr_password
```

説明:

- SERVERは、デモがデータベースにアクセスするときに使用するデータベース別名(mtsdemo)です。
- USERNAMEは、このアプリケーションのデータベース・ユーザー名で、たとえば、hrが入ります。
- PASSWORDは、ユーザー名hrのデータベース・パスワードです。

hrスキーマにaccount表およびreceipt表が含まれていることを確認します。

8. 「OK」をクリックします。
9. 「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。
10. Microsoftサンプル・アプリケーションの場合は、mtssamples.dsn(MicrosoftのODBCの名前)と入力します。この名前は、ステップ4で削除した名前と完全に一致している必要があります。
開発中のアプリケーションの場合は、使用するDSNファイル名を入力します。
11. 「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。
12. 「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。
13. 「コントロール パネル」ウィンドウを閉じます。

関連項目:

詳細は、Microsoft Transaction ServerのSDKを参照してください。

Microsoft社製Oracle用ODBCドライバの使用

Oracle ODBCドライバのかわりに、Microsoft Oracle ODBCドライバを使用できます。Microsoftドライバを使用すると、Oracle Provider for OLE DBおよびOracle Data Provider for .NETとの統合はできませんので注意してください。また、Oracle ODBCドライバ、統合用のAPIサポート、またはOracleクライアントのサポート・サービスによるパフォーマンスの向上もありません。

Microsoft社製Oracle用ODBCドライバを有効にした後、さらに次のステップを実行して、Microsoft Oracle ODBCドライバを構成します。

Microsoft Oracle 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)およびSQL*Netリリース2.3以上をインストールします。
2. `ORACLE_BASE¥ORACLE_HOME¥oramts¥samples¥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。この別名は、mtssamples.dsnファイルで使用されます。
4. Oracle NetがインストールされているホームにRSF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。

関連項目:

Microsoft Oracle ODBCドライバを有効にする方法は、Microsoft Transaction Serverのオンライン・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。

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

これらのトピックでは、[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。

- [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](#)
- [トランザクション・タイムアウト・パラメータ値の増加](#)
- [初期化パラメータの設定の変更](#)
- [追加のパラメータ](#)
- [MSDTC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、アプリケーションのパフォーマンスが向上します。たとえば、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。その後、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でもORAMTSSvcGet ()を使用できますが、ORAMTS_CFLG_NOIMPLICITフラグも使用できます。複数のOracle Databaseインスタンス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 () の使用方法の詳細は、[\[Microsoft Transaction ServerとのOCI統合\]](#)を参照してください。

Microsoft Transaction Server接続の管理

.NETまたはCOMコンポーネントがOracle Databaseでのセッションを終了しても、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、他のコンポーネントがOracle Databaseに新たに接続するときに再利用されます。

接続プールのレジストリ・パラメータ

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上のファイルHKEY_LOCAL_MACHINE¥SOFTWARE¥ORACLE¥HOMEIDの中で変更できます。

ORAMTS_CONN_POOL_TIMEOUT

クライアント側の接続プール内で、接続がタイムアウトして解放されるまでに、アイドルで再利用可能な状態である時間を秒単位で表します。このパラメータのデフォルト値は120秒です。

ORAMTS_SESS_TXNTIMETOLIVE

