

Oracle Call Interface for Windows

スタート・ガイド

リリース 9.2

2002 年 7 月

部品番号 : J06333-01

ORACLE®

Oracle Call Interface for Windows スタート・ガイド, リリース 9.2

部品番号 : J06333-01

原本名 : Oracle Call Interface Getting Started, Release 9.2 for Windows

原本部品番号 : A95497-01

原本協力者 : Eric Belden, Janis Greenberg, Herbert Kelly, Helen Slattery, Ravi Thammaiah,
and Bob Thome

Copyright © 1995, 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に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	iii
対象読者	iv
このマニュアルの構成	iv
関連資料	iv
表記規則	v
Oracle Call Interface の新機能	xi
Oracle9i リリース 1 (9.0.1) の Oracle Call Interface における新機能	xii
Oracle8i リリース 8.1.5 の Oracle Call Interface における新機能	xii
OCI リリース 7.x の関数	xii
Oracle9i リリース 2 (9.2) の使用できないコンポーネント	xii
1 Oracle Call Interface の概要	
Oracle Call Interface の概要	1-2
OCI パッケージの内容	1-2
Oracle のディレクトリ構造	1-2
サンプル・プログラム	1-3
2 OCI アプリケーションの作成	
OCI アプリケーションの作成	2-2
OCI アプリケーションのコンパイル	2-2
OCI アプリケーションのリンク	2-3
oci.lib	2-3
LoadLibrary() を使用したクライアント DLL のロード	2-4

OCI アプリケーションの実行	2-4
Oracle XA ライブラリ	2-4
Oracle XA ライブラリを使用した OCI プログラムのコンパイルおよびリンク	2-5
XA 動的登録の使用方法	2-5
現行のセッションに対する環境変数の追加	2-6
すべてのセッションに対するレジストリ変数の追加	2-6
XA および TP モニターの情報	2-7
Windows NT 固有の Oracle C++ Call Interface のメソッド	2-7
OTT と INTYPE File Assistant の使用方法	2-10

索引

はじめに

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 のオペレーティング・システム環境で使用される Oracle Call Interface (OCI) の機能のみです。

この章の項目は次のとおりです。

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

対象読者

このマニュアルは、1 つ以上の Oracle サーバーと対話するアプリケーションを C 言語で作成する開発者を対象としています。

このマニュアルを使用するには、次の知識が必要です。

- C プログラムのコンパイルおよびリンクの方法
- 使用している Microsoft Windows オペレーティング・システム

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 Call Interface の概要」

OCI で作業を始めるのに役立つ初歩的な情報を提供します。

第 2 章「OCI アプリケーションの作成」

OCI を使用した Oracle データベース・アプリケーションの作成方法の概要を示します。

関連資料

詳細は、次の Oracle のマニュアル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 リリース・ノート』
- 『Oracle9i Database for Windows 管理者ガイド』
- 『Oracle Enterprise Manager 管理者ガイド』
- 『Oracle9i Net Services 管理者ガイド』
- 『Oracle9i Real Application Clusters 概要』
- 『Oracle9i データベース新機能』
- 『Oracle9i データベース概要』
- 『Oracle9i データベース・リファレンス』
- 『Oracle9i データベース・エラー・メッセージ』
- 『Oracle Call Interface プログラマーズ・ガイド』

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。

規則	意味	例
小文字（固定幅） フォント	<p>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。</p> <p>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。</p>	<p>sqlplus を入力して、SQL*Plus を開きます。</p> <p>パスワードは、orapwd ファイルで指定されます。</p> <p>¥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。</p> <p>department_id、department_name および location_id 列は、hr.departments 表にあります。</p> <p>QUERY_REWRITE_ENABLED 初期化パラメータを true に設定します。</p> <p>oe ユーザーとして接続します。</p> <p>JRepUtil クラスは、これらのメソッドを実装します。</p>
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、プロセスホルダまたは変数を示します。	<p>parallel_clause を指定できます。</p> <p>Uold_release.SQL を実行します。</p> <p>old_release 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。</p>

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・コードで使用され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枝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]

