

# Oracle DBA & Developer Days 2011

日本オラクル、今年最大の技術トレーニングイベント

2011年11月9日(水)～11月11日(金) シェラトン都ホテル東京



## ORACLE®

### DBアプリ開発Tips .NET + Oracle Database 次世代アプリケーション開発手法を徹底解説

日本オラクル株式会社 製品事業統括 - 製品戦略統括本部 - ISV/OEM営業本部  
プリンシパルエンジニア 大田 浩

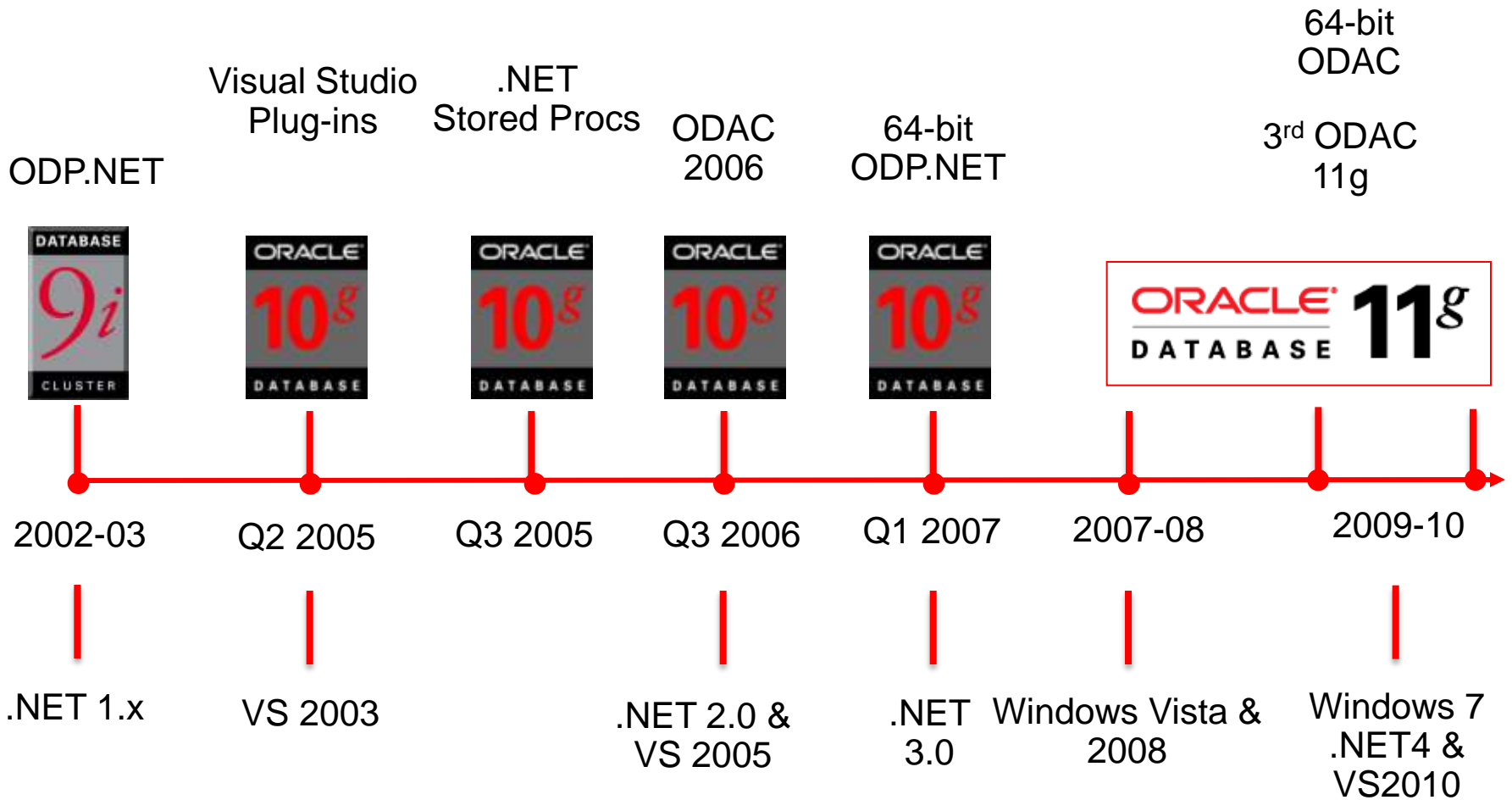
以下の事項は、弊社の一般的な製品の方向性に関する概要を説明するものです。また、情報提供を唯一の目的とするものであり、いかなる契約にも組み込むことはできません。以下の事項は、マテリアルやコード、機能を提供することをコミットメント(確約)するものではないため、購買決定を行う際の判断材料になさらないで下さい。オラクル製品に関して記載されている機能の開発、リリースおよび時期については、弊社の裁量により決定されます。

OracleとJavaは、Oracle Corporation 及びその子会社、関連会社の米国及びその他の国における登録商標です。文中の社名、商品名等は各社の商標または登録商標である場合があります。

# Agenda

- **.NETとOracle Databaseについて**
- Oracle Data Provider for .NET(ODP.NET)の新機能
- 最新のデータアクセス手法について
- 64bitに対応したODP.NETの利用

