

**Oracle® Essbase**

**SQL Interface Guide**

リリース 11.1.2.3.000

Essbase SQL Interface Guide, 11.1.2.3.000

Copyright © 1998, 2013, Oracle and/or its affiliates. All rights reserved.

著者: EPM 情報開発チーム

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。

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 RIGHTS:

Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

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際、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より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。

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。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。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

---

ドキュメントのアクセシビリティについて .....	5
<b>第 1 章 SQL インタフェースの概要</b> .....	7
SQL インタフェース・プロセスの理解 .....	7
SQL データ・ソースまたはリレーショナル・データ・ソースを使用する準備 .....	8
<b>第 2 章 データ・ソースの構成</b> .....	9
データ・ソースの構成について .....	9
Windows でのデータ・ソースの構成 .....	9
UNIX でのデータ・ソースの構成 .....	10
Oracle Call Interface の使用 .....	10
<b>第 3 章 複数のテーブルのデータ・ソースの準備</b> .....	13
複数のテーブルのデータ・ソースを準備する方法 .....	13
アクセス権限の要件 .....	13
優先される方法 - 1 つのテーブルまたはビューの作成 .....	13
データ・ロード中のテーブルの結合 .....	13
<b>第 4 章 SQL データのロード</b> .....	15
データのロードと次元の構築について .....	15
代替変数の使用 .....	15
代替変数のルール .....	16
代替変数の作成と使用 .....	16
ルール・ファイルの作成と SQL データ・ソースの選択 .....	16
SQL データ・ソースの選択 .....	17
SQL クエリーの作成(オプション) .....	17
集約ストレージ・データベースへの複数 SQL データ・ロードの並列での実行 .....	18
<b>第 5 章 非 DataDirect ドライバの使用</b> .....	21
非 DataDirect ドライバについて .....	21
非 DataDirect ドライバ用の構成ファイルの作成 .....	21
構成ファイル内で使用されるキーワードと値 .....	22
Windows でのドライバ名の検索 .....	24

UNIX でのドライバ名の検索 .....	24
非 DataDirect ドライバの構成 .....	24
<b>付録 A. Teradata データ・ソースからの高速データ・ロードの使用可能化 .....</b>	<b>27</b>
Teradata データ・ソースの使用 .....	27
必要な Teradata ソフトウェアのインストール .....	27
Teradata をデータ・ソースとして構成 .....	28
Teradata Parallel Transporter を使用するための環境の設定 .....	30
Teradata Parallel Transporter を使用した Teradata データのロード .....	31
Teradata TPT-API ロード設定のカスタマイズ .....	32
TD_MAX_SESSIONS .....	32
TD_TRACE_OUTPUT .....	32
TD_TRACE_LEVEL .....	33
TD_TENACITY_HOURS .....	33
TD_TENACITY_SLEEP .....	33
Unicode 文字セットとマルチバイト文字セットのサポート .....	34
<b>索引 .....</b>	<b>35</b>

---

# 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。

## Access to Oracle Support

Oracle サポート・サービスでは、My Oracle Support を通して電子支援サービスを提供しています。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。



# 1

## SQLインタフェースの概要

### この章の内容

SQL インタフェース・プロセスの理解.....	7
SQL データ・ソースまたはリレーショナル・データ・ソースを使用する準備.....	8

## SQL インタフェース・プロセスの理解

次元を構築し、SQL データベースおよびリレーショナルデータベースから値をロードするために、SQL インタフェース機能を使用できます。たとえば、要約データのみを取得を指定する SQL 文を実行できます。

Oracle Essbase Administration Services、MaxL、または Oracle Essbase Administration Services を使用してロードできる、スプレッドシート・データ・ソースまたはテキストファイル・データ・ソース用の SQL インタフェースは必要ありません。Oracle Essbase Database Administrator's Guide および Oracle Essbase Technical Reference を参照してください。

SQL インタフェースを使用すると、Unicode モードのリレーショナル・データベースから Unicode モードの Oracle Essbase アプリケーションにデータをロードできます。Unicode の Essbase での実装については、Oracle Essbase Database Administrator's Guide を参照してください。

SQL インタフェースは、Administration Services と連携してデータを取得します：

1. Administration Services を使用して、SQL で SELECT ステートメントを記述します。
2. SQL インタフェースは、SQL サーバーまたはリレーショナル・データベース・サーバーにステートメントを渡します。

**注：** 必要に応じて、SQL インタフェースは、非 SQL データベースに適した要求に SQL 文を変換します。

3. データロード・ルール・ファイルで定義されるルールを使用して、SQL インタフェースはデータベース・サーバーから受け取ったレコードを解釈します。第4章「SQL データのロード」を参照してください。
4. SQL インタフェースは、解釈済の要約レベル・データをデータベースにロードします。

# SQL データ・ソースまたはリレーショナル・データ・ソースを使用する準備

SQL インタフェースは、Essbase サーバーのインストール時にインストールされます。初期構成タスクの詳細は、Oracle Enterprise Performance Management System Installation and Configuration Guide を参照してください。

▶ SQL データ・ソースまたはリレーショナル・データ・ソースを使用する準備をするには、次の手順を実行します:

- 1 ODBC ドライバを構成し、データ・ソースにポイントします。第 2 章「データ・ソースの構成」を参照してください。
- 2 データが複数テーブルに含まれる場合、1 つのアクションを実行します:
  - SQL インタフェースを使用する前に、SQL データベースで、1 つのテーブルまたはビューを作成します。
  - データ・ロード中に、Administration Services に SELECT ステートメントを入力することにより、テーブルを結合します。  
手順は、13 ページの「複数のテーブルのデータ・ソースを準備する方法」を参照してください。
- 3 管理サービス・コンソールのデータ準備エディタを使用することによりデータ・ソース接続を確認し、SQL ソース・ファイルを開きます。第 4 章「SQL データのロード」を参照してください。
- 4 SQL インタフェースに Essbase データベース内で使用する SQL データの解釈方法を示すルール・ファイルを作成します。第 4 章「SQL データのロード」を参照してください。

これらの手順が完了したら、データをロードするか、次元を構築します; 第 4 章「SQL データのロード」を参照してください。



# 2

## データ・ソースの構成

### この章の内容

データ・ソースの構成について .....	9
Windows でのデータ・ソースの構成 .....	9
UNIX でのデータ・ソースの構成 .....	10
Oracle Call Interface の使用 .....	10