oraMTSSvcGet ()を使用して確立した接続が、oraMTSSvcRel ()コールにより解放された後、クライアント側の接続プールで有効である時間を秒単位で表します。ORAMTS_CONN_POOL_TIMEOUTの値とORAMTS_NET_CACHE_TIMEOUTの値の合計は、接続が完全に終了するまでに実際にかかる時間を表します。このパラメータのデフォルト値は120秒です。

ORAMTS_NET_CACHE_TIMEOUT

Oracle Services for Microsoft Transaction Server内に実装されたリソース・ディスペンサが、Oracleデータベースにプール済の接続を確立します。これらの接続が使用されなくなると、ORAMTS_CONN_POOL_TIMEOUTで指定されたタイムアウト後、ユーザー・セッションは切断されます。ただし、基礎となるOracle Net接続は、このパラメータで指定された期間キャッシュされます。ミリ秒で示されたこの時間が経過した後、データベースへのキャッシュ済Oracle Net接続は終了します。このパラメータのデフォルト値は120000ミリ秒です。このパラメータには、ORAMTS_CONN_POOL_TIMEOUTの値よりも大きな値を設定することをお勧めします。ORAMTS_CONN_POOL_TIMEOUTとORAMTS_NET_CACHE_TIMEOUTに指定された時間の合計は、接続が完全に終了するまでに実際にかかる時間を表します。

ORAMTS_NET_CACHE_MAXFREE

所定の時間にクライアント側の接続プールで維持している必要がある最大空きサーバー接続数。このパラメータのデフォルト値は5です。

ORAMTS_OSCREDS_MATCH_LEVEL

init.oraファイルのOS_ROLES初期化パラメータがtrueの場合に実装されるWindowsセキュリティ・チェックのレベル。

ユーザーがCONNECTコマンドを使用して、Oracle Databaseへの接続を確立すると、Windowsのユーザー名は特定のデータベース・ロールと権限に関連付けられます。ユーザーが切断するとこの接続はプールされ、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に、このユーザーのWindowsユーザー名が1人目のユーザーと同一である場合、2人目のユーザーは1人目のユーザーと同じデータベース・ロールと権限を取得できます。2人目のユーザーがCREATE SESSIONおよびRESOURCEデータベース・ロールしか持っていないにもかかわらず、1人目のユーザーのDBA権限を取得してしまうと、セキュリティ上重要な問題となります。

デフォルトでは、ORAMTS_OSCREDS_MATCH_LEVELパラメータの値はOS_AUTH_LOGINで、ユーザー名とパスワードがNULLの場合のみ、Windowsセキュリティ・チェックが実行されます。

このパラメータの最も安全な設定はALWAYSです。これにより、Windowsセキュリティ・チェックが必ず実行され、NULLを除く同一のWindowsユーザー名が原因のセキュリティ違反がチェックされます。

Windowsのセキュリティ・チェックには多くのリソース・コストがかかるので、このパラメータの値をNEVERに設定することもできます。ただし、OS_ROLESがtrueであることがわかっている場合、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、このオプションを使用しないでください。

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。

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。

「コンポーネント サービス」ウィンドウが表示されます。

3. 「コンポーネント サービス」ウィンドウで、「コンソール ルート」をダブルクリックして、ツリー構造を展開します。
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とともに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。通常的环境下では、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。

初期化パラメータの設定の変更

Microsoft Transaction ServerでOracle Databaseを使用するには、初期化パラメータをいくつか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必要がある値は、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. Oracle Database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、SESSIONSおよびPROCESSパラメータの両方とも現在の値が適しています。 .NETまたはCOMベースのアプリケーションを作成してデプロイする場合、これらのパラメータの値は、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、Oracle Database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すぎないでください。

7. 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)

8. Oracle Databaseを停止します。