# 最新の.NET環境にいち早く対応



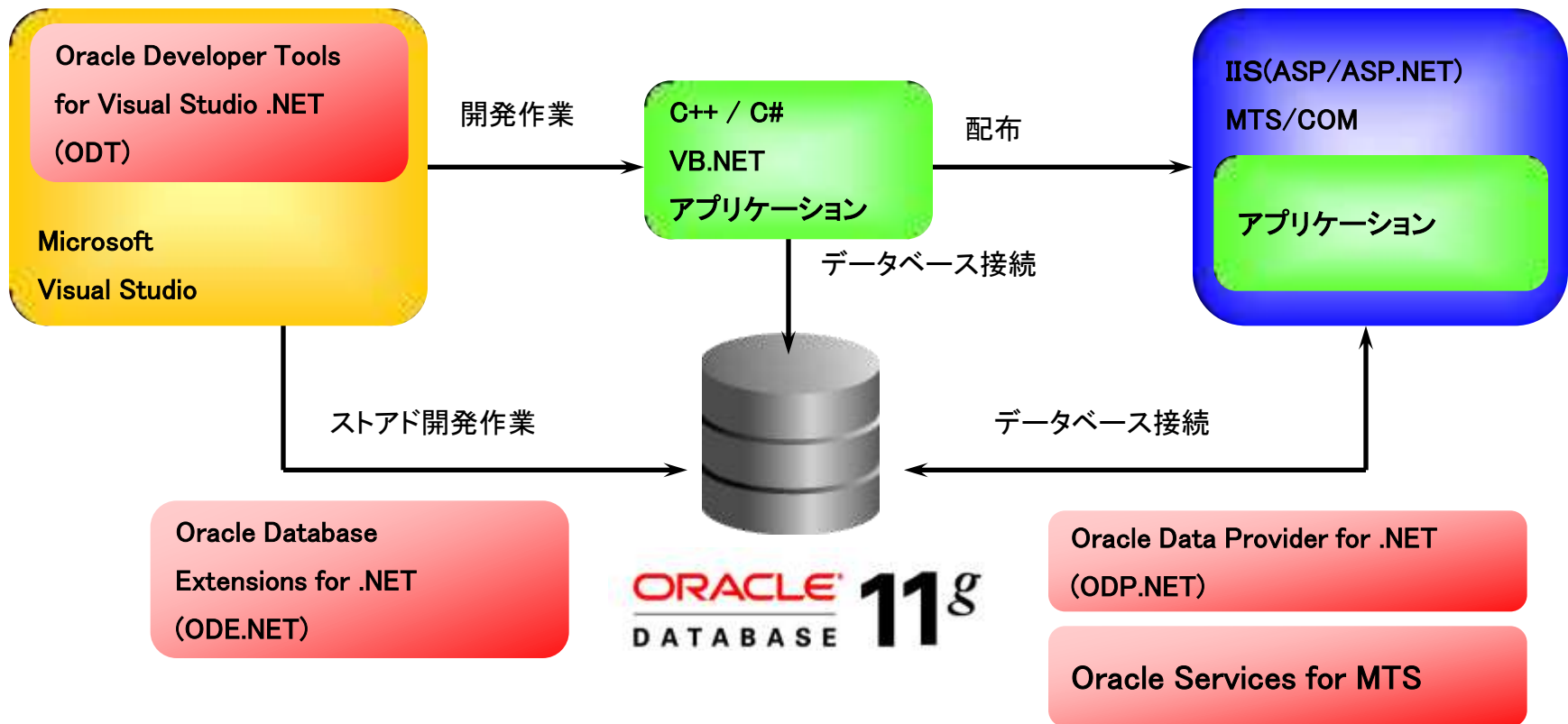
# .NET + Oracleでしか出来ない様々な機能

- 開発生産性を高める様々な機能

機能	Oracle	他社DB
パフォーマンスチューニングオプション	Yes	No
データキャッシュの自動更新	Yes	No
中間層のデータキャッシュ機能	Yes	No
クラスタ構成の負荷分散・フェイルオーバー	Yes	No
さまざまなデータタイプへの対応	Yes	No
高度なセキュリティオプション	Yes	No
Visual StudioからDatabaseのチューニング	Yes	No

# 充実した開発ツール

- Visual Studio とも完全統合



Visual Studio 2005, 2008, 2010 および .NET Framework 2.0/3.x/4に対応

ORACLE

# Agenda

- .NETとOracle Databaseについて
- **Oracle Data Provider for .NET(ODP.NET)の新機能**
- 最新のデータアクセス手法について
- 64bitに対応したODP.NETの利用