## データ・ソースの構成について

SQL インタフェースを使用してデータにアクセスする前に、各データ・ソースのオペレーティング・システムと各データ・ソースに必要とされるドライバを構成する必要があります。

Essbase のインストール時に DataDirect ODBC ドライバが提供されます。

**注：** Oracle 11g データベースに接続する DataDirect ODBC ドライバは、マルチスレッド接続を使用可能にし、大文字変換を使用不可にするように構成されています。

各 DataDirect ドライバの詳細で、ドライバ固有の情報は、DataDirect Connect for ODBC のリファレンスを参照してください。このリファレンスの場所(通常/`EPM_ORACLE_HOME/common/... /books/odbc/odbceref/`ディレクトリ内にある)は、プラットフォームによって異なります。

非 DataDirect ドライバを構成する、または DataDirect ODBC ドライバのデフォルト設定を変更するには、第 5 章「非 DataDirect ドライバの使用」を参照してください。

サポートされているドライバのリストについては、Oracle Enterprise Performance Management System Installation and Configuration Guide を参照してください。

## Windows でのデータ・ソースの構成

Windows では、ODBC Administrator を使用してデータ・ソースを構成します。

▶ ODBC Administrator を使用してデータ・ソースを構成するには、次の手順を実行します:

1 「開始」、「管理ツール」、「データ・ソース(ODBC)」の順に選択します。

- 2 データ・ソースを選択または追加し、ドライバに関する必要な情報を入力します。  
詳細な手順は、ODBC プロバイダのドキュメントを参照してください。

## UNIX でのデータ・ソースの構成

▶ UNIX でデータ・ソースを構成するには、次の手順を実行します:

- 1 \$ARBORPATH/bin/.odbc.ini ファイルを開き、データ・ソースの説明を追加します。
- 2 データ・ソースを追加するか、ドライバ製品またはデータ・ソースを変更する場合、ODBC 接続およびデータ・ソース名やドライバ製品名などの構成情報を更新するために、.odbc.ini を編集する必要がある場合があります。更新の手順と要件は、プラットフォームによって異なります。

### 例: DB2 に対する.odbc.ini の更新

次のシナリオを想定します:

- isaix7 サーバー上にある「tbc\_data」という名前の DB2 データベースへの接続
- DataDirect 6.1 Wire Protocol ドライバを起動する ODBC データ・ソース (「db2data」という名前)を使用

.odbc.ini ファイルを編集するには、vi コマンドを使用して、次のステートメントを挿入します:

```
[ODBC Data Sources]
db2data=DB2 Source Data on isaix7
...
[db2data]
Driver=/vol1/Oracle/Middleware/EPMSysstem11R1/common/ODBC/Merant/6.1/lib/ARdb225.so
Database=tbc_data
IpAddress=isaix7
TcpPort=50000
```

**注:** 各 DataDirect ドライバの詳細で、ドライバ固有の情報は、DataDirect Connect for ODBC のリファレンスを参照してください。このリファレンスの場所(通常/EPM\_ORACLE\_HOME/common/ ... /books/odbc/odbceref/ディレクトリ内にある)は、プラットフォームによって異なります。

## Oracle Call Interface の使用

ODBC にかわる手法として Oracle Call Interface (OCI)を使用すると、データ・ロードと次元構築のパフォーマンスが大幅に向上します。この手法では、データ準備エディタを使用して OCI 接続識別子を指定します。

Oracle OCI 接続識別子を使用するには、データ・ソース名(DSN)識別に次の構文を使用します:

```
host:port/Oracle_service_name
```

ホスト・サーバー名が `myserver`、ポート番号が `1521`、および Oracle サービス名が `orcl.us.oracle.com` の OCI 接続識別子の例を次に示します:

```
myserver:1521/orcl.us.oracle.com
```

Oracle Essbase Administration Services Online Help も参照してください。

AIX システムでは、OCI を使用して SQL データをロードする際に非同期入出力を使用可能にする必要があります。使用可能にしない場合は、データ・ロードが失敗して次のメッセージが表示されます:

```
Cannot get async process state. Essbase Error(1021104): Cannot load instant client shared library [libociei.so].Make sure that the required binaries are present with correct environment variables set.
```

▶ AIX で非同期入出力を使用可能にする手順は、次のとおりです:

**1 aio0 ドライバの状態を判断するには、次のコマンドを実行します:**

```
lsdev -C -l aio0
```

**出力例:**

```
aio0 Defined Asynchronous I/O
```

「Defined」は、aio0 ドライバがシステムにインストールされているがアプリケーションから使用できないことを示します。ドライバをアプリケーションで使用できない場合には、aio0 ドライバの状態を「Defined」から「Available」に変更します。

**2 AIX の `cfgmgr` コマンドを実行します:**

```
cfgmgr -l aio0
```

**3 「Available」の状態を恒久的にする(システムの再起動後も同じ状態を維持する)には、AIX の `chdev` コマンドを発行します:**

```
chdev -l aio0 -P -a autoconfig='available'
```

変更を有効にするために、システムを再起動する必要はありません。

**出力例:**

```
aio0 changed
```

- 4 aio0 ドライバの状態を確認するには、次のコマンドを実行します:

```
lsdev -C -l aio0
```

出力例:

```
aio0 Available Asynchronous I/O
```

# 3

## 複数のテーブルのデータ・ソースの準備

### この章の内容

複数のテーブルのデータ・ソースを準備する方法.....	13
データ・ロード中のテーブルの結合.....	13

## 複数のテーブルのデータ・ソースを準備する方法

- SQL インタフェースを使用する前に、SQL データベースで1つのテーブルまたはビューを作成します。
- データのロード時に、管理サービス・コンソールに SELECT ステートメントを入力することによりテーブルを結合します。

### アクセス権限の要件

1つのテーブルまたはビューの作成、およびテーブルの結合には、データが格納されるテーブルへの SELECT アクセス権限が必要です。1つのテーブルまたはビューを作成するには、SQL データベースで CREATE アクセス権限が必要です。

### 優先される方法 - 1つのテーブルまたはビューの作成

SQL データベース・サーバーは、複数テーブルの SELECT ステートメントを処理するよりも効率的に、1つのテーブルから読み取り、1つのビューを維持します。したがって、SQL インタフェースを作成する前に1つのテーブルまたはビューを作成すると、SQL サーバーで必要な処理時間が大幅に短縮されます。