規則	意味	例
...	<p>水平の省略記号は、次のいずれかを示します。</p> <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 	<pre>CREATE TABLE ...AS subquery; SELECT col1, col2, ..., coln FROM employees;</pre>
.	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。	<pre>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.</pre>
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。	<pre>acctbal NUMBER(11,2); acct CONSTANT NUMBER(4) := 3;</pre>
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。	<pre>CONNECT SYSTEM/system_password DB_NAME = database_name</pre>
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。	<pre>SELECT last_name, employee_id FROM employees; SELECT * FROM USER_TABLES; DROP TABLE hr.employees;</pre>
小文字	<p>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。</p> <p>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。</p>	<pre>SELECT last_name, employee_id FROM employees; sqlplus hr/hr CREATE USER mjones IDENTIFIED BY ty3MU9;</pre>

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、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」→「Database Administration」→「Database Configuration Assistant」を選択します。	「スタート」→「プログラム」→「Oracle - HOME_NAME」→「Database Administration」→「Database Configuration Assistant」を選択します。
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。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) 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。コンピュータにリリース 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¥ora90 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 Call Interface の新機能

Oracle9i リリース 2 (9.2) のこの製品には、新機能は追加されていません。多数の新機能が、Oracle9i リリース 1 (9.0.1) で追加されています。これより前のリリースからアップグレードする場合、「[Oracle9i リリース 1 \(9.0.1\) の Oracle Call Interface における新機能](#)」を参照してください。

次の項では、Oracle Call Interface の新機能を説明します。

- [Oracle9i リリース 1 \(9.0.1\) の Oracle Call Interface における新機能](#)
- [Oracle8i リリース 8.1.5 の Oracle Call Interface における新機能](#)
- [OCI リリース 7.x の関数](#)
- [Oracle9i リリース 2 \(9.2\) の使用できないコンポーネント](#)

Oracle9i リリース 1 (9.0.1) の Oracle Call Interface における新機能

この項の項目は、次のとおりです。

- **Borland C++ のサポート**

oci.lib インポート・ライブラリは、Microsoft 社製のコンパイラとともに使用する目的でのみ提供されています。Borland C++ などのその他のコンパイラについては、Oracle Dynamic Link Library (DLL) に対応している場合でも、OCI との使用に関するテストおよびサポートは行われていません。

- **Windows 2000 での Oracle9i の使用方法**

Windows 2000 と Windows NT 4.0 では、Oracle9i の使用方法に若干の相違があります。

関連資料：『Oracle9i Database for Windows スタート・ガイド』

Oracle8i リリース 8.1.5 の Oracle Call Interface における新機能

OCI では、Oracle8i データベースでオブジェクトを処理する OCI の機能が拡張され、多くの新機能が組み込まれるとともにパフォーマンスが向上しています。オブジェクト機能を使用するには、Oracle8i Enterprise Edition をインストールしておく必要があります。

Windows プラットフォームの OCI には、OCI の以前のリリース (7.x および 8.x) で作成されたアプリケーションのサポートも組み込まれています。ライブラリ名 oci.lib からバージョン番号が削除されました。

OCI リリース 7.x の関数

リリース 7.x で使用可能な OCI 関数は使用できますが、Oracle8i の新機能を十分に活用できません。パフォーマンスの向上と機能の拡張を図るために、アプリケーションで新しいコールを使用することをお勧めします。

Windows 95、Windows 98 または Windows NT で実行される Win32 アプリケーションの場合、継続してサポートを受けるには、リリース 8.x の新しい OCI コールに移行する必要があります。リリース 8.x では、OCI コールの含まれるライブラリと DLL の名前はそれぞれ oci.lib と oci.dll です。リリース 7.x では、ociw32.lib と ociw32.dll という名前でした。将来、ociw32.lib と ociw32.dll はサポートやリリースがなくなるため、新しいコールへの移行は避けられません。

Oracle9i リリース 2 (9.2) の使用できないコンポーネント

INTYPE File Assistant (IFA) は、今回のリリースでは使用できません。また、次期リリースでもサポートされません。

Oracle Call Interface の概要

この章では、Oracle Call Interface (OCI) for Windows で作業を始めるための初歩的な情報を提供します。

この章の項目は次のとおりです。

- [Oracle Call Interface の概要](#)
- [OCI パッケージの内容](#)
- [Oracle のディレクトリ構造](#)
- [サンプル・プログラム](#)