# .NET + Oracleでしか出来ない一押し機能 — パフォーマンス・チューニング —

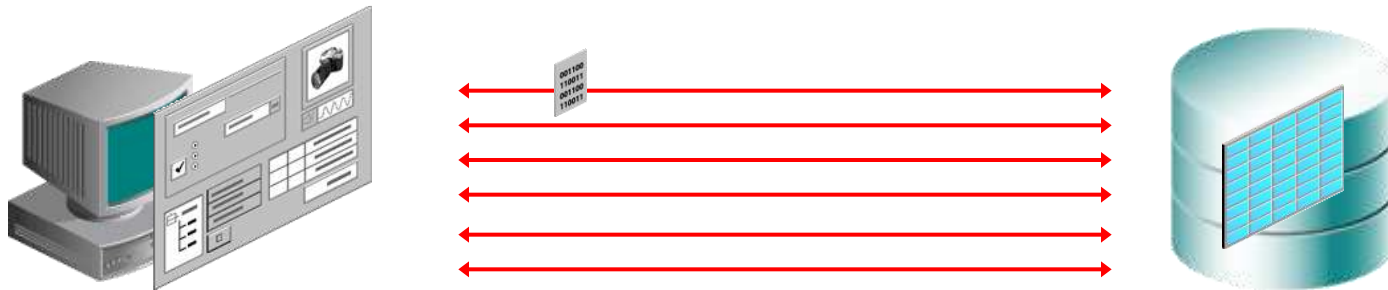
「OTN」「ODP.NET」「パフォーマンス」で検索

「ODP.NETによるデータ・アクセス・パフォーマンスの最適化」に幾つかのサンプルコードが掲載されております。

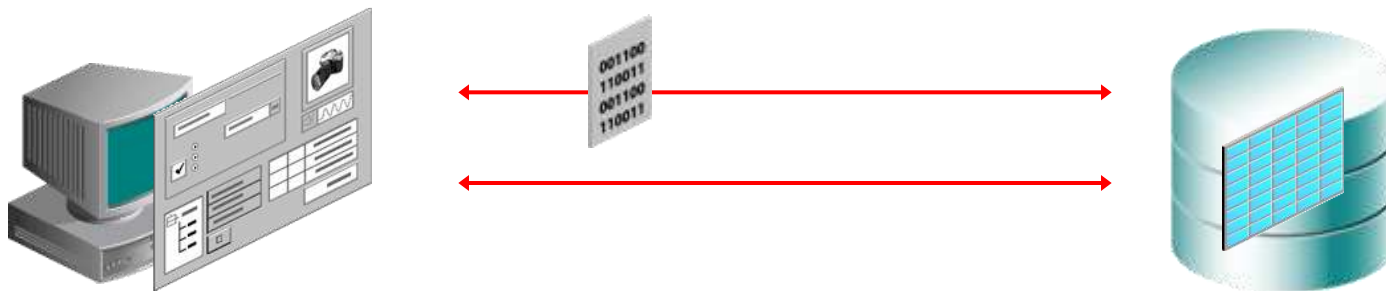


# ODP.NET チューニングポイント 1 ～ ラウンドトリップの回数を減らす ～

取得レコードが多い場合、フェッチサイズによっては多くのラウンドトリップが発生



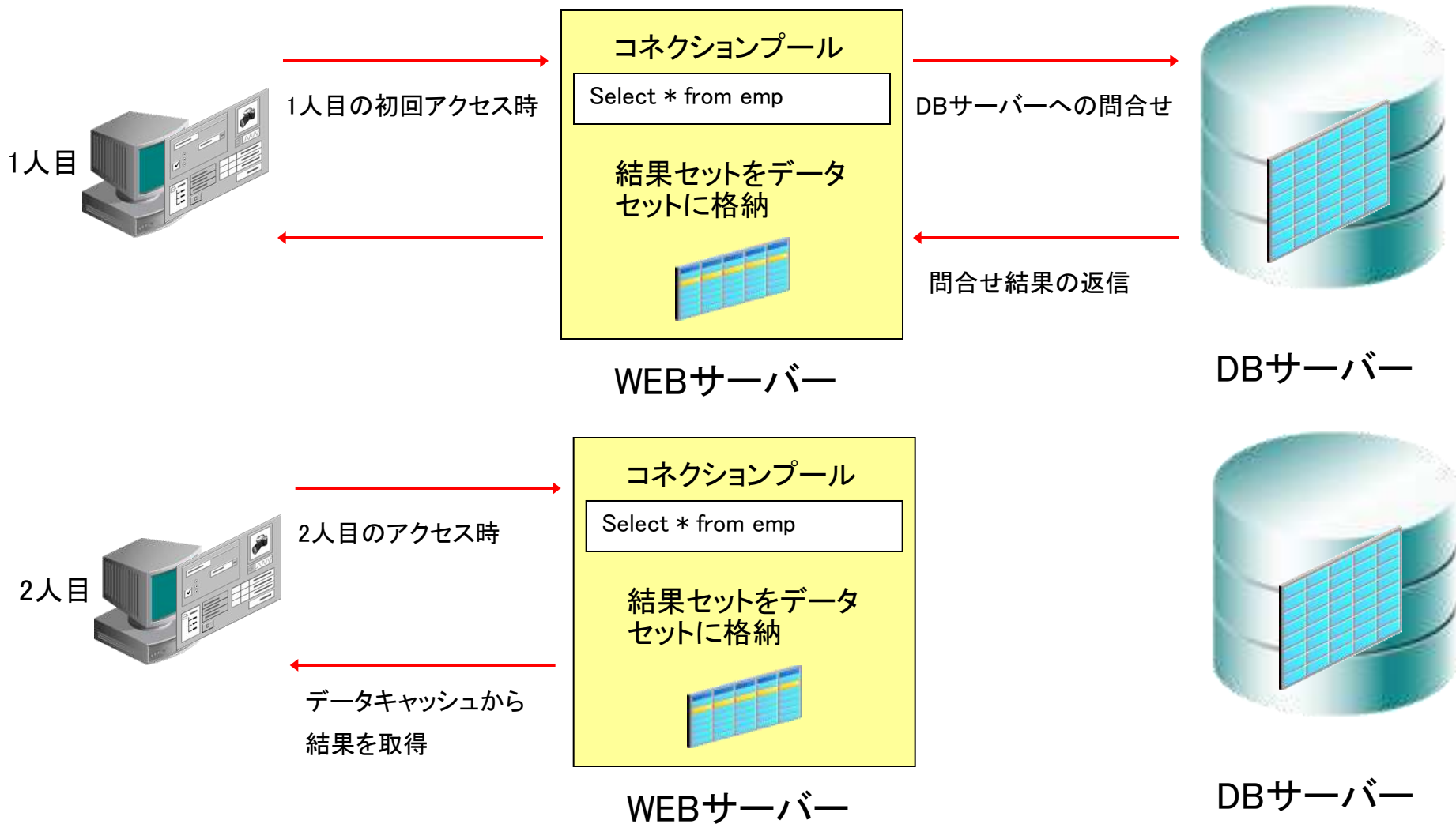
データ取得の際の packet サイズを変更することにより、ラウンドトリップ回数を減少



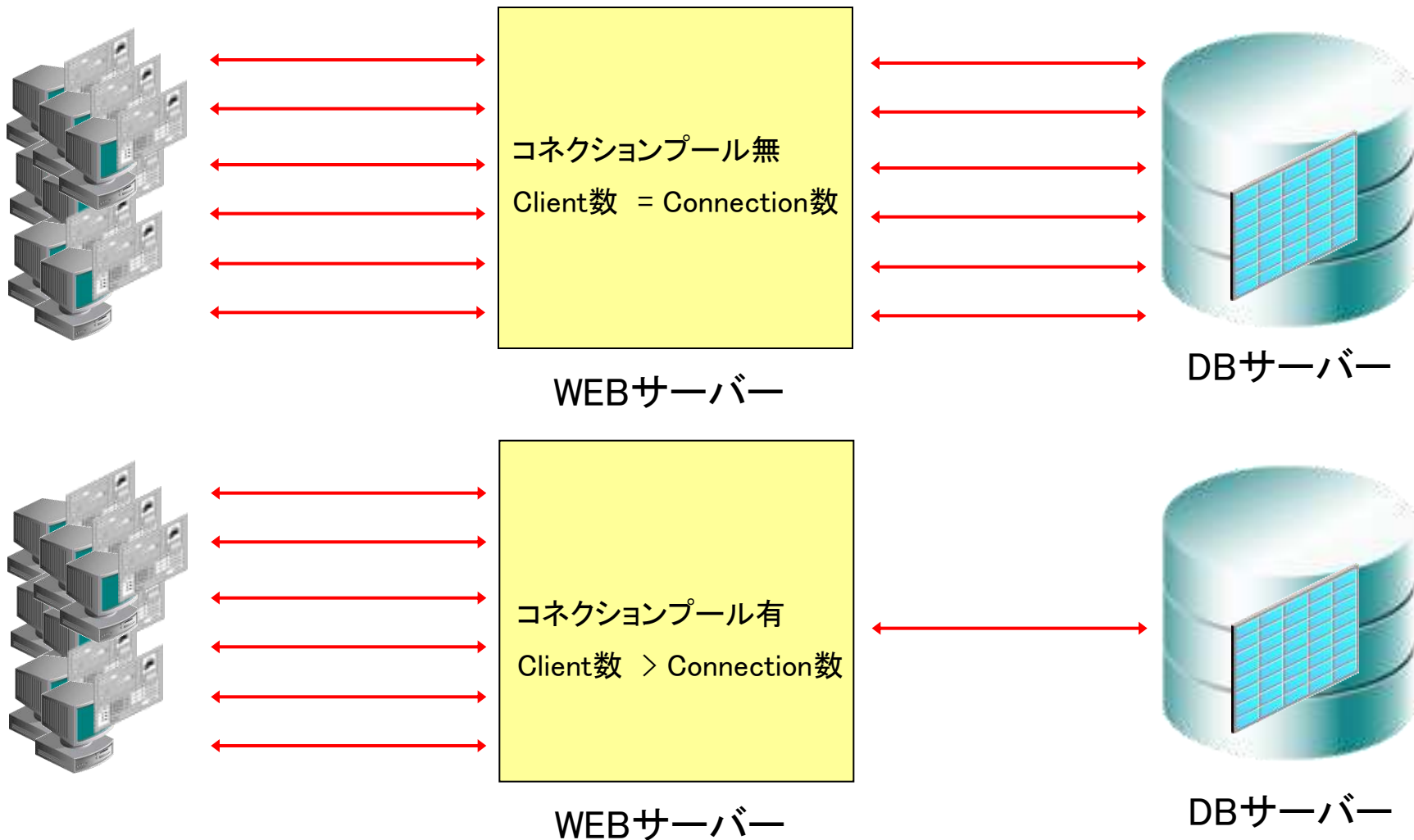
FetchSize・・・デフォルト値は、65,536バイト

# ODP.NET チューニングポイント 2

## ～ キャッシュの利用 ～



# ODP.NET チューニングポイント 3 ～ コネクション・プーリング ～



# ODP.NET

— ラウンドトリップの回数を減らす —

# ODP.NET ラウンドトリップの回数を減らす ～ フェッチ・サイズの制御 ～

エンドユーザーが大量のデータを使用する場合、アプリケーションによるデータ・フェッチのラウンドトリップ回数を最小限にするための仕組み。

```
OracleCommand cmd = con.CreateCommand();
cmd.CommandText = "select * from fetch_test ";
OracleDataReader dr = cmd.ExecuteReader();
dr.FetchSize = cmd.RowSize * numRows;

while (dr.Read())
{
    rowsFetched++;
}
```