### データ・ロード中のテーブルの結合

CREATE 権限を取得できない場合、管理サービスを使用して、データ・ロード中にテーブルを結合する必要があります。

- ▶ データ・ロード中にテーブルを結合するには、次の手順を実行します：
  - 1 関連するデータが保管されているテーブルへの SELECT アクセス権限を取得します。

## 2 管理サービス・コンソールで、テーブルを結合する SELECT ステートメントを作成します。

1. Essbase にロードするデータを含むテーブルと列を識別します。
2. 「ファイル」を選択してから「SQL を開く」を選択して、「SQL データ・ソースを開く」を表示します。

Oracle Essbase Administration Services Online Help を参照してください。

3. テーブルを結合する SELECT ステートメントを書き込みます。

[17 ページの「SQL データ・ソースの選択」](#)と[17 ページの「SQL クエリーの作成\(オプション\)」](#)を参照してください。

**注：** Essbase は、構文を確認せずにデータベースへ SELECT ステートメントを渡します。

# 4

## SQLデータのロード

### この章の内容

データのロードと次元の構築について.....	15
代替変数の使用 .....	15
ルール・ファイルの作成と SQL データ・ソースの選択.....	16
集約ストレージ・データベースへの複数 SQL データ・ロードの並列での実行 .....	18

## データのロードと次元の構築について

1つ以上の SQL データ・ソースを構成し、複数のテーブルのデータを作成した後、Oracle Essbase Administration Services を使用してデータをロードし、次元を構築できます。

▶ データをロードして次元を構築するには、次の手順を実行します:

1 代替変数の使用を予定している場合は、それを作成します。

15 ページの「代替変数の使用」を参照してください。

2 ルール・ファイルを作成し、データ・ソースを選択します。

参照:

- 16 ページの「ルール・ファイルの作成と SQL データ・ソースの選択」
- 18 ページの「集約ストレージ・データベースへの複数 SQL データ・ロードの並列での実行」

3 Essbase データベースにデータをロードします。

Oracle Essbase Administration Services Online Help を参照してください。

## 代替変数の使用

SQL 文字列とデータ・ソース名で代替変数を使用すると、複数のデータ・ソースに1つのルール・ファイルを使用できます。1つの代替変数は、Essbase サーバー上のすべてのアプリケーションとデータベース、または特定のアプリケーションまたはデータベースに適用できます。

データ・ソース名(DSN)に対する代替変数を定義して、代替変数名のルール・ファイルを指定することもできます。

## 代替変数のルール

- 有効で適切な SQL 値のみを使用してください。Essbase は、値を検証しません。
- 特に引用符(一重引用符と二重引用符)に注意してください。必要な規則は、データベースによって異なります。
- アンパサンド(&)は、代替変数に対する Essbase の識別子であるため、SELECT、FROM、または WHERE 句内の SQL 演算子をアンパサンドで開始しないでください。

## 代替変数の作成と使用

▶ 代替変数を作成して使用するには、次の手順を実行します:

- 1 Oracle Essbase Administration Services Online Help の手順を使用して、代替変数を作成します。
- 2 ルール・ファイルを編集するには、「ファイル」、「SQL を開く」の順に選択してから、SQL データ・ソースを開きます。

Oracle Essbase Administration Services Online Help を参照してください。

- 3 「SQL データ・ソースを開く」ダイアログ・ボックスで、アクションを実行します:
  - DSN の代替変数を指定するには、「代替変数」を選択し、代替変数を 1 つ選択します。
  - クエリーで代替変数を指定するには、「Select」、「From」、または「Where」に、"field=value"文字列ではなく(アンパサンドを前に付けて)代替変数を入力します。
- 4 「OK/取得」をクリックして、ルール・ファイルのデータを取得します。

**注:** データ・ロードまたは次元の構築のためにルール・ファイルを使用する前に、代替変数の値を設定する必要があります。

## ルール・ファイルの作成と SQL データ・ソースの選択

1. データロード・ルール・ファイルを作成します; Oracle Essbase Administration Services Online Help を参照してください。

データ・ロード・ルールおよび次元構築ルールは、データが Essbase データベースにロードされるときに Essbase がデータに対して実行する、あるいは Essbase アウトラインの次元を構築するときに使用される一連の操作です。これらの操作は、ルール・ファイルに保管されます。

2. SQL データ・ソースを選択します。

17 ページの「SQL データ・ソースの選択」を参照してください。



3. Essbase で SQL クエリーを作成するプランがある場合は、17 ページの「SQL クエリーの作成(オプション)」を参照してください。

## SQL データ・ソースの選択

▶ SQL データ・ソースを選択するには、次の手順を実行します:

- 1 管理サービス・コンソールで、「データ準備エディタ」またはルール・ファイルを開きます。
- 2 「ファイル」を選択し、次に「SQL を開く」を選択します。
- 3 「データベースの選択」で、Essbase サーバー、アプリケーションおよびデータベースの名前を入力し、「OK」をクリックします。
- 4 「SQL データ・ソースを開く」で、データ・ソースまたは代替変数を選択し、必要な情報を入力します。

Oracle Essbase Administration Services Online Help の SQL データベースを開くに関する項を参照してください。

- 5 「OK/取得」をクリックします。
- 6 「SQL 接続」に、ソース・データベースのユーザー名とパスワードを入力し、「OK」をクリックします。

### データ・ソース・ファイルに関する事実:

- サーバー・コンピュータでデータソース・ファイルが構成してある必要があります。
- UNIX プラットフォームでは、SQL データ・ソース・ファイルのパスは、.odbc.ini ファイルで定義されています。
- Windows では、SQL ソース・ファイルのパスが ODBC Administrator で定義されていない場合、「SQL の定義」ダイアログ・ボックスの「データベース」ボックスで入力できます。
- パスが定義されていない場合、Essbase は、Essbase サーバーが実行されているディレクトリ内で、データ・ソース・ファイルを探します。

## SQL クエリーの作成(オプション)

取得するデータを選択するためにテーブルまたはビューを作成するかわりに、データ・ロードの実行時に SELECT ステートメントを記述できます。

**注:** Essbase で SELECT ステートメントを作成する場合は通常、ソース・データベースでテーブルまたはビューを作成するより時間がかかります。

「SQL データ・ソースを開く」ダイアログ・ボックスの「SQL 文」ボックスでは、SQL クエリーの記述に役立つ Select、From、および Where テキスト・ボックスが提供されます。複数のデータ・ソースの提供、レコードの表示のフィルタ、およ