```
SQL> SHUTDOWN
```

9. Oracle Databaseを再起動します。

```
SQL> STARTUP
```

10. SQL*Plusを終了します。

```
SQL> EXIT
```

関連項目:

これらのパラメータについては、[『Oracle Databaseリファレンス』](#)を参照してください。

その他のパラメータ

レジストリ変数ORAMTS_ABORT_MODEを使用して、新しい接続が常に強制終了を実行するか、または事前に登録済の接続を使用して強制終了を実行するかを制御します。つまり、強制終了が同期か非同期かを制御します。

デフォルトでは、事前に登録された接続がトランザクションの強制終了を実行します(可能な場合)。

レジストリ変数: ORAMTS_ABORT_MODE

値:

- ORAMTS_ABORT_MODE_NEW_CONN_ONLY: 非同期の強制終了になります。データベースへの新しい接続は、トランザクションの強制終了を実行するために開かれています。
- その他の値は、デフォルトの動作を示します。

MSDTCの起動

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を起動するには、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で、「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。

「コンポーネント サービス」ウィンドウが表示されます。

2. 「コンポーネント サービス」ウィンドウで、「コンソール ルート」の下の「コンポーネント サービス」を展開します。
3. 「コンポーネント サービス」の下の「コンピュータ」を展開します。
4. 「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。

5. 「MS DTC開始」を選択します。

MS DTCが開始されます。

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

これらのトピックでは、Oracle Microsoft Transaction Serverのトラブルシューティングについて説明します。

- [Oracle Services for Microsoft Transaction Serverパフォーマンスの追跡](#)
- [接続プールに影響するOracle Netの変更の修正](#)
- [複数のデータベースを使用するアプリケーションの設計](#)
- [様々な種類の接続プールの使用](#)
- [インダウト・トランザクションの使用](#)
- [Microsoft Transaction Server管理者ユーザー・アカウントの削除](#)

Oracle Services for Microsoft Transaction Server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パフォーマンスに関する情報が記録されています。これには次の情報が含まれます。

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

レジストリ・パラメータは、次のタスクを実行するorants.dll内で追跡を処理します。

- Oracleデータベースを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接続があるクライアントを、[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Oracle Databaseを登録します。

MTSベースのCOMコンポーネントは、データベースの専用Oracleサーバーおよび共有Oracleサーバーの両方に対する接続を取得できます。次に、このコンポーネントでは、これらのデータベースの間にあらかじめ存在するデータベース・リンクを使用して、もう1つのデータベースで、データ操作言語を使用した分散更新が試行されます。共有サーバーからの分散更新には成功しますが、専用サーバーからの分散更新は失敗します。

追跡を処理するレジストリ・パラメータは、Oracle Services for Microsoft Transaction Serverのインストール中に`%%HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\HOME1D`に自動的に設定されます。

Oracle Services for Microsoft Transaction Serverを使用して、すべてのアプリケーションを再インストールします。また、OracleMTSRecoveryServiceのトレースに関連するパラメータを変更した場合は、Windowsサービスを再起動します。

[表6-1](#)は、ORAMTS_CP_TRACE_LEVELトレース値の範囲を示しています。

表6-1 ORAMTS_CP_TRACE_LEVELトレース・レジストリ・パラメータの値

レベル	説明
0	トレースを使用不可にします。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、トレースが無効になります。これはレベルを 0 に設定することと同じです。
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。
3	このレベルはサポートされていません。このパラメータを 3 に設定すると、レベル 2 のトレースが有効になります。
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。

注意:



[Oracle MTS Recovery Service](#) では、`ORACLE_BASE`、`ORACLE_HOME`、`oramts`、`trace` ディレクトリにもトレース・ファイル出力が生成されます。

接続プールに影響するOracle Netの変更の修正

OraMTSレイヤー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・サービス名の変更時に、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。

- アプリケーションがプロセス外のMicrosoft Transaction Serverコンポーネント(サーバー・パッケージ)である場合は、次のアプリケーションを実行します。