FetchSizeの値はデフォルトで、「64KB」に設定

# ODP.NET ラウンドトリップの回数を減らす ～ フェッチ・サイズ / BLOB ～

BLOBでのラウンドトリップ回数を最小限にするための仕組み。

```
Dim cnn As New OracleConnection( _  
    "user id=scott;password=tiger;data source=orcl")  
Dim strSql As String = _  
    "select blob_fld from blob_test where blob_id=1"  
Dim cmd As New OracleCommand(strSql, cnn)  
cmd.InitialLOBFetchSize = 32767  
  
cnn.Open()  
Dim dr As OracleDataReader = cmd.ExecuteReader  
If dr.Read Then  
    Dim blob As OracleBlob = dr.GetOracleBlob(0)  
    Dim ms As New System.IO.MemoryStream(blob.Value)  
    PictureBox1.Image = New Bitmap(ms)  
End If
```

Oracle Data Provider for .NET 11.1から小さなサイズのLOB検索のパフォーマンスが向上。この機能拡張を使用できるのは、Oracle 11gリリース1.0以降のデータベース・バージョンで対応

# ODP.NET ラウンドトリップの回数を減らす ～ OracleBulkCopy ～

Oracleデータベース以外のデータソースからOracleデータベース内の表に対してデータを高速に一括ロード

```
' // ExcelからOracle Databaseへデータを一括コピー
Dim eConn As New OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0;" & _
    "Data Source=C:\Temp\SampleData.xls;Extended Properties=""Excel 8.0;HDR=YES;"";")
eConn.Open()
Dim cmdFrom As New OleDbCommand("Select EMPID, EMPNAME From [Sheet1$]")
cmdFrom.Connection = eConn
Dim rdrFrom As OleDbDataReader = cmdFrom.ExecuteReader
Dim oConn As New OracleConnection("User Id=scott;Password=tiger;Data Source=orcl11g")
oConn.Open()
Dim obCopy As New OracleBulkCopy(oConn)
obCopy.DestinationTableName = "emp"
obCopy.ColumnMappings.Add("empid", "empno")
obCopy.ColumnMappings.Add("empname", "job")
obCopy.WriteToServer(rdrFrom)
```

Oracle Data Provider for .NET 11.1からの機能

# ODP.NET ラウンドトリップの回数を減らす ～ Statement Batching ～

- DataSetの値をバッチでアップデート
  - OracleDataAdapter.UpdateBatchSizeを指定
- 複数のSQL Commandを一回のラウンドトリップで実行



# ODP.NET

## — キャッシュの利用 —

# ODP.NET キャッシュの利用

## ～ ステートメント・キャッシュ ～

同じSQL文を(場合によっては異なるパラメータ値を使用して)後で実行する場合、カーソルから解析された情報を再利用して、文を解析せずに実行

```
"User Id=SCOTT;Password=TIGER;Data Source=ORCL;Statement Cache Size=100"
```

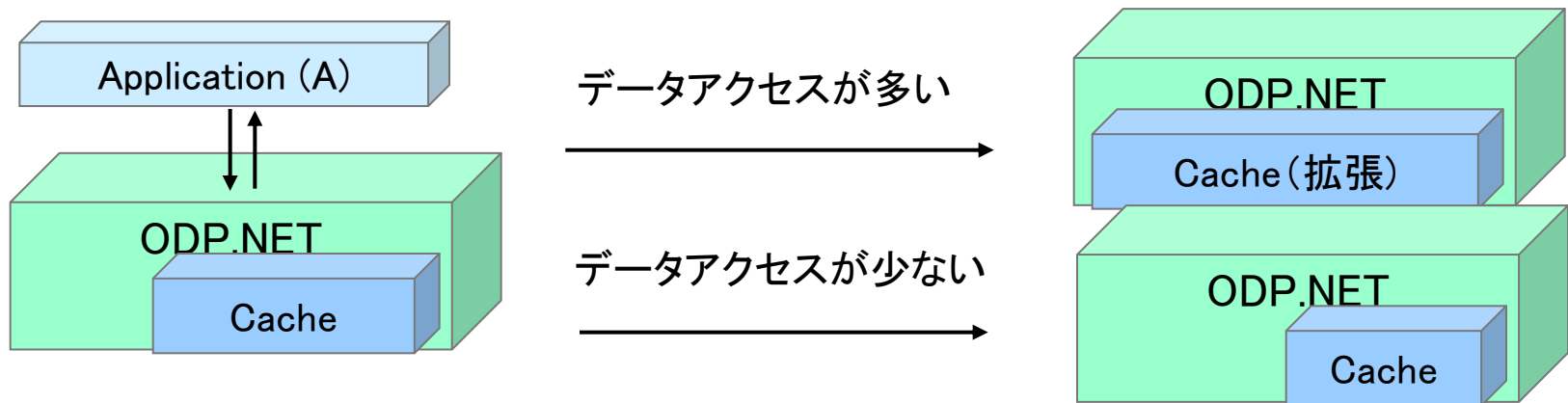
Where句に指定されている条件を次のようにパラメータ変数を使用する必要がある

```
OracleCommand cmd = new OracleCommand(  
    "SELECT * FROM emp WHERE empno=:pEmpNo", conn);  
OracleParameter pEmpNo =  
    cmd.Parameters.Add("pEmpNo", OracleDbType.Int32, ParameterDirection.Input);  
pEmpNo.Value = 7369;
```

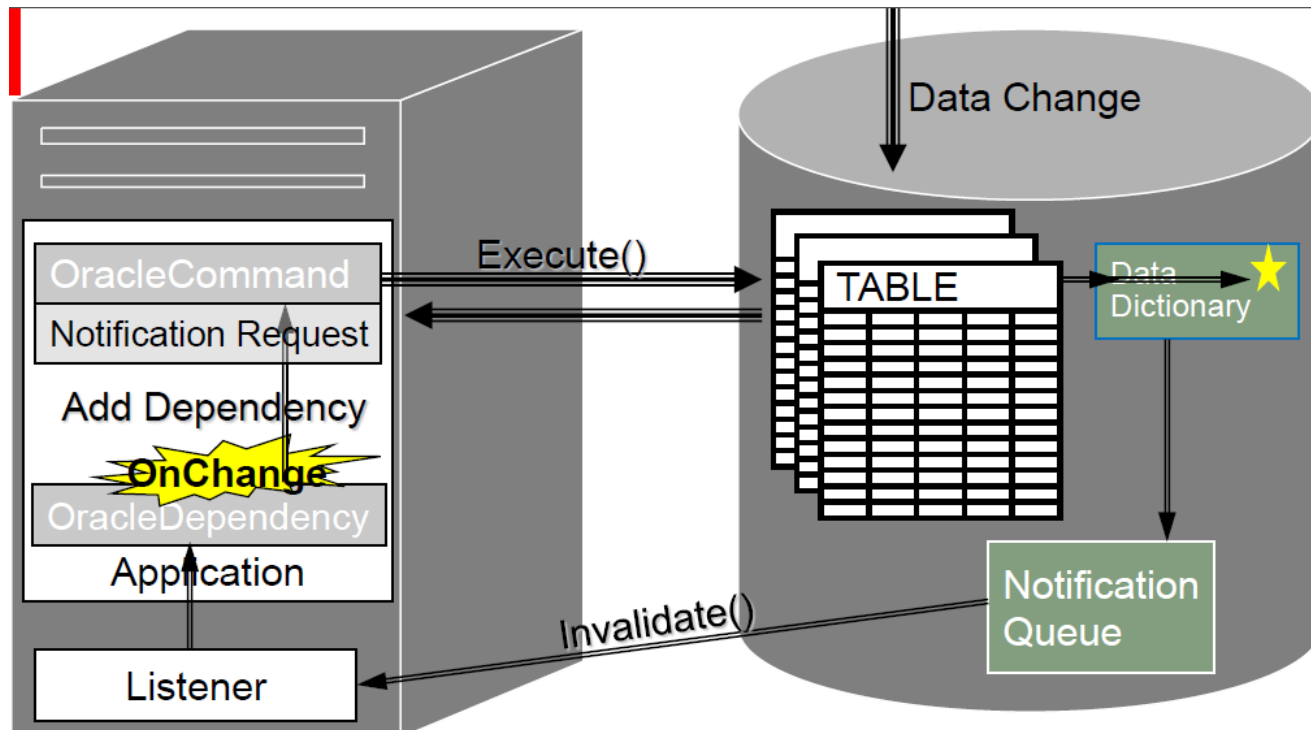
# ODAC 11.1.0.7.20 新機能 - ODP.NET ステートメント・キャッシュの自動チューニング