関連資料： 新機能および関数の説明を含めた OCI の詳細は、『Oracle Call Interface プログラマーズ・ガイド』を参照してください。

Oracle Call Interface の概要

Oracle Call Interface（OCI）は、C 言語で作成されたアプリケーションが 1 つ以上の Oracle サーバーと対話できるようにするための Application Program Interface（API）です。OCI を使用すると、ユーザー・プログラムは、SQL 文の処理やオブジェクトの操作など、Oracle9i データベースで実行可能なすべてのデータベース操作を実行できるようになります。

OCI パッケージの内容

Oracle Call Interface for Windows のパッケージには、次のものが含まれています。

- Oracle Call Interface
- Required Support Files（RSF）
- Oracle Universal Installer
- OCI アプリケーションをコンパイルするためのヘッダー・ファイル
- OCI アプリケーションをリンクするためのライブラリ・ファイル
- OCI アプリケーションの作成方法を示すサンプル・プログラム

OCI for Windows パッケージには、Windows NT、Windows 2000 および Windows 98 で OCI プログラムをリンクする際に必要なその他のライブラリも組み込まれています。

Oracle のディレクトリ構造

Oracle Call Interface for Windows のインストール時に、Oracle Universal Installer によって、使用しているコンピュータのハード・ディスク・ドライブに ORACLE_HOME ディレクトリが作成されます。デフォルトの Oracle ホーム・ディレクトリは、C:¥oracle¥ora92 です。

ORACLE_HOME ディレクトリには、OCI ファイル以外にも、OCI アプリケーションのリンクおよび実行や、Oracle Forms などその他の Windows NT 用の Oracle 製品とのリンクに必要なライブラリ・ファイルがあります。

ORACLE_HOME ディレクトリには、OCI に関連した次のディレクトリがあります。

表 1-1 ORACLE_HOME ディレクトリおよび内容

ディレクトリ名	内容
¥bin	実行可能ファイルおよびヘルプ・ファイル
¥oci	Windows ファイル用の Oracle Call Interface ディレクトリ
¥oci¥include	oci.h および ociap.h などのヘッダー・ファイル

表 1-1 ORACLE_HOME ディレクトリおよび内容（続き）

ディレクトリ名	内容
%oci%samples	サンプル・プログラム
%precomp%admin%ottcfg.cfg	Object Type Translator（OTT）ユーティリティおよびデフォルトの構成ファイル

サンプル・プログラム

OCI のインストール時に、一連のサンプル・プログラムと関連するプロジェクト・ファイルが %ORACLE_HOME%\oci\samples サブディレクトリにコピーされます。OCI が正しくインストールされたことを検証し、OCI アプリケーションの開発手順を理解するために、これらのサンプル・プログラムをビルドし、実行することをお勧めします。

サンプルをビルドするには、コマンド・プロンプトでバッチ・ファイル（make.bat）を実行します。たとえば、cdemo1.c サンプルをビルドするには、次のコマンドを入力します。

```
C:> make cdemo1
```

サンプル・プログラムは、使用後は削除しても構いません。

Windows プラットフォーム固有の OCI サンプル・アプリケーションが 1 つ含まれています。cdemomt.c は、Oracle9i のスレッド・セーフティ機能である OCI マルチスレッドのデモ・プログラムです。このサンプル・プログラムには、デフォルト・データベース内の emp 表が必要です。このプログラムからは、同じ ID 番号を持つ異なる従業員名を挿入しようとする 2 つの同時スレッドが起動されます。これは、スレッド同期のデモを示します。

ociucb.c は、ociucb.bat を使用してコンパイルする必要があります。このバッチ・ファイルによって DLL が作成され、%ORACLE_HOME%\bin ディレクトリに置かれます。ユーザー・コールバック関数をロードするには、環境変数またはレジストリ変数 ORA_OCI_UCBPKG = OCIUCB を設定します。