びデータ準備エディタに表示されるレコードの順序付けと分類方法を指定できません。

## 集約ストレージ・データベースへの複数 SQL データ・ロードの並列での実行

集約ストレージ・データベースに SQL データをロードするときは、データの並行ロードに、最大 8 つのルール・ファイルを使用できます。各ルール・ファイルには、同じ認証情報(SQL ユーザー名およびパスワード)を使用する必要があります。

Essbase は、データ値がソートおよび蓄積される、複数の一時集約ストレージ・データ・ロード・バッファ(各ルール・ファイルごとに 1 つ)を初期化します。データがデータ・ロード・バッファに完全にロードされると、Essbase は、すべてのバッファのコンテンツを 1 つの操作でデータベースにコミットします。この操作は、個別にバッファをコミットするより時間がかかりません。

**注：** この機能は、`import ... data to load_buffer with buffer_id` 文法を使用してバッファにデータをロードし、`import ... data from load_buffer with buffer_id` 文法を使用してバッファのコンテンツをデータベースに明示的にコミットするものとは異なります。集約ストレージ・データ・ロード・バッファの詳細は、Oracle Essbase Database Administrator's Guide を参照してください。

MaxL で、`using multiple rules_file` 文法を使用して、`import databaseMaxL` ステートメントを使用します。Oracle Essbase Technical Reference を参照してください。

次の例では、2 つのルール・ファイル(rule1.rul および rule2.rul)から SQL データがロードされます:

```
import database AsoSamp.Sample data
connect as TBC identified by 'password'
using multiple rules_file 'rule1' , 'rule2'
to load_buffer_block starting with buffer_id 100
on error write to "error.txt";
```

ルール・ファイルのリストを指定する場合は、カンマで区切られたルール・ファイル名の文字列(.rul 拡張子を除く)を使用します。ルール・ファイルのファイル名は、8 バイトを越えてはならず、ルール・ファイルは Essbase サーバーに常駐している必要があります。

各ルール・ファイルのデータ・ロード・バッファを初期化する際、Essbase は、リストの最初のルール・ファイル(たとえば、rule1 に対する ID 100)に指定する、開始データ・ロード・バッファ ID を使用し、後続のデータ・ロード・バッファごとに、1 ずつ ID 番号を増分します(たとえば、rule2 に対して ID 101)。

デフォルトで、SQL インタフェースは Essbase で提供される DataDirect ODBC ドライバの並列接続を使用不可にします。この機能では並列 SQL 接続が必要なので、構成ファイル(ARBORPATH/bin/esssql.cfg)を作成して、使用する ODBC ドライバのデフォルト設定を変更する必要があります。Essbase で提供される SQL Server Wire

Protocol ドライバの `esssql.cfg` ファイルの次の例では、並列 SQL 接続が使用可能になります:

```
[  
Description "SQL Server Wire Protocol"  
DriverName ARMSSS  
UpperCaseConnection 0  
UserId 1  
Password 1  
Database 1  
SingleConnection 0  
IsQEDriver 0  
]
```

変更を有効にするには、Essbase サーバーを再起動する必要があります。



## この章の内容

非 DataDirect ドライバについて .....	21
非 DataDirect ドライバ用の構成ファイルの作成 .....	21
非 DataDirect ドライバの構成 .....	24

## 非 DataDirect ドライバについて

すべての非 DataDirect ドライバ(すべてのデータ・ソース用に Essbase で配布される非 DataDirect ドライバ)を構成する必要があります。

全部ではなく一部の非 DataDirect ドライバは、Essbase 用にテスト済で、サポートされています。適格なドライバおよびデータ・ソースの詳細は、Oracle Enterprise Performance Management System Installation and Configuration Guide を参照してください。

このセクションの情報はまた、Essbase で配布される DataDirect ODBC のデフォルト設定を変更する場合にも適用されます。

## 非 DataDirect ドライバ用の構成ファイルの作成

非 DataDirect ドライバを使用してデータベースに接続する場合、または Essbase で配布される DataDirect ODBC ドライバのデフォルト設定を変更する場合に、構成ファイル(ARBORPATH/bin/esssql.cfg)を作成します。

**注：** Oracle Hyperion Enterprise Performance Management System コンフィグレータでは、ODBC ドライバの構成中、および Essbase サーバーの構成、クラスタの構成、JVM のセットアップ中に essbase.cfg にエントリが追加される場合があります。

デフォルトでは、essbase.cfg のデータ・ソース・エントリの先頭にセミコロン(;)のコメント・インディケータがあるために、一部の ODBC データ・ソース・ドライバが無効になっている可能性があります。非 DataDirect データ・ソースに接続できない場合は、essbase.cfg を編集して、使用しているデータ・ソースがリストされ、セミコロンのコメント・インディケータによって無効になっていないことを確認する必要があります。

- 22 ページの「構成ファイル内で使用されるキーワードと値」
- 24 ページの「Windows でのドライバ名の検索」
- 24 ページの「UNIX でのドライバ名の検索」

## 構成ファイル内で使用されるキーワードと値

構成ファイルは、ドライバ・ファイル名(DriverName)と、オプションの説明(Description)、二重引用符で囲む必要がある値(Description のデフォルト値は")を含む必要があります;追加のキーワードが含まれる場合があります、その値は、表 1 で示すように、0 または 1 です。

キーワードと値は、少なくとも 1 つのスペースで区切る必要があります。一連のキーワードと各ドライバの値は、カッコ([ ])で囲む必要があります。

異なるドライバには異なる値が必要な場合があります。特定の情報については、ドライバのドキュメントを参照してください。

次の例で、Oracle の最初のエントリにより、Essbase で配布される DataDirect ドライバのデフォルト設定が変更されます。2 つ目のエントリは、非 DataDirect ドライバである、Microsoft SQL Server 用のエントリです。

### サンプル `esssql.cfg`

---

```
[
Description "Oracle Wire Protocol"
DriverName ARORA
UpperCaseConnection 0
UserId 1
Password 1
Database 1
SingleConnection 0
IsQEDriver 1
]

[
Description "Microsoft SQL Server 32-bit"
DriverName SQLSRV32
UpperCaseConnection 0
UserId 1
Password 1
Database 1
SingleConnection 0
IsQEDriver 0
]

[
Description "Oracle BI Server"
DriverName libnqsodbc
UpperCaseConnection 0
UserId 1
Password 1
Database 1
```

```
SingleConnection 1
ConvertUTF16toUTF8 1
]
```