## 機能

- ODP.NETが自動的に稼動状態のモニタを行い、ステートメント・キャッシュ・サイズを最適な値に設定
- ステートメント・キャッシュによりSQL or PL/SQL のリパースを排除
- 実行頻度に応じてステートメント・キャッシュの大きさを動的に変更



# ODP.NETキャッシュのプログラミング ～ Oracle Continuous Query Notification ～

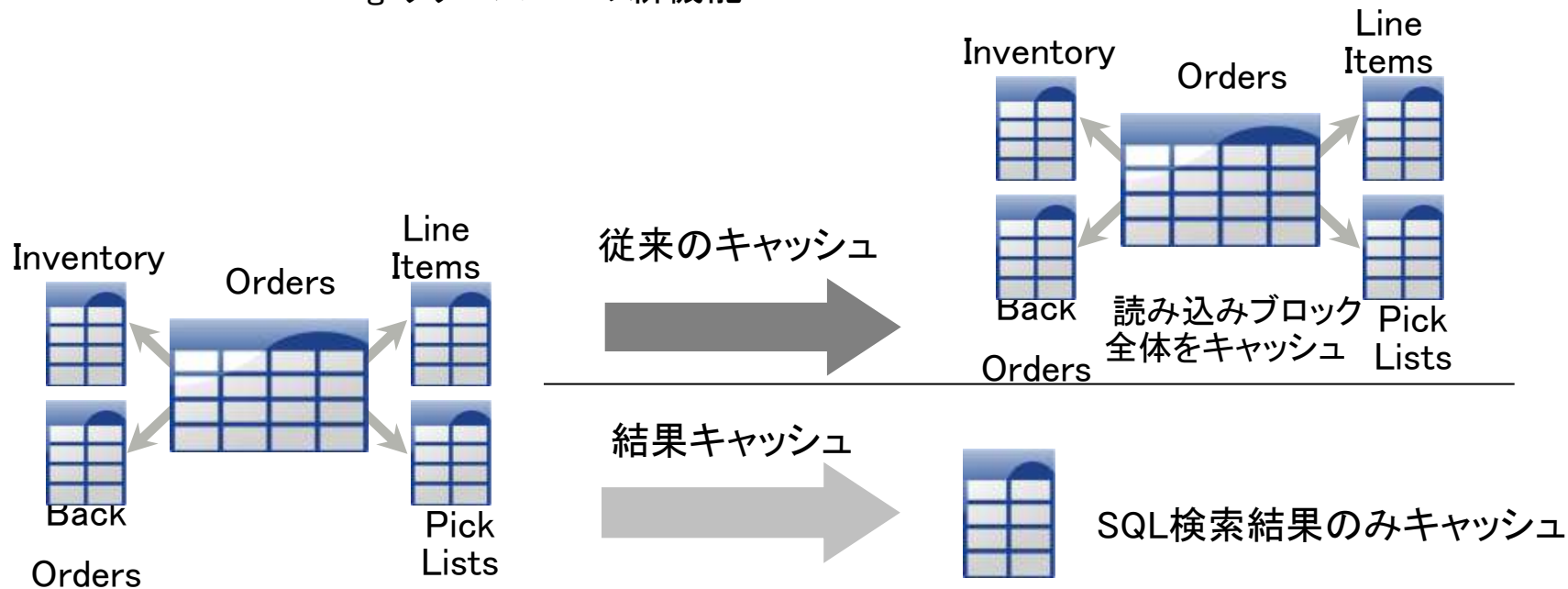


## Client Result Cacheとの比較

- ーキャッシュするデータを細かく絞り込みたい
- ーキャッシュ期間の制御
- ーキャッシュがリフレッシュタイミングで他のロジックを実行

# Oracle Database キャッシュの利用 ～ Oracle Client Result Cache ～

Oracle Database 11g リリース 1.0の新機能



メモリの使用効率をあげ、レスポンスの高速化をはかる  
アプリケーション側の変更は一切不要  
キャッシュの自動更新



# ODP.NET

— コネクション・プーリング —

# Oracle Database コネクションプーリング

ODP.NETの接続プーリング機能を使用すると、アプリケーションは事前に作成された接続を用意するプールを作成および維持可能

- 接続プーリングを使用しない接続文字列 (Pooling=FALSE)  
string no\_pool = "User Id=hr; Password=hr; Data Source=oramag; Enlist=false; Pooling=false"
- 接続プーリングを使用する接続文字列 (Pooling=TRUE)  
string with\_pool = "User Id=hr; Password=hr; Data Source=oramag; Enlist=false; Pooling=true"

Poolingの値はデフォルトで、「TRUE」に設定

# Oracle Database コネクションプーリング ～ OS認証でも利用可能 ～

ODP.NET ONLY

```
Dim cnn As New OracleConnection
```

```
cnn.ConnectionString = "User Id=/;Data Source=orcl;DBA Privilege=SYSDBA" —— ①
```

```
cnn.Open()
```

```
MsgBox("Connect OK!!")
```

```
cnn.Close()
```

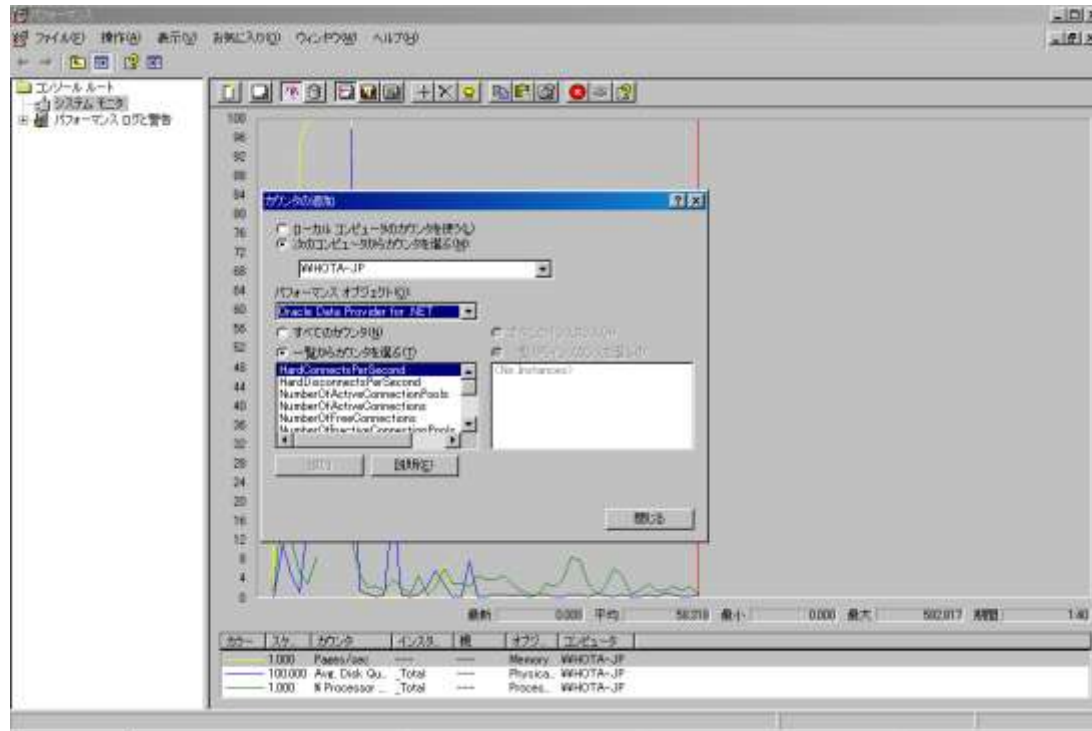
- ①ConnectionString属性のUser Idを / に設定することにより、データベース・ユーザーの認証にWindowsユーザー・ログイン資格証明を使用できます。また、DBA Privilege属性を介してSYSDBA権限またはSYSOPER権限のいずれかを使用してOracleデータベースに接続