関連資料： マルチスレッドの詳細は、『Oracle Call Interface プログラマーズ・ガイド』を参照してください。

OCI アプリケーションの作成

この章では、OCI を使用した Oracle データベース・アプリケーションの作成方法の概要を説明します。

この章の項目は次のとおりです。

- [OCI アプリケーションの作成](#)
- [OCI アプリケーションのコンパイル](#)
- [OCI アプリケーションのリンク](#)
- [Oracle XA ライブラリ](#)
- [OTT と INTYPE File Assistant の使用方法](#)

関連資料： OCI アプリケーションの作成の詳細は、『Oracle Call Interface プログラマーズ・ガイド』を参照してください。

OCI アプリケーションの作成

OCI アプリケーションは、一般的には Oracle サーバーに接続し、ある種のデータ交換を行って、必要なデータ処理を実行する目的で使用されます。順序は多少異なりますが、どのような OCI アプリケーションにも必要な処理がいくつかあります。

OCI で使用される基本的なプログラミング手順は次のとおりです。

1. OCI プログラミング環境およびプロセスを初期化します。
2. 必要なハンドルを割り当て、サーバー接続とユーザー・セッションを確立します。
3. サーバーに SQL 文を発行し、必要なアプリケーション・データ処理を実行します。
4. 再使用しない文およびハンドルを解放するか、作成した文を再実行するか、または新しい文を作成します。
5. ユーザー・セッションとサーバー接続を終了します。

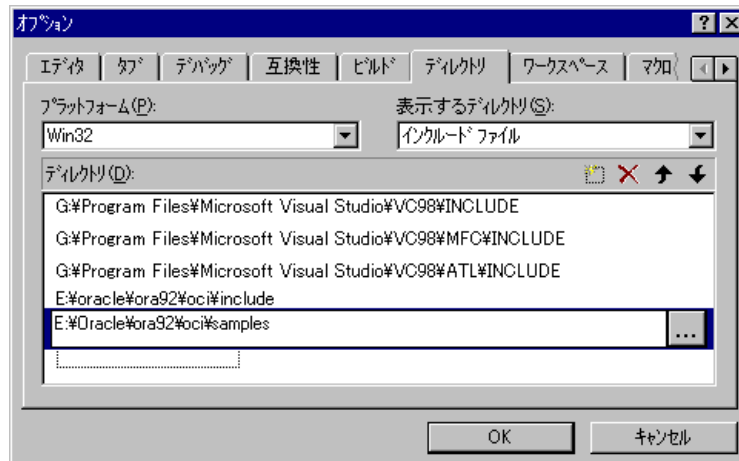
注意：『Oracle Call Interface プログラマーズ・ガイド』で説明されている共有データ・モードでの OCI 環境の初期化は、Windows ではサポートされません。

OCI アプリケーションのコンパイル

OCI アプリケーションをコンパイルするときは、適切な OCI ヘッダー・ファイルをインクルードする必要があります。ヘッダー・ファイルは、`%ORACLE_HOME%\oci\include` ディレクトリにあります。

たとえば、Microsoft Visual C++ 6.0 を使用する場合、「ツール」メニューの「オプション」ダイアログ・ボックスの「ディレクトリ」タブで適切なパスを入力する必要があります。
[図 2-1 「「オプション」ダイアログ・ボックスの「ディレクトリ」タブ](#)を参照してください。

図 2-1 「オプション」 ダイアログ・ボックスの「ディレクトリ」タブ



関連資料： アプリケーションのコンパイルとコンパイル・オプションの情報は、コンパイラのドキュメントを参照してください。

OCI アプリケーションのリンク

OCI コールは、Oracle 提供の DLL に実装されています。DLL は、Required Support Files (RSF) の一部で、%ORACLE_HOME%\bin ディレクトリにあります。

Oracle DLL を使用して OCI をコールするには、動的に DLL および関数エントリ・ポイントをロードするか、アプリケーションをインポート・ライブラリ `oci.lib` とリンクします。`oci.lib` インポート・ライブラリは、Microsoft 社製のコンパイラとともに使用する目的でのみ提供されています。その他のコンパイラについては、Oracle DLL に対応している場合でも、OCI との使用に関するテストおよびサポートは行われていません。