指定されていない値に適用されるデフォルトです。構成ファイル内で適用されるデフォルトは、esssql.cfg ファイルがない場合に適用される Essbase のデフォルト値とは異なります。

**表 1 構成ファイルのキーワードおよび非 DataDirect ドライバの値**

キーワード	値 = 0	値 = 1
UserId	ユーザー ID が不要(デフォルト)	ユーザー ID が必要
Password	パスワードが不要(デフォルト)	パスワードが必要
Database	データベース名が不要(デフォルト)	データベース名が必要
Server	サーバー名が不要(デフォルト)	サーバー名が必要
Application	アプリケーション名が不要(デフォルト)	アプリケーション名が必要
Dictionary	ディクショナリ名が不要(デフォルト)	ディクショナリ名が必要
Files	ファイル名が不要(デフォルト)	ファイル名が必要
SingleConnection	ドライバがスレッドセーフ - 複数のアクティブ接続が許可される <b>注：</b> 非 DataDirect ドライバ、またはこれがデフォルトである Oracle 11g データベースへの接続に使用されない DataDirect ドライバには推奨されません; 不安定になる場合があります。	ドライバがスレッドセーフではない - 1 つのアクティブ接続が許可される Oracle 11g データベースへの接続に使用されるものを除く、すべての DataDirect ドライバのデフォルトおよび推奨事項。
UpperCaseConnection	ドライバに大文字と小文字の区別あり - 接続情報が変換されない(デフォルト) <b>ヒント：</b> データベース・サーバーへの接続が失敗し、アプリケーション・ログに「ユーザー名またはパスワードが無効です。ログオンが拒否されました」というメッセージが表示された場合、データベース内のユーザー名およびパスワードの大文字と小文字の区別を確認し、管理サービス・コンソールに入力している内容と比較してください。大文字と小文字の区別をオフにするには、この値を 0 から 1 に変更します。	ドライバに大文字と小文字の区別なし - 接続情報が大文字に変換される
IsQEDriver	ドライバが非 DataDirect のドライバ(デフォルト)	ドライバが DataDirect ドライバ <b>注：</b> DataDirect ドライバの構成情報を指定できます。たとえば、Essbase がサポートしない DataDirect ドライバのバージョンの情報を指定できます。

キーワード	値 = 0	値 = 1
ConvertUTF16toUTF8	UTF8 への UTF16 データの変換なし(これはデフォルトです)。	Oracle BI データ・ソースから UTF8 に UTF16 エンコード・データを変換します。これは、OBI から Essbase に SQL データをロードするために UNIX で必要です。

**注：** Oracle 11g データベースに接続する DataDirect ODBC ドライバは、マルチスレッド接続を使用可能にし、大文字変換を使用不可にするように構成されています。SQL Server Wire Protocol ドライバでマルチスレッド接続を使用可能にするには、[18 ページの「集約ストレージ・データベースへの複数 SQL データ・ロードの並列での実行」](#)を参照してください。

## Windows でのドライバ名の検索

▶ Windows でドライバ名を検索するには、次の手順を実行します：

- 1 [9 ページの「Windows でのデータ・ソースの構成」の手順 1](#)の方法を使用して、**ODBC Administrator** を起動します：

「ODBC Data Source Administrator」のダイアログ・ボックスが開きます。

構成済のデータ・ソースは「ユーザー・データ・ソース」ボックスに表示されます。適切に構成されていないが、「ユーザー・データ・ソース」ボックスに表示されているドライバは、無視できます。

- 2 「ドライバ」タブを選択します。
- 3 右へスクロールすることにより、優先ドライバのファイル名を取得します。  
たとえば、Microsoft Access Driver のファイル名は、ODBCJT32.DLL です。

## UNIX でのドライバ名の検索

▶ UNIX でドライバ名を検索するには、`.odbc.ini` ファイルを表示します。

[10 ページの「UNIX でのデータ・ソースの構成」](#)を参照してください。

## 非 DataDirect ドライバの構成

Essbase は、ドライバの名前やパスワードで大文字と小文字が区別されるかなど、DataDirect ドライバの基本的な構成情報を認識します。非 DataDirect ドライバの場合、あるいは Essbase で配布される DataDirect ドライバのデフォルト設定を変更する場合は、構成情報を提供する必要があります。

▶ 構成情報を提供するには：

- 1 `esssql.cfg` という名前の構成ファイル(テキスト・ファイル)を作成します。



**2 Essbase Server 上の\$ARBORPATH/bin ディレクトリにファイルを配置します。**

デフォルト・インストールでは、ARBORPATH は EPM\_ORACLE\_INSTANCE/  
EssbaseServer/essbaseserver1 です

**注：** 構成ファイルを作成しない場合、Essbase で、SQL データベースへ接続できない場合があるデフォルト値が使用されます。



# A

## Teradata データ・ソースからの 高速データ・ロードの使用 可能化

### この付録の内容

Teradata データ・ソースの使用 .....	27
必要な Teradata ソフトウェアのインストール .....	27
Teradata をデータ・ソースとして構成 .....	28
Teradata Parallel Transporter を使用するための環境の設定 .....	30
Teradata Parallel Transporter を使用した Teradata データのロード .....	31
Teradata TPT-API ロード設定のカスタマイズ .....	32
Unicode 文字セットとマルチバイト文字セットのサポート .....	34

## Teradata データ・ソースの使用

Teradata ツールおよびユーティリティの Teradata Parallel Transporter(TPT)を使用して、データ・ロードのパフォーマンスを大幅に改善できます。この方法では、データベース・スキーマを抽出するために ODBC が使用されます;その後、TPT によってデータが取得されます。

Essbase がデータ・ソースとしてサポートする Teradata データベースのバージョン、およびサポートされている Teradata ODBC ドライバの詳細は、Oracle Enterprise Performance Management System 動作保証マトリックス(<http://www.oracle.com/technetwork/middleware/bi-foundation/hyperion-supported-platforms-085957.html>)を参照してください。

## 必要な Teradata ソフトウェアのインストール

お客様には、正しい Teradata ライセンスを持ち、正しいバージョンの ODBC を Essbase サーバー・コンピュータ上でインストールして構成する責任があります。インストールの手順は、Teradata のドキュメントを参照してください。