ODP.NET 11.1.0以上では、OS認証でも接続プーリングが有効



# Oracle Database コネクションプーリング ～ Perfmonでの確認 ～

Windowsパフォーマンス・モニター(Perfmon)を使用してODP.NETのコネクション・プーリング  
状況等を監視



ODP.NETのパフォーマンス・カウンタはOracle Data Provider for .NETというカテゴリ名で公開

# ODP.NET パフォーマンス – 新機能

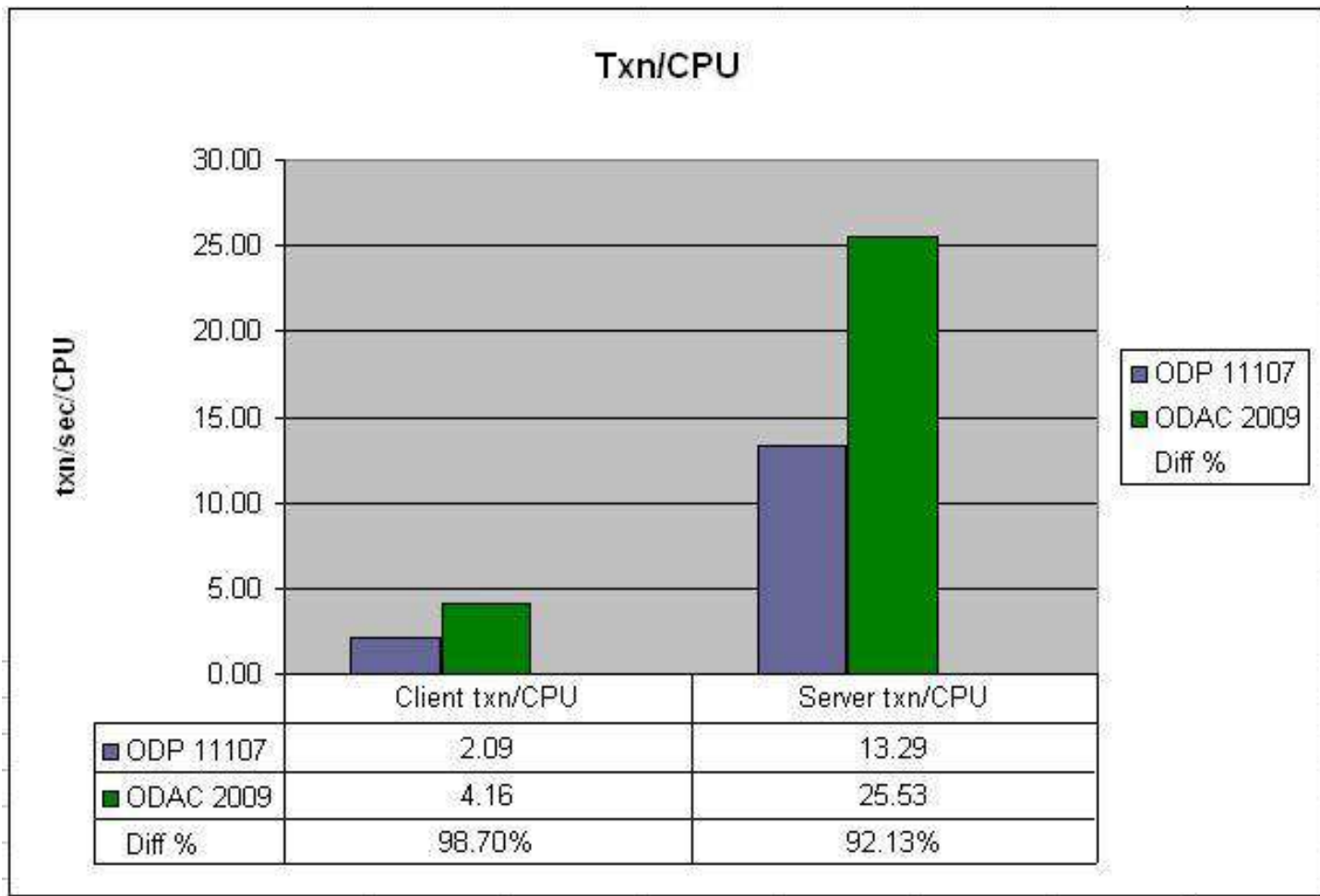
- 自動チューニング
  - 文キャッシュの自動最適化
- データ取得時の高速化
  - DataSet、もしくは OracleDataReader使用時の高速化
- メモリ消費量の削減

上記新機能を利用する際にコード変更は不必要。ODP.NETをアップグレードするのみ

# ODP.NET パフォーマンス – 新機能の効果

- パフォーマンス比較
  - ODP.NET 11.1.7.0 vs ODP.NET 11.1.0.7.20
    - Oracle DB 11.1.0.7を共に使用
  - 80同時ユーザー、1ユーザー当たり4万クエリー実行
  - 60ユニーク・クエリー
  - Datasetにデータを読み込み
- クライアントとサーバーのマシンスペック
  - Quad 3.2GHz intel Xeon 4GM RAM
  - Windows Server 2003 Standard Edition
- チューニングは一切せず。