`oci.lib` を Microsoft 社製のコンパイラとともに使用する場合、特別なリンク・オプションを指定する必要はありません。

oci.lib

`oci.lib` は、Oracle に対する単一プログラム・インタフェースです。ライブラリ名からバージョン番号が削除されました。

LoadLibrary() を使用したクライアント DLL のロード

LoadLibrary により、次のディレクトリが記載されている順序で検索されます。

- ロードされるアプリケーションのあるディレクトリ
- カレント・ディレクトリ
- Windows NT または Windows 2000 の場合：
 - 32 ビット Windows システム・ディレクトリ (system32)。このディレクトリのパスを取得するには、GetWindowsDirectory 関数を使用します。
 - 16 ビット Windows ディレクトリ (system)。このディレクトリのパスを取得する Win32 関数はありませんが、検索が行われます。
- Windows 98 の場合：
 - Windows ディレクトリ。このディレクトリのパスを取得するには、GetWindowsDirectory 関数を使用します。
- 環境変数 PATH にリストされているディレクトリ

OCI アプリケーションの実行

OCI アプリケーションを実行するには、すべての対応する Required Support Files (RSF) が、OCI アプリケーションを実行するコンピュータにインストールされていることを確認してください。

Oracle XA ライブラリ

Oracle9i データベースが次のトランザクション処理 (TP) モニターと対話できるようにするには通常、XA API を使用します。

- BEA Tuxedo
- IBM Transarc Encina
- IBM CICS

クライアント・プログラムで TP モニターの文を使用することもできます。XA API の使用は、OCI からサポートされます。

Oracle XA ライブラリは、Oracle9i Enterprise Edition の一部として自動的にインストールされます。Oracle ホーム・ディレクトリに、次のコンポーネントが作成されます。

表 2-1 Oracle XA ライブラリ・コンポーネント

コンポーネント	場所
oraxa9.lib	%ORACLE_HOME%\rdbms\x
xa.h	%ORACLE_HOME%\rdbms\demo

Oracle XA ライブラリを使用した OCI プログラムのコンパイルおよびリンク

OCI プログラムをコンパイルおよびリンクする手順：

1. %ORACLE_HOME%\rdbms\x がパスに含まれていることを確認し、Microsoft Visual C++ を使用して program.c をコンパイルします。
2. program.obj と次のライブラリをリンクします。

ライブラリ	場所
oraxa9.lib	%ORACLE_HOME%\rdbms\x
oci.lib	%ORACLE_HOME%\oci\lib\msvc

3. program.exe を実行します。

XA 動的登録の使用方法

Oracle9i データベースは、XA 動的登録の使用をサポートしています。XA 動的登録により、XA 準拠 TP モニターとのインタフェースであるアプリケーションのパフォーマンスが向上します。Windows NT の Oracle データベースで TP モニターが XA 動的登録を使用するには、環境変数またはレジストリ変数のどちらかを、TP モニターが実行されている Windows NT コンピュータに追加する必要があります。手順は、次のいずれかを参照してください。

- 現行のセッションに対する環境変数の追加
- すべてのセッションに対するレジストリ変数の追加

現行のセッションに対する環境変数の追加

コマンド・プロンプトで環境変数を追加すると、現行のセッションのみが適用対象となります。

環境変数を追加する手順：

TP モニターがインストールされているコンピュータから、コマンド・プロンプトで次のように入力します。

```
C:¥> set ORA_XA_REG_DLL = vendor.dll
```

`vendor.dll` は、ベンダーから提供された TP モニターの DLL です。

すべてのセッションに対するレジストリ変数の追加

レジストリ変数を追加すると、使用している Windows NT コンピュータのすべてのセッションが適用対象となります。TP モニターが 1 つだけ実行中のコンピュータでは、この方法が便利です。