- Teradata ツールおよびユーティリティから、Teradata Parallel Transporter Export Operator、Shared ICU Libraries for Teradata、Teradata GSS Client、および CLI をインストールします。(Linux インストールの場合、GCC 3.3 により構築されたライブラリを選択します。)
- Teradata ODBC ドライバをインストールします。

# Teradata をデータ・ソースとして構成

▶ Teradata をデータ・ソースとして構成するには:

1 Teradata ドライバをインストールします。このドライバは、Teradata から取得する必要があります。

- Oracle Essbase Studio では JDBC ドライバを使用します。JDBC Teradata ドライバは、Essbase Studio サーバーが実行されているコンピュータにインストールする必要があります。

Essbase Studio では JDBC Teradata ドライバを使用して、キューブをストリーミング・モードで配置します。

キューブを非ストリーミング・モードで配置するには、Essbase サーバーが実行されているコンピュータに ODBC ドライバをインストールする必要があります。

- Essbase サーバーでは ODBC ドライバを使用します。ODBC Teradata ドライバは、Essbase サーバーが実行されているコンピュータにインストールする必要があります。

2 Oracle Process Manager and Notification Server (OPMN) サービス: EPM\_epmsystem1 を使用して、Windows の「サービス」パネルから Essbase サーバーを停止します。

3 OPMN 構成ファイル(opmn.xml)をバックアップします。

例:

```
C:\Oracle\Middleware\user_projects\epmsystem1\config\OPMN\opmn\opmn.xml
```

4 opmn.xml ファイルをテキスト・エディタで開きます。

5 Teradata ドライバを適切にロードするには、Teradata ライブラリの場所を指すステートメントを opmn.xml ファイルに含める必要があります。

1. opmn.xml ファイルで次のステートメントを探します:

```
<variable id="ESS_CSS_JVM_OPTION7" value="-  
Djava.util.logging.config.class=oracle.core.ojdl.logging.LoggingConfiguration"  
>
```

2. このステートメントの後に、次のようなステートメントを追加します:

```
<variable append="true" id="PATH" value="C:\Program Files\Teradata  
\Client\14.00\Shared ICU Libraries for Teradata\lib"/>
```

6 Essbase サーバーで Teradata データ・ソースを使用し、OPMN を使用して Oracle Essbase エージェント・プロセスを監視および制御する場合は、使用しているオペレーティング・システムの変数で opmn.xml ファイルを更新する必要があります。

**注:** 絶対パスの値にスペースを含めることはできません。絶対パスの値の例は 64 ビットのマシン構成に基づいています。

## 64 ビット Windows

次の変数を追加します:

- TWB\_ROOT: Teradata ルート
- PATH: Teradata 共有ライブラリ
- PATH: Teradata クライアント DLL ライブラリ
- PATH: Teradata 呼出しレベル・インタフェース・バージョン 2 ルーチン
- PATH: Teradata メッセージ DLL ライブラリ

64 ビット Windows の例:

```
<variable id="TWB_ROOT" value="C:\PROGRA~1\Teradata\Client\14.00"/>
<variable append="true" id="PATH" value="C:\PROGRA~1\Teradata\Client\14.00\SHARED~1\lib"/>
<variable append="true" id="PATH" value="C:\PROGRA~1\Teradata\Client\14.00\TERADA~1\bin64"/>
<variable append="true" id="PATH" value="C:\PROGRA~1\Teradata\Client\14.00\CLIV2"/>
<variable append="true" id="PATH" value="C:\PROGRA~1\Teradata\Client\14.00\TERADA~1\msg64"/>
```

## 64 ビット AIX

次の変数を追加します:

- LIBPATH: Teradata ODBC ライブラリ
- LIBPATH: Teradata 共有ライブラリ
- LIBPATH: Teradata ODBC ドライバのロードに必要な ODBC コンポーネント
- LIBPATH: Teradata クライアント・ライブラリ
- COPERR: errmsg.txt ファイルが存在するディレクトリ
- NLSPATH: Teradata メッセージ・ライブラリ

64 ビット AIX の例:

```
<variable append="true" id="LIBPATH" value="/opt/teradata/client/ODBC_64/lib"/>
<variable append="true" id="LIBPATH" value="/opt/teradata/client/13.10/tdicu/lib64"/>
<variable append="true" id="LIBPATH" value="/usr/odbc/lib:/usr/odbc/drivers"/>
<variable append="true" id="LIBPATH" value="/usr/lib:/usr/teragss/aix-power/client/lib"/>
<variable id="COPERR" value="/usr/libperion/essbase"/>
<variable id="NLSPATH" value="/opt/teradata/client/13.10/odbc_32/msg/%N"/>
<variable append="true" id="NLSPATH" value="/usr/lib/nls/msg/%L/%N"/>
<variable append="true" id="NLSPATH" value="/usr/lib/nls/msg/%L/%N.cat"/>
```

## 64 ビット LINUX

次の変数を追加します:

- TWB\_ROOT: Teradata ルート
- TD\_ICU\_DATA: Teradata 共有ライブラリ
- NLSPATH: Teradata ODBC メッセージ・ライブラリ
- COPERR: errmsg.txt ファイルが存在するディレクトリ
- COPLIB: libcliv2.so ライブラリ・ファイルが存在するディレクトリ
- LD\_LIBRARY\_PATH: Teradata ライブラリ
- PATH: Teradata クライアント・ディレクトリ

**注:** errmsg.txt および libcliv2.so ファイルは通常、同じディレクトリにあります。したがって、COPERR および COPLIB 変数の値は通常は同じです。

64 ビット LINUX の例:

```
<variable id="TWB_ROOT" value="/opt/teradata/client/13.10/tbuild"/>
<variable id="TD_ICU_DATA" value="</opt/teradata/client/13.10/tdicu/lib64"/>
<variable id="NLSPATH" value="</opt/teradata/client/13.10/odbc_64/msg/%N >"/>
<variable append=true id="NLSPATH" value="/opt/teradata/client/13.10/tbuild/msg64/%N/
>
<variable id="COPERR" value="/usr/lib64"/>
<variable id="COPLIB" value="/usr/lib64"/>
<variable append=true id="LD_LIBRARY_PATH" value="/opt/teradata/client/13.10/tbuild/
lib64/>
<variable append=true id="LD_LIBRARY_PATH" value="/usr/lib64/>
<variable append=true id="PATH" value="/opt/teradata/client/13.10/tbuild/bin/>
<variable append=true id="PATH" value="/opt/teradata/client/13.10/tbuild/lib64/>
```