# 秒間当りのトランザクション数(CPU毎)



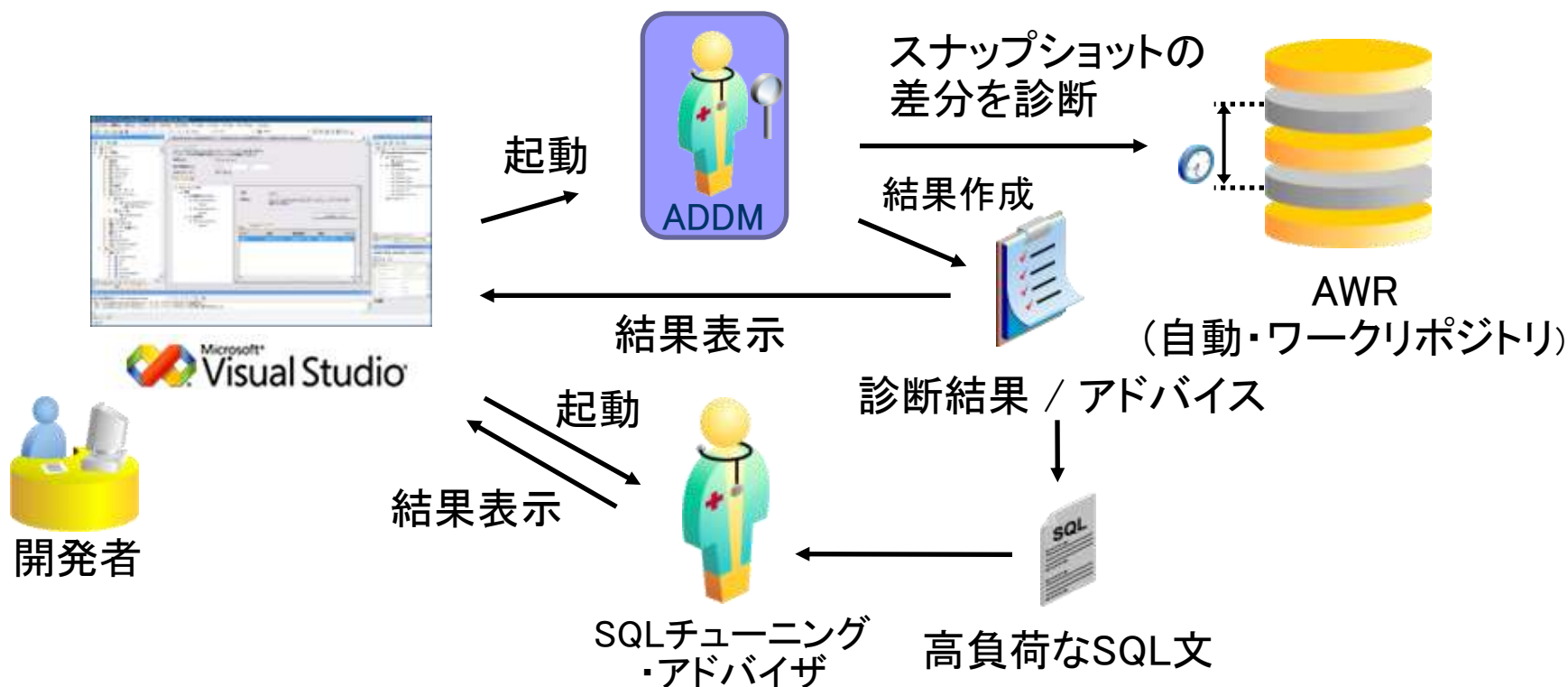
# ODP.NET 自動パフォーマンスチューニング

- ODP.NET 11.0.7.20 vs 11.1.0.7.0 – OLTPでの比較
  - スループットの向上 (60%)
  - レスポンスタイムの短縮 (38%)
  - ネットワークトラフィックの削減 (21%)
  - 低いCPU使用率 クライアント(19%) サーバー(17%)
  - CPU毎の秒間あたりのトランザクション数(TPS)  
クライアント(99%) サーバー(92%)
- コード変更は一切不要

# ODAC 11.1.0.7 新機能 - ODT

## Oracleパフォーマンス・アナライザ

- Visual StudioからAutomatic Database Diagnostic Monitor(ADDM)、SQL Tuning Adviserが使用可能



# Agenda

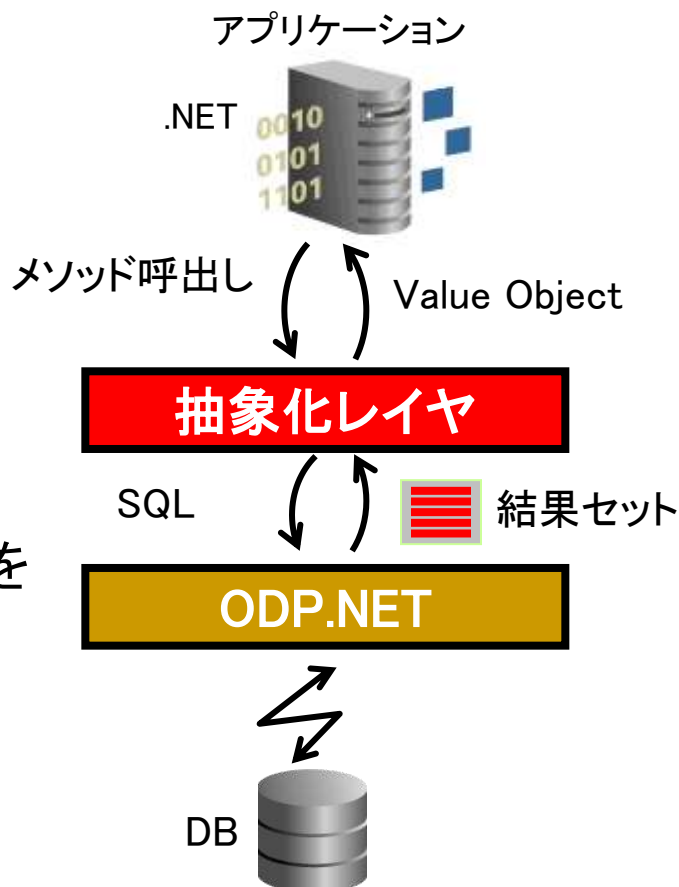
- .NETとOracle Databaseについて
- Oracle Data Provider for .NET(ODP.NET)の新機能
- **最新のデータアクセス手法について**
- 64bitに対応したODP.NETの利用

# O/Rマッピング(ORM) for .NET developers

- O/R マッピングとは
  - 「オブジェクト」と「リレーショナルデータベース(RDB)のレコード」をマッピング

- O/Rマッピングのメリット

- アプリケーション側ではSQLのコードを書かないオブジェクトでデータのやり取り
- オブジェクトへのデータ取得やオブジェクトデータの永続化といった処理を透過的に行える
- データベース操作にかかわる煩雑な作業を軽減し、拡張性・柔軟性を持ったアプリケーションの構築をサポート