```
C:\> mtxstop
```

これにより、接続プールが空になります。

- アプリケーションがインプロセスのMicrosoft Transaction Serverコンポーネント(ライブラリ・パッケージ)である場合は、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。

複数のデータベースを使用するアプリケーションの設計

Oracleクライアントは、2種類の方法でデータベース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。

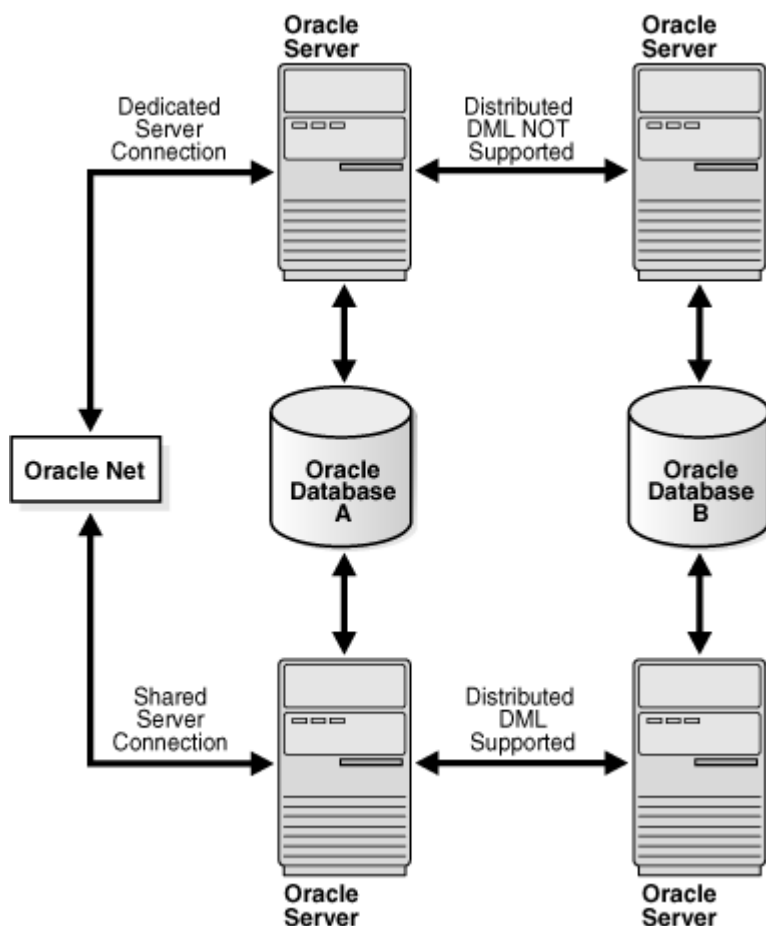
Microsoft Transaction Serverは、分散トランザクションを通じてデータベースと通信します。専用サーバー構成では、他のデータベースからの分散更新(データベース・リンクをまたがる[データ操作言語文](#))は使用できません。ただし、データベースへの元の接続が共有サーバー構成を使用して確立された場合は、他のデータベースからの分散更新が可能です。

共有サーバー構成でデータ操作言語文を使用するには、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。[図6-1](#)はこのプロセスを表しています。

図6-1 MTSアプリケーションからの分散DML文



様々な種類の接続プールの使用

Oracle Netの接続プーリング

Oracle Net接続プーリングはサーバー側の機能であり、Oracle Database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接続プーリング

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

OCI接続プーリング・レイヤーでは、MTS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。

インダウト・トランザクションの使用

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:

[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、`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を後から変更した場合、旧データベース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;
```

ここでの*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)。

参照:

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

用語集

原子性、一貫性、独立性および永続性(ACID)

ACIDは、トランザクション・マネージャ(トランザクション・マネージャとも呼びます)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ります。

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。

データ操作言語

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があります。

JOB_QUEUE_PROCESSES

この初期化パラメータは、1インスタンスで同時に実行されるDBMS_JOBジョブおよびOracle Scheduler(DBMS_SCHEDULER)ジョブの最大数を指定します。Oracle SchedulerジョブやDBMS_JOBジョブを実行したり、それらのジョブに依存するデータベース機能を使用するには、このパラメータを1以上に設定する必要があります。

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 .NET

Microsoft .NETは、Webサービスを介して、情報、人、システムおよびデバイスを相互に、あるいはインターネット上でより大きなアプリケーションに接続するために使用されるMicrosoft社のソフトウェア・テクノロジーのセット。

Microsoftアプリケーション・デモ

Windows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分散トランザクション・コーディネータ(MS DTC)

Microsoft分散トランザクション・コーディネータ(MS DTC)はMicrosoft Transaction Serverのコンポーネントで、トランザ

クシオン処理の中心となります。

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。

mtssys

デフォルトのMicrosoft Transaction Server管理者ユーザー名。Oracle9i Databaseリリース1(9.0.1)より前のリリースでは、これはOraMTSのユーザー名でした。

ネット・サービス名

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ます。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ます。

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も示します。

OraMTS

Oracle Services for Microsoft Transaction Serverの略。

Oracle Call Interface(OCI)

データベース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します。前処理やプリコンパイルを別途行う必要はありません。

Oracle Data Provider for .NET (ODP.NET)