レジストリ変数を追加する手順：

1. TP モニターがインストールされているコンピュータに移動します。
2. Windows NT または Windows 2000 の場合、コマンド・プロンプトで次のように入力します。

```
C:¥> regedt32
```

Windows 98 の場合、次のように入力します。

```
C:¥> regedit
```

「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¥HOME ID` に移動します。
4. 「編集」メニューから「値の追加」を選択します。「値の追加」ダイアログ・ボックスが表示されます。
5. 「値の名前」テキスト・ボックスに `ORA_XA_REG_DLL` と入力します。
6. 「データタイプ」リスト・ボックスから「`REG_EXPAND_SZ`」を選択します。
7. 「OK」を選択します。「文字列エディタ」ダイアログ・ボックスが表示されます。
8. 「文字列」フィールドに `vendor.dll` と入力します。`vendor.dll` は、ベンダーから提供された TP モニターの DLL です。
9. 「OK」を選択します。レジストリ エディタでパラメータが追加されます。
10. 「レジストリ」メニューから「レジストリ エディタの終了」を選択します。
レジストリ エディタが終了します。

XA および TP モニターの情報

XA および TP モニターについては、次の一般的な情報を参照してください。

- 使用している TP モニター固有のドキュメント

Windows NT 固有の Oracle C++ Call Interface のメソッド

クラス `Statement` および `ResultSet` から `Refs` のコレクションを取得または設定するためのグローバル・メソッドは、Windows NT 用に次のように変更されています。

- Windows NT では、`getVector` のかわりに `getVectorOfRefs` を使用します。
- Windows NT では、`setVector` のかわりに `setVectorOfRefs` を使用します。

メソッド名は変更されていますが、パラメータの数およびデータ型は、これらのクラスの `Refs` に対する元の `getVector` メソッドおよび `setVector` メソッドと同じままです。

- Windows NT プラットフォームで実行するアプリケーションの場合、`Refs` のコレクションの取得および挿入にかぎり、これらの新規メソッドをコールする必要があります。
- Windows NT プラットフォームで実行しないアプリケーションの場合、既存の `getVector` メソッドおよび `setVector` メソッドをコールします。ただし、`Refs` とのベクター演算には、新規メソッドを使用することをお勧めします。

ResultSet クラス : Refs のコレクションのフェッチ

```
void getVectorOfRefs(ResultSet *rs, unsigned int index,  
    OCCI_STD_NAMESPACE::vector<Ref<T> > &vect);
```

このメソッドは、列索引によって指定された列値 (`Refs` のコレクション) を結果セットからフェッチします。

パラメータは次のとおりです。

- `rs`: `ResultSet` オブジェクト
- `index`: `Refs` のコレクションである列の列索引
- `vect`: `Refs` のフェッチ先となるベクター

Statement クラス : Refs のコレクションのフェッチ

```
void getVectorOfRefs(Statement *stmt, unsigned int index,  
    OCCI_STD_NAMESPACE::vector<Ref<T> > &vect);
```

このメソッドは、列索引によって指定された列値 (Refs のコレクション) を文からフェッチします。このメソッドは、OUT バインドおよびデータ操作言語 (DML) の RETURNING 句で使用されます。パラメータは次のとおりです。

- stmt: Statement オブジェクト
- index: Refs のコレクションである列の列索引
- vect: Refs のフェッチ先となるベクター

Statement クラス : Refs のコレクションの挿入

```
template <class T>  
void setVectorOfRefs(Statement *stmt, unsigned int paramIndex,  
    const OCCI_STD_NAMESPACE::vector<Ref<T> > &vect,  
    const OCCI_STD_NAMESPACE::string &sqltype);
```

このメソッドは、列索引によって指定された列に Refs のコレクションを挿入します。パラメータは次のとおりです。

- stmt: Statement オブジェクト
- paramIndex: Refs のコレクションである列の列索引
- vect: 列に挿入される Refs のベクター
- sqltype: データベースに作成されたコレクションのデータ型名

オブジェクトのコレクションをフェッチおよび挿入するためのグローバル・メソッドは、Windows NT 用に変更されています。インタフェースは、メソッド名、パラメータの数およびデータ型については変更されていませんが、Windows NT 用のテンプレート・パラメータ定義は異なります。特に、Windows NT 上のオブジェクト (オブジェクト・ポインタ) のテンプレート・メソッド `getVector` および `setVector` のテンプレート・パラメータには、次の API で示されるように `T*` のかわりに `T` と指定されます。