7 opmn.xml ファイルを保存します。

8 Oracle Process Manager and Notification Server サービス(EPM\_epmsystem1)を使用して、Windows の「サービス」パネルから Essbase サーバーを停止します。

9 次を検証します:

- Essbase サーバー: 管理サービス・コンソールでデータ準備エディタを使用し、DNS を使用して Teradata データベースに接続します。
- Oracle Essbase Studio: 非ストリーミング・モードでキューブ配置を実行します。Teradata ODBC ドライバが使用されます。

## Teradata Parallel Transporter を使用するための環境の設定

第2章「データ・ソースの構成」の手順に従って、次のタスクを実行します:

- Teradata データベースのホスト・ファイルにエントリを追加します; 例:

```
172.27.24.181  tera2db  tera2cop1
```

- \$TELAPI\$<tera>用にシステム ODBC DSN を構成します。ここで、<tera>は Teradata データ・ソースの名前です: 例:

```
DSN = $TELAPI$tera2db
```

- UNIX オペレーティング・システムでは、必要な環境変数パスが、適切な場所で定義されていることを確認します(Windows インストールでは、必要な環境変数が自動的に更新されます):

- TD ODBC ドライバ
- CLIV2
- TD GSS
- 共有 ICU
- TPT Export Operator ファイル
- DataDirect ODBC ドライバ

- さらに、オペレーティング・システムの適切なパスに、Teradata Parallel Transporter 用に次の変数を設定します。(詳細は、『Teradata Parallel Transporter Application Programming Interface Programmer Guide』の付録「Code Samples」を参照してください)。

- export LD\_LIBRARY\_PATH = <ライブラリ・パス>:\$LD\_LIBRARY\_PATH

```
export LD_LIBRARY_PATH = /usr/tbuild/12.00.00/lib:$LD_LIBRARY_PATH
```

- export NLSPATH = <カタログのディレクトリ・パス>/%N:\$NLSPATH

```
export NLSPATH = /usr/tbuild/12.00.00/msg/%N:$NLSPATH
```

- (CLI がデフォルト・ディレクトリにインストールされていない場合) export COPERR = <errmsg.catのディレクトリの場所>

```
export COPERR = /usr/lib
```

## Teradata Parallel Transporter を使用した Teradata データのロード

第4章「SQL データのロード」の手順に従います。SQL データ・ソースを開くときに、ODBC DSN として定義する接頭辞\$TELAPI\$が付いている必要なデータ・ソース名を選択します。SQL 文の場合、SQL SELECT、FROM、および WHERE ステートメントでネイティブ Teradata クエリーを定義します。これらのステートメントに、改行を含めないでください。各エントリは、1つのステートメントに入っ

ている必要があります。ネイティブ Teradata SQL クエリー規則については、関連する Teradata ドキュメントを参照してください。

## Teradata TPT-API ロード設定のカスタマイズ

データ・ロードに Teradata TPT-API を使用する場合、TPT-API を使用してデータをロードすると同時に柔軟性を大幅に向上する設定をカスタマイズできます。次の設定は、`essbase.cfg` に含まれている場合、Teradata TPT-API データ・ロード・オプションをカスタマイズできます。

- `TD_MAX_SESSIONS`
- `TD_TRACE_OUTPUT`
- `TD_TRACE_LEVEL`
- `TD_TENACITY_HOURS`
- `TD_TENACITY_SLEEP`

### TD\_MAX\_SESSIONS

ログオンできる Teradata データ・ロード・セッションの最大数を指定します。

#### 構文

```
TD_MAX_SESSIONS  
n
```

`n` は、ログオンできるデータ・ロード・セッションの最大数を指定する整数です。値: 0-255。ゼロはデータ・ロード・セッションを終了します。デフォルト値は、4 です。

### TD\_TRACE\_OUTPUT

Teradata トレース・メッセージの場所を設定します。

#### 構文

```
TD_TRACE_OUTPUT  
tracefile
```

`tracefile` は、トレース・メッセージのファイル名またはファイル名へのパスです。デフォルト値は"`Essbase_TPT_Trace.txt`"です。このファイルは `ORACLE_INSTANCE` の場所(`/scratch/aime/Oracle/Middleware/user projects/epmsystem1/`など)にあります。



## TD\_TRACE\_LEVEL

Teradata ドライバのトレース・レベルを設定します。デフォルトは TD\_OFF です。

### 構文

```
TD_TRACE_LEVEL  
trace_constant
```

trace\_constant は次の定数のいずれになります:

- "TD\_OPER" - ドライバ固有のアクティビティのトレースを有効にします
- "TD\_OFF" - トレースは無効です
- "TD\_OPER\_ALL" - すべてのドライバレベルのトレースを有効にします
- "TD\_OPER\_CLI" - CLIV2 を含むアクティビティのトレースを有効にします
- "TD\_OPER\_NOTIFY" - 通知機能を含むアクティビティのトレースを有効にします
- "TD\_OPER\_OPCOMMON" - 演算子を含むアクティビティのトレースを有効にします
- NULL - 引数リストを終了します

## TD\_TENACITY\_HOURS

許可されている操作の最大数が Teradata データベースですでに実行されている場合に、Teradata ロード・ドライバがログオンの再試行を続行する時間数を設定します。

### 構文

```
TD_TENACITY_HOURS  
n
```

n は、-n から n までの正または負の整数になります。デフォルト値は、4 です。ゼロ値はこの再試行オプションを無効にします。負の値はデータ・ロードを終了します。

## TD\_TENACITY\_SLEEP

許可されている操作の最大数が Teradata データベースですでに実行されている場合に、Teradata ロード・ドライバがログオンの再試行を続行する分数を設定します。

### 構文

```
TD_TENACITY_SLEEP
```

n はゼロまたは正の整数になります。デフォルト値は 6 です。

## Unicode 文字セットとマルチバイト文字セットのサポート

Teradata は、Essbase サーバーが TPTapi を使用して取得するマルチバイト文字セット(MBCS)と Unicode テキストをサポートします。

この機能を使用するには、次のタスクを実行します:

- Essbase サーバーで使用されるクライアント文字セットが、Teradata データベースにインストールされていること、または Teradata データベースで使用可能になっていることを確認します。
- ODBC ドライバの文字セットが、Essbase サーバーが TPTapi に渡す文字セットに一致することを確認します。