Oracle Data Provider for .NET(ODP.NET)は、.NET環境からOracleデータベースへのデータ・アクセスを最適化します。ODP.NETでは、接続プール、PL/SQL、LOBs、RefCursors、グローバルゼーションまたはローカライゼーション、プロキシ・ユーザー認証またはパラメータ配列バインド、名前付きパラメータ、およびOracleタイプと.NETタイプ間の安全なタイプ・マッピングをサポートします。

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)します。

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します。Microsoft Transaction Server対応の各データベース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。

Oracle Open Database Connectivity (ODBC)ドライバ

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

提供します。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ありません。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ます。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です。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ます。

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よいアクセスを提供するインタフェース。

Oracle Services for Microsoft Transaction Server (OraMTS)

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デプロイすることが可能になります。

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。

リソース・マネージャ(RM)

Microsoft Transaction Serverは、データベースがリソース・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・プロセスに登録します。

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ます。

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権限も含まれます。

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ます。

トランザクション識別子(XIDs)

トランザクションが開始されたクライアント・コンピュータを示します。

索引

[A](#) [C](#) [D](#) [F](#) [G](#) [I](#) [J](#) [L](#) [M](#) [N](#) [O](#) [P](#) [R](#) [S](#) [T](#) [U](#) [V](#) [W](#)

A

- 管理者ユーザー名
 - 削除 [1](#)
 - Microsoft Transaction Server [1](#)
-

C

- Component Object Model(COM)
 - コンポーネントのトランザクション化 [1](#)
 - Oracle Call InterfaceおよびMicrosoft Transaction Serverを使用したプログラミング [1](#)
 - Microsoft Transaction Server環境への登録 [1](#)
 - Microsoft Transaction Server内での実行
 - 調整されたトランザクション [1](#)
 - MS DTCによって調整されるトランザクション内での実行 [1](#)
 - Microsoft社製Oracle用ODBCドライバでの使用 [1](#)
 - Oracle ODBCドライバおよびMicrosoft Transaction Serverとの使用 [1](#)
 - 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 [1](#)
 - 構成要件
 - クライアント・コンピュータ [1](#)
 - Microsoft Transaction Serverがインストールされているコンピュータ [1](#)
 - Oracle Databaseがインストールされているコンピュータ [1](#)
 - 接続
 - 接続プールの管理 [1](#)
 - 接続属性
 - ODBCでの設定 [1](#)
 - 接続プーリング
 - クライアント側のレジストリ・パラメータ [1](#)
 - 接続プールを空にする [1](#)
 - 接続の管理 [1](#)
 - サービス・ハンドルの取得 [1](#)
 - 接続の解放 [1](#)
 - OraMTSSvcGet()関数の使用方法 [1](#)
 - CREATE SESSIONロール[1](#)
-

D

- データベース・ジョブ・キュー・プロセス

- [起動 1](#)
 - データ操作言語 (DML)
 - [共有サーバー構成での使用 1](#)
 - [DBMS_JOBSパッケージ1](#)
 - [DBMS_TRANSACTIONパッケージ 1](#)
 - DCOM
 - 「Distributed Component Object Model (DCOM)」を参照
 - [OraMTSJoinTxn\(\)関数との違い 1](#)
 - [Distributed Component Object Model\(DCOM\) 1](#)
 - 分散トランザクション
 - [インダウト 1](#)
 - [RACサポート 1](#)
 - [シリアライズ可能分離レベルのサポート 1](#)
 - DML
 - 「データ操作言語(DML)」を参照
-

F

- [FORCE_ANY_TRANSACTION権限 1](#)
-

G

- はじめに
 - [Microsoft Transaction ServerおよびOracle Database 1](#)
-

I

- インダウト・トランザクション
 - [JOB_QUEUE_PROCESSES初期化パラメータ 1](#)
 - [解決 1](#)
 - [自動リカバリのスケジューリング 1](#)
 - [データベース・ジョブ・キュー・プロセスの起動 1](#)
 - [表示 1](#)
- 初期化パラメータ
 - [JOB_QUEUE_PROCESSES 1](#)
 - [PROCESSES 1](#)
 - [SESSIONS 1](#)
- インストール要件
 - [Microsoft Transaction Serverがインストールされているコンピュータ 1](#)
 - インストール要件
 - [Oracle Fail Safeが稼働中のコンピュータ 1](#)
 - [Microsoft Transaction Server 1](#)
 - [Oracle Database Client 1](#)