メソッドの使用方法はプラットフォーム間で違いはありません (ユーザーはこれらのメソッドに対するコールを変更する必要はまったくありません)。メソッドのコールでオブジェクト・ポインタとして渡されるテンプレートの引数は、他のプラットフォームではパラメータ `T*` ですが、Windows NT ではパラメータ `T` です。

ResultSet クラス : オブジェクトのコレクションのフェッチ

```

#ifdef WIN32COMMON
    template <class T>
    void getVector( ResultSet *rs, unsigned int index,
        OCCI_STD_NAMESPACE::vector< T > &vect ) ;
#else
    template <class T>
    void getVector( ResultSet *rs, unsigned int index,
        OCCI_STD_NAMESPACE::vector< T* > &vect ) ;
#endif

```

このメソッドは、列索引によって指定された列の結果セットからオブジェクトのコレクションをフェッチします。

パラメータは次のとおりです。

- rs: ResultSet オブジェクト
- index: 列索引
- vect: オブジェクトのフェッチ先となるベクター

Statement クラス : オブジェクトのコレクションのフェッチ

```

#ifdef WIN32COMMON
    template <class T>
    void getVector( Statement *stmt, unsigned int index,
        OCCI_STD_NAMESPACE::vector< T > &vect ) ;
#else
    template <class T>
    void getVector( Statement *stmt, unsigned int index,
        OCCI_STD_NAMESPACE::vector< T* > &vect ) ;
#endif

```

このメソッドは、列索引によって指定された列の文からオブジェクトのコレクションをフェッチします。このメソッドは、OUT バインドおよび DML の RETURNING 句で使用されます。パラメータは次のとおりです。

- stmt: Statement オブジェクト
- index: 列索引
- vect: オブジェクトのフェッチ先となるベクター

Statement クラス : オブジェクトのベクターの挿入

```
#ifndef WIN32COMMON
template <class T>
void setVector( Statement *stmt, unsigned int paramIndex,
               const OCCI_STD_NAMESPACE::vector< T > &vect,
               const OCCI_STD_NAMESPACE::string &sqltype) ;
#else
template <class T>
void setVector( Statement *stmt, unsigned int paramIndex,
               const OCCI_STD_NAMESPACE::vector<T* > &vect,
               const OCCI_STD_NAMESPACE::string &sqltype) ;
#endif
```

このメソッドは、列索引によって指定された列の文にオブジェクトのコレクションを挿入します。パラメータは次のとおりです。

- stmt: Statement オブジェクト
- paramIndex: 列索引
- vect: オブジェクトのフェッチ先となるベクター
- sqltype: データベースに作成されたコレクションのデータ型名

関連資料： Oracle XA ライブラリおよび XA 動的登録の使用の詳細は、『Oracle9i アプリケーション開発者ガイド - 基礎編』を参照してください。

OTT と INTYPE File Assistant の使用方法

OTT は、Oracle9i データベースで作成されて格納されている抽象データ型の C 構造体表現を作成するために使用します。

オブジェクトを使用するには、データベースに対して OTT を実行します。C 構造体を含むヘッダー・ファイルが生成されます。たとえば、データベースで PERSON 型が作成されている場合、PERSON の属性に相当する要素を持つ C 構造体が OTT により生成されます。さらに、C 構造体のインスタンスに対して NULL 情報を表す NULL インジケータが作成されます。

INTYPE ファイルは、変換対象となるオブジェクト型を OTT に示します。また、生成される構造体の命名も、INTYPE ファイルにより制御されます。INTYPE File Assistant は、開発者が INTYPE ファイルを作成するのを補助するウィザードです。

INTYPE ファイル内の CASE=LOWER などの CASE 指定は、INTYPE ファイル内の TYPE 文や TRANSLATE 文で特別にリストされていない C 識別子のみに適用されることに注意してください。INTYPE ファイルでは、TYPE Person や Type PeRsOn のように、大文字と小文字を適切に区別して指定することが重要です。

INTYPE File Assistant が INTYPE ファイルに生成する型名の大文字と小文字は、データベース内のものと同じです。デフォルトでは、データベース内の型はすべて大文字で作成されます。