そのためには、表 2 で示すように、`$ESSLANG` 変数によって使用される文字セットに一致する文字セット名で ODBC 接続 DSN を作成する必要があります。

表 2 サポートされている文字セット

文字セット	<code>\$ESSLANG</code> 変数の設定
ラテン(ほぼすべての西洋の言語に適用)	各種
日本語	KANJISJIS_05
Unicode	UTF8

**注:** Essbase は、サポートされている文字セットでデータを取得します;ただし、SQL クエリーは英語である必要があります。

# 索引

## 記号

.odbc.ini ファイル、DB2 の例, 10

## A - Z

Application キーワード, 23

bin ディレクトリ, 24

CFG (.cfg)ファイル、作成, 24

ConvertUTF16toUTF8 キーワード, 24

Database キーワード, 23

DataDirect ドライバ

デフォルト設定の変更, 21

情報, 9, 10

DB2、.odbc.ini ファイルの例, 10

Description キーワード, 22

Dictionary キーワード, 23

DriverName キーワード, 22

Essbase

アウトライン、データのマッピング先, 16

データのロード, 15

ドライバのデフォルト値, 23, 25

ESSBASEPATH, 24

Essbase アウトラインへの SQL データのマッピング, 16

Essbase 次元の構築, 7

esssql.cfg

キーワードと値, 23

作成, 21

例, 22

非 DataDirect ドライバの構成, 24

Files キーワード, 23

IsQEDriver キーワード, 23

Microsoft Access、ODBC ドライバ, 24

ODBC Administrator, 9

ODBCJT32.DLL, 24

ODBC ドライバ

UNIX、名前, 24

テスト済, 21

ファイル名, 24

非 DataDirect, 21

Password キーワード, 23

SELECT ステートメント

テーブルの結合, 13

処理時間, 13

Server キーワード, 23

SingleConnection キーワード, 23

SQL クエリー, 17

SQL サーバー, 13

SQL データ

ソースの選択, 17

テーブルの結合, 13

マッピング, 16

ロード, 7

SQL データベース、1 つのテーブル, 13

SQL データ・ソースの選択, 17

SQL レコードの解釈, 7

SQL 以外のデータの要求, 7

SQL 文, 7

SQL 文の変換, 7

TD\_MAX\_SESSIONS, 32

TD\_TENACITY\_HOURS, 32

TD\_TENACITY\_SLEEP, 32

TD\_TRACE\_LEVEL, 32

TD\_TRACE\_OUTPUT, 32

Teradata データ・ソース、Teradata エクスポート演算子の使用, 27

TPT-API, 32

Unicode、サポート, 7, 34

UNIX 環境、odbc.ini ファイル, 24

UpperCaseConnection キーワード, 23

UserId キーワード, 23

Windows 環境、データ・ソースの構成, 9

## あ行

インストール済 ODBC ドライバ、リスト済, 24

インポート・データベース MaxL ステートメント, 18

## か行

キーワード、ドライバ構成ファイル, 23

## さ行

スプレッドシート・データ・ソース, 7

## た行

ダイアログ・ボックス、ODBC Data Source Administrator, 24

テキスト・ファイル・データ・ソース, 7

テスト済の非 DataDirect ドライバ, 21

テーブル

SQL データベース, 13

ロード, 13

作成, 13

効率, 13

結合, 13

テーブルの結合, 13

テーブルの結合、管理サービス・コンソール, 13

ディレクトリ

bin, 24

データ・ソース、デフォルト, 17

ドライバ構成ファイル, 24

データ

SQL、マッピング, 16

ロードの準備, 13

ロード準備方法, 13

データのロード

1つのテーブルから, 13

テーブルの結合, 13

データ・ロードのルール, 16

並列, 18

概要, 8, 15

準備, 13

管理サービスの使用, 15

要約レベル・データ, 7

集約ストレージ・データベース, 18

データ・ソース

パス, 17

選択, 17

データ・ロード・ルール・ファイル。「ルール・ファイル。」を参照

ドライバ

ODBC Administrator での選択, 9

UNIX、名前, 24

パスワード, 24

ファイル名, 24

命名, 22

説明, 22

非 DataDirect, 21

ドライバのパスワード, 24

ドライバ構成ファイル

bin ディレクトリ, 24

キーワード, 23

デフォルト値, 23, 25

作成, 21, 23, 24

必要な値, 21

## は行

パス

esssql.cfg, 24

データ・ソース, 17

ビュー, 13

SQL データベース, 13

作成, 13

ファイル、CFG(構成), 24

## ま行

マルチバイト文字セット、サポート, 34

## ら行

リレーショナル・データ、ロード, 7。「データ。」も参照

ルール・ファイル

SQL レコード, 7

ファイル名の最大サイズ, 18

並列データのロード, 18

作成, 16

場所, 18

定義, 16

集約ストレージ・データベース, 18

ルール・ファイル、複数、使用, 18

## わ行

並列 SQL 接続, 18

作成

1つのテーブルまたはビュー, 13

データ・ロード・ルール・ファイル, 16

- ドライバ構成ファイル, [21](#), [23](#), [24](#)
- 例、.odbc.ini ファイル, [10](#)
- 処理時間, [13](#)
- 多次元データベース
  - Essbase 次元の構築, [7](#)
  - データのロード, [7](#)
- 構成
  - データ・ソース
    - 概要, [9](#)
  - 非 DataDirect ドライバ, [24](#)
- 構成ファイル、非 DataDirect ドライバ, [21](#)
- 次元、構築, [7](#)
- 渡す、SQL 文, [7](#)
- 管理サービス・コンソール
  - SELECT ステートメント, [13](#)
  - SQL データのロード, [15](#)
  - テーブルの結合, [13](#)
- 複数のテーブル
  - ロード, [13](#)
  - 準備, [13](#)
  - 結合, [13](#)
- 複数のルール・ファイル、使用, [18](#)
- 複数ルール・ファイルの使用, [18](#)
- 要約データ、ロード, [7](#)
- 選択、ドライバ, [9](#)
- 集約ストレージ・データベース、SQL データのロード先, [18](#)
- 集約ストレージ・データ・ロード・バッファ, [18](#)
- 非 DataDirect ドライバ
  - テスト済, [21](#)
  - 構成, [21](#)