- Oracle ODBCドライバ [1](#)
 - Microsoftトランザクション・サーバー用のOracleサービス [1](#)
 - Service Pack 5.0以上 [1](#)
-

J

- JOB_QUEUE_PROCESSES初期化パラメータ [1](#)
-

L

- ローカル・トランザクション
 - 昇格 [1](#)
 - ローカル・ユーザー、ドメイン・ユーザー、管理対象サービス・アカウント [1](#)
-

M

- Microsoft分散トランザクション・コーディネータ(DTC) [1](#)
- 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 [1](#)
 - クラスタ内 [1](#)
 - 起動 [1](#)
- Microsoft Transaction Server
 - 利点 [1](#)
 - 管理者ユーザー名の変更 [1](#)
 - トランザクション内で実行されるCOMコンポーネント [1](#)
 - MS DTCによって調整されるトランザクションで実行されるコンポーネント [1](#)
 - 管理者ユーザー・アカウントの作成 [1](#)
 - 定義 [1](#)
 - Oracle Databaseに関する参照先 [1](#)
 - インストール要件 [1](#)
 - Oracle Databaseとの統合 [1](#)
 - Microsoft社製Oracle用ODBCドライバを使用したプログラミング [1](#)
 - Oracle Call Interfaceを使用したプログラミング [1](#)
 - Oracle ODBCドライバを使用したプログラミング [1](#)
 - COMコンポーネントの登録 [1](#)
 - トランザクション・リカバリのスケジューリング [1](#)
 - MS DTCの起動 [1](#)
 - Microsoft社製Oracle用ODBCドライバでの使用 [1](#)
 - Oracle ODBCドライバでの使用 [1](#)
- MTS Recovery Service [1](#)
- MTSSamples.dsnファイル
 - Oracle ODBCドライバでの使用 [1](#)
- mtssysユーザー名

- パスワードの変更 [1](#)
 - デフォルトの管理者ユーザー・アカウント [1](#)
 - mtxstop.exeファイル
 - 実行 [1](#)
-

N

- ネット・サービス名
 - 接続プールに影響する変更 [1](#)
 - 接続プールに影響する変更 [1](#)
 - プールされていないOracle Call Interface接続
 - OraMTSJoinTxn関数 [1](#)
-

O

- OCI_THREADEDフラグ
 - 受渡し [1](#)
- OCIInitialize関数
 - コール [1](#)
- ODBC
 - 「Open Database Connectivity (ODBC)」を参照
- omtssamp.sqlスクリプト [1](#)
- Open Database Connectivity
 - Microsoft Transaction ServerでのMicrosoft社製Oracle用ODBCドライバの構成 [1](#)
 - Microsoft Transaction ServerでのOracle ODBCドライバの構成 [1](#)
 - Oracle ODBCドライバのインストール要件 [1](#)
 - Microsoft Transaction Serverを使用したプログラミング [1](#)
 - 接続属性の設定 [1](#)
 - Microsoft Transaction ServerでのMicrosoft社製Oracle用ODBCドライバの使用 [1](#)
 - Oracle ODBCドライバでのMTSSamples.dsnファイルの使用 [1](#)
 - Microsoft Transaction ServerでのOracle ODBCドライバの使用 [1](#)
 - SQL_ATTR_ENLIST_IN_DTCパラメータの使用 [1](#)
 - SQLSetConnectAttr関数の使用 [1](#)
- Oracle Call Interface(OCI)
 - MS DTCによって調整されるトランザクションの登録 [1](#)
 - プール済のOracle Call Interface接続の取得 [1](#)
 - プール済または標準のOracle Call Interface接続の取得 [1](#)
 - OraMTSEnCtxGet()関数 [1](#)
 - OraMTSEnCtxGet()関数パラメータ [1](#)
 - OraMTSJoinTxn()関数 [1](#)
 - OraMTSOCIErrGet()関数 [1](#)
 - OraMTSOCIErrGet()関数パラメータ [1](#)
 - OraMTSSvcEnlist()関数 [1](#)
 - OraMTSSvcEnlist()関数パラメータ [1](#)