大文字と小文字の区別をするには、データベースで型を作成する際に二重引用符を使用します。たとえば、次のように指定します。

```
CREATE TYPE "PeRsOn" AS OBJECT...
```

オブジェクト型の依存性は、Oracle INTYPE File Assistant ではチェックされません。INTYPE ファイルに含めるためにオブジェクト型を追加する際、依存関係を持つ他のオブジェクト型が INTYPE File Assistant によって追加されることはありません。

INTYPE File Assistant では、オブジェクト型や属性に ASCII 以外の文字が含まれている場合は、明示的に変換する必要があります。これらのオブジェクト型や属性は、変換結果が入力されるフィールドに、事前に定義されたタグ識別子が入って示されます。ユーザーは、該当するオブジェクト型や属性を C の識別子に変換したもので、このタグを上書きする必要があります。必要な変換がすべて入力されるまで、INTYPE File Assistant は INTYPE ファイルを作成しません。

Windows NT では、OTT はコマンドラインから起動できます。また、コマンドラインから構成ファイルに名前を付けることもできます。Windows NT の場合、構成ファイルは `ottcfg.cfg` で、`%ORACLE_HOME%\precomp\admin` にあります。

INTYPE File Assistant は、今回のリリースでは使用できません。また、リリース 10i でもサポートされません。

関連資料： OTT と INTYPE ファイルの詳細は、『Oracle Call Interface プログラマーズ・ガイド』を参照してください。

索引

B

bin ディレクトリ, 1-2
Borland C++ のサポート, xii

C

cdemomt.c, 1-3

E

emp 表, 1-3

I

include ディレクトリ, 1-2
INTYPE File Assistant, 2-10

L

LoadLibrary, 2-4

M

make.bat, 1-3

O

Object Type Translator (OTT), 2-10
OCI
 Oracle XA ライブラリ, 2-5
 アプリケーションの作成, 2-1
 概要, 1-2
 サンプル・プログラム, 1-3
 新機能, xi

 リリース 7.x の関数, xii
oci.dll, xii
oci.lib, xii, 2-3
ociw32.dll, xii
ociw32.lib, xii
OCI アプリケーション
 コンパイル, 2-2
 作成, 2-2
 実行, 2-4
 リンク, 2-3
OCI アプリケーションの作成, 2-1, 2-2
OCI アプリケーションの実行, 2-4
oci ディレクトリ, 1-2
Oracle Call Interface, 「OCI」を参照
Oracle XA ライブラリ
 OCI プログラムのコンパイルおよびリンク, 2-5
 概要, 2-4
 機能, 2-4
 その他のドキュメント, 2-7
 動的登録, 2-5
Oracle9i データベース
 トランザクション処理 (TP) モニター, 2-4
OTT (Object Type Translator), 2-10
ottcfg.cfg, 1-3

R

Required Support Files (RSF), 1-2

S

samples ディレクトリ, 1-3

X

XA, 「Oracle XA ライブラリ」を参照

き

機能

新しい, xi

共通マニュアルの参照先

OCI アプリケーションのコンパイルおよびリンク,
2-2, 2-3

OTT 構成ファイル, 2-11

XA リンク・ファイル名, 2-4

コマンドラインからの OTT の起動, 2-11

スレッド・セーフティ, 1-3

デモ・プログラム, 1-3

共有データ・モード, 2-2

こ

構成ファイル, 1-3

場所, 1-3

コンパイル

OCI アプリケーション, 2-2

Oracle XA と OCI, 2-5

Oracle XA ライブラリ, 2-5

さ

サンプル・プログラム, 1-3

て

ディレクトリ構造, 1-2

デモ・プログラム, 1-3

と

動的登録

Oracle XA ライブラリ, 2-5

トランザクション処理 (TP) モニター

Oracle9i データベースとの対話, 2-4

種類, 2-4

その他のドキュメント, 2-7

へ

ヘッダー・ファイル

場所, 1-2, 2-2

ま

マルチスレッド, 1-3

ら

ライブラリ

oci.lib, 2-3

り

リンク

OCI アプリケーション, 2-3

Oracle XA と OCI, 2-5

Oracle XA ライブラリ, 2-5

れ

レジストリ

regedt32, 2-6