- OraMTSSvcGet()関数 [1](#)
- OraMTSSvcGet()関数パラメータ [1](#)
- OraMTSSvcRel()関数 [1](#)
- Microsoft Transaction Serverを使用したプログラミング [1](#)
- プール済のOracle Call Interface接続の解放 [1](#)
- Oracle Database
 - init.oraファイルのパラメータ設定の変更 [1](#)
 - Microsoft Transaction Serverとの統合 [1](#)
- Oracle Databaseクライアント
 - インストール要件 [1](#)
- Oracle Fail Safe [1](#), [2](#)
 - インストール要件 [1](#)
 - レジストリ・パラメータの変更 [1](#)
- Oracleホーム・ユーザー [1](#)
- Oracle MTS Recovery Service
 - インダウト・トランザクションの解決 [1](#)
 - トレース・ファイル出力 [1](#)
- Oracle Recovery MTS Service [1](#)
- Oracle Services for Microsoft Transaction [1](#)
- Microsoftトランザクション・サーバー用のOracleサービス
 - インストール要件 [1](#)
- oramts_2pc_pending
 - ビュー [1](#)
- ORAMTS_ABORT_MODE [1](#)
- ORAMTS_CFLG_SYSDBALOGNフラグ
 - 使用 [1](#)
- ORAMTS_CFLG_SYSOPRLOGNフラグ
 - 使用 [1](#)
- ORAMTS_CONN_POOL_TIMEOUTレジストリ・パラメータ [1](#)
- ORAMTS_NET_CACHE_MAXFREEレジストリ・パラメータ [1](#)
- ORAMTS_NET_CACHE_TIMEOUTレジストリ・パラメータ [1](#)
- ORAMTS_OSCREDS_MATCH_LEVELレジストリ・パラメータ [1](#)
- oramts.dllファイル
 - 定義 [1](#)
- oramtsadmin.sqlスクリプト
 - PL/SQLパッケージの作成 [1](#)
- OraMTSEnlCtxGet()関数
 - Oracle Call Interface関数 [1](#)
- OraMTSEnlCtxRel()関数
 - 以前に設定した登録コンテキストの破棄 [1](#)
 - パラメータ [1](#)
 - ORAMTSERR_NOERRORを返す [1](#)
 - 構文 [1](#)
- ORAMTSERR_ILLEGAL_OPER
 - 接続取得時の戻り値 [1](#)

- ORAMTSERR_NOERROR
 - 接続取得時の戻り値 [1](#), [2](#)
 - 接続取得時の戻り値 [1](#)
 - 接続解放時の戻り値 [1](#)
- OraMTSJoinTxn()関数
 - プールされていないOracle Call Interface接続の登録 [1](#)
 - Oracle Call Interface関数 [1](#)
 - 接続取得時にORAMTSERR_NOERRORを返す [1](#)
 - 構文 [1](#)
- OraMTSOCIErrGet()関数
 - パラメータ [1](#)
 - Oracle Call Interfaceエラー・コードの取得 [1](#)
 - 構文 [1](#)
- OraMTSSvcEnlist()関数
 - プール済または標準のOracle Call Interface接続の登録 [1](#)
 - Oracle Call Interface関数 [1](#)
 - パラメータ [1](#)
 - 使用上の制限事項 [1](#)
 - 接続取得時にORAMTSERR_NOERRORを返す [1](#)
 - 構文 [1](#)
- OraMTSSvcEnlistEx()関数
 - 使用上の制限事項 [1](#)
 - 接続取得時にORAMTSERR_ILLEGAL_OPERを返す [1](#)
 - 構文 [1](#)
- OraMTSSvcGet()関数
 - Oracle Call Interface関数 [1](#)
 - 概要 [1](#)
 - パラメータ [1](#)
 - 機能 [1](#)
 - プール済の接続を返す [1](#)
 - 接続取得時にORAMTSERR_NOERRORを返す [1](#)
 - 構文 [1](#)
- OraMTSSvcRel()関数
 - Oracle Call Interface関数 [1](#)
 - 概要 [1](#)
 - プール済の接続の解放 [1](#)
 - 接続解放時にORAMTSERR_NOERRORを返す [1](#)
 - 構文 [1](#)
- OraMTSTransTest()関数
 - 構文 [1](#)
- ORAOCIレジストリ・パラメータ
 - 設定 [1](#)

- パッケージ
 - DBMS_JOBS [1](#)
 - DBMS_TRANSACTION [1](#)
 - パスワード
 - mtssysユーザー名に対する変更 [1](#)
 - プールされた接続
 - 解放 [1](#)
 - 権限
 - FORCE_ANY_TRANSACTION [1](#)
 - 管理者ユーザー・アカウント [1](#)
 - utl_oramts.sqlスクリプト [1](#)
 - PROCESSES初期化パラメータ
 - 値の変更 [1](#)
 - 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 [1](#)
 - 昇格可能ローカル・トランザクション [1](#)
 - パブリック・プロシージャ
 - recover_automatic [1](#)
 - show_indoubt [1](#)
 - utl_oramts.forget_RMs [1](#)
-

R

- Real Application Clusters(RAC) [1](#)
 - recover_automatic
 - パブリック・プロシージャ [1](#)
 - リカバリ
 - インダウト・トランザクション [1](#)
 - レジストリ
 - Oracle Fail Safe構成用の値の変更 [1](#)
 - トレース・ファイルの設定 [1](#)
 - レジストリ・パラメータ
 - Oracle Fail Safe用の変更 [1](#)
 - ORAMTS_CONN_POOL_TIMEOUT [1](#)
 - ORAMTS_NET_CACHE_MAXFREE [1](#)
 - ORAMTS_NET_CACHE_TIMEOUT [1](#)
 - ORAMTS_OSCREDS_MATCH_LEVEL [1](#)
 - レジストリ変数
 - ORAMTS_ABORT_MODE [1](#)
 - ロール
 - CREATE SESSION [1](#)
 - 管理者ユーザー・アカウント [1](#)
 - SELECT_CATALOG_ROLE [1](#)
-

S

- [SELECT_CATALOG_ROLE](#)ロール [1](#)
 - シリアライズ可能トランザクション [1](#)
 - サービス・ハンドル [1](#)
 - Service Pack 5.0以上
 - インストール要件 [1](#)
 - SESSIONS初期化パラメータ
 - 値の変更 [1](#)
 - 共有サーバー構成 [1](#)
 - show_indoubt
 - パブリック・プロシージャ [1](#)
-

T

- 3層アーキテクチャ [1](#)
 - tnsnames.oraファイル
 - 共有サーバー構成用の設定 [1](#)
 - トレース・ファイル
 - ファイル名の規則 [1](#)
 - Oracle MTS Recovery Service [1](#)
 - oramts.dll [1](#)
 - レジストリの設定 [1](#)
 - 使用 [1](#)
 - トランザクション・リカバリ
 - JOB_QUEUE_PROCESSES初期化パラメータ [1](#)
 - Oracle Fail Safe環境 [1](#)
 - 概要 [1](#)
 - スケジュール [1](#)
 - データベース・ジョブ・キュー・プロセスの起動 [1](#)
 - トラブルシューティング [1](#)
 - トランザクション
 - データ・リソース間の一貫性の確保 [1](#)
 - 分散トランザクションの透過的なRACサポート [1](#)
 - トラブルシューティング
 - 接続プールに影響するOracle Netの変更の修正 [1](#)
 - 管理者ユーザー・アカウントの削除 [1](#)
 - MS DTCの起動 [1](#)
 - トランザクション・リカバリ [1](#)
 - トレース・ファイルの使用 [1](#)
 - チューニング
 - 変更 [1](#)
 - 接続プールの管理 [1](#)
 - 2フェーズ・コミット・プロトコル [1](#)
-

U

- utl_oramts.forget_RMs
 - パブリック・プロシージャ [1](#)
-

V

- ビュー
 - oramts_2pc_pending [1](#)
-

W

- Windowsユーザー・アカウント [1](#)