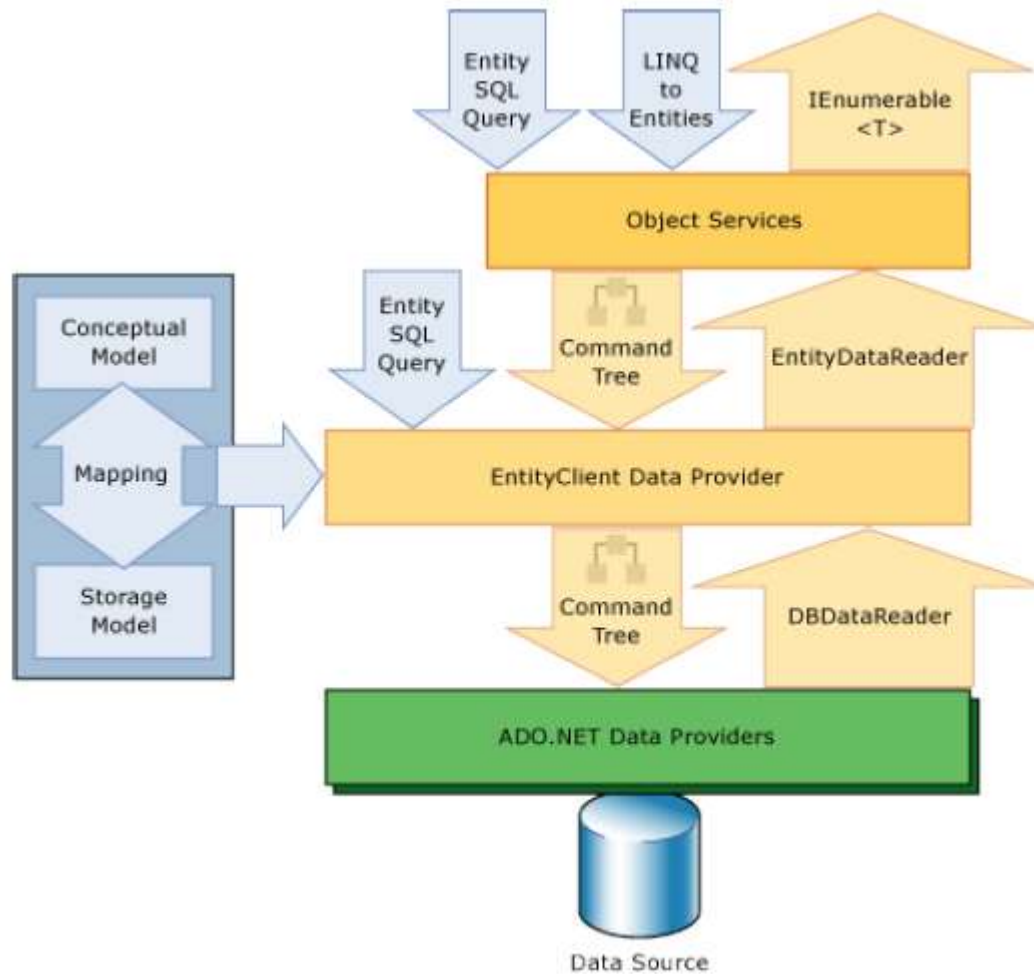




# O/Rマッピング(ORM) for .NET developers

- .NETで使用する主なORMツール
  - **Microsoft Entity Framework**
  - Nhibernate
- **Entity Framework with Oracle Data Provider for .NET**
  - ODAC Entity Framework and LINQ Beta 3 (11.2.0.2.50)
  - 以下のURLよりダウンロード
    - <http://www.oracle.com/technetwork/topics/dotnet/downloads/oracleefbeta-302521.html>
    - インストール方法については以下のURLを参照してください。(英語)
    - <http://www.oracle.com/technetwork/topics/dotnet/downloads/odac-efbetainstallinstructions-302527.html>

# Entity Framework Architecture



# Oracle Support for LINQ and Entity Framework

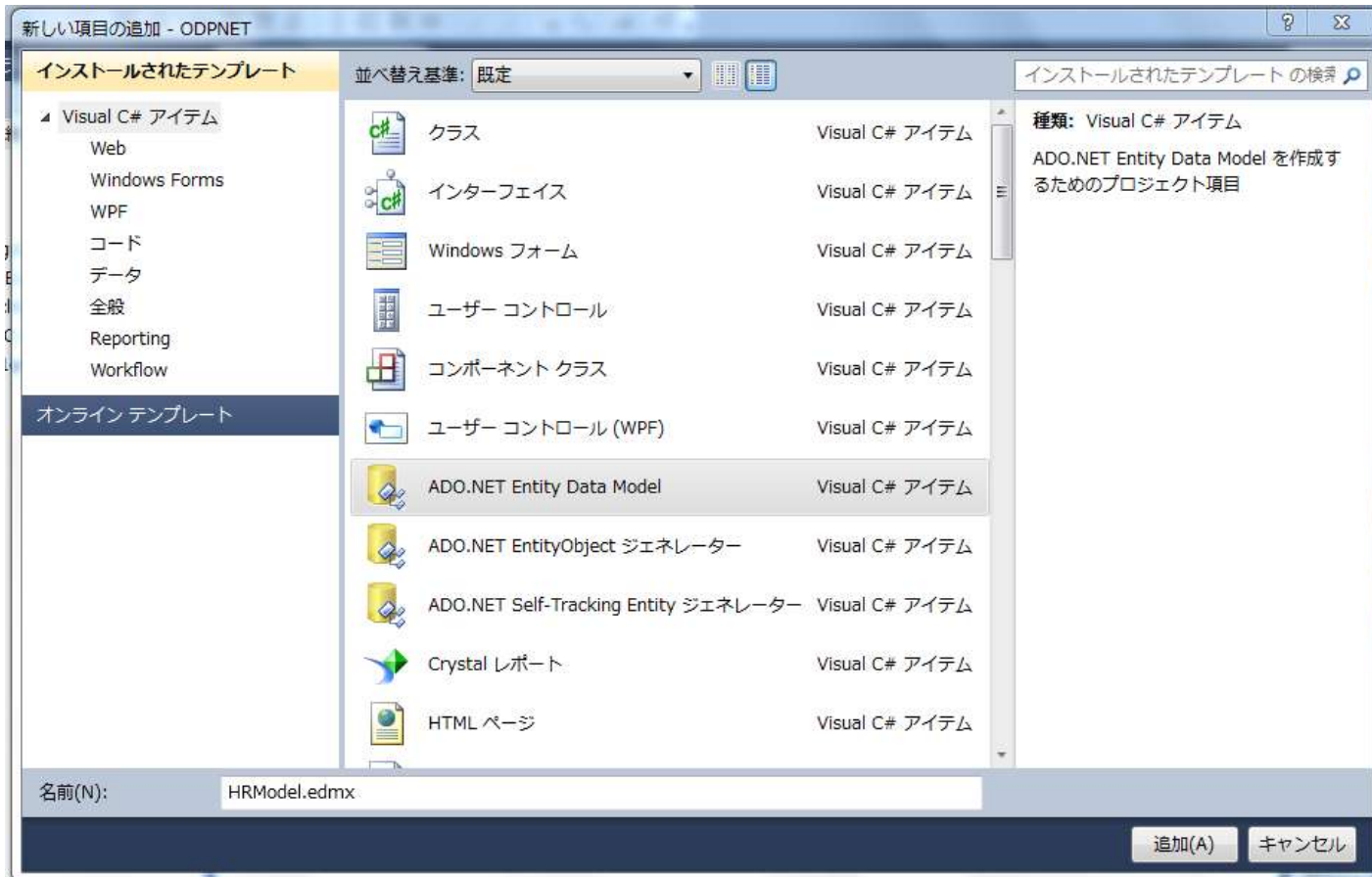
- Entity Data Model Wizard and Designer
  - Oracle Databaseからの概念モデルの生成
  - Oracle Databaseから概念モデル、ストレージモデル、マッピングの更新
  - SQL DDL スクリプトを概念モデルから生成

# Oracle Support for LINQ and Entity Framework

- Query language インターフェース
  - LINQ to Entities
  - Entity SQL
  - DML 対応(insert / update/ delete)
    - Entity FrameworkがOracle DatabaseのDMLを自動生成
    - PL/SQLストアードプロシージャをDMLとして利用することも可能

# Entity Framework with ODP.NET (1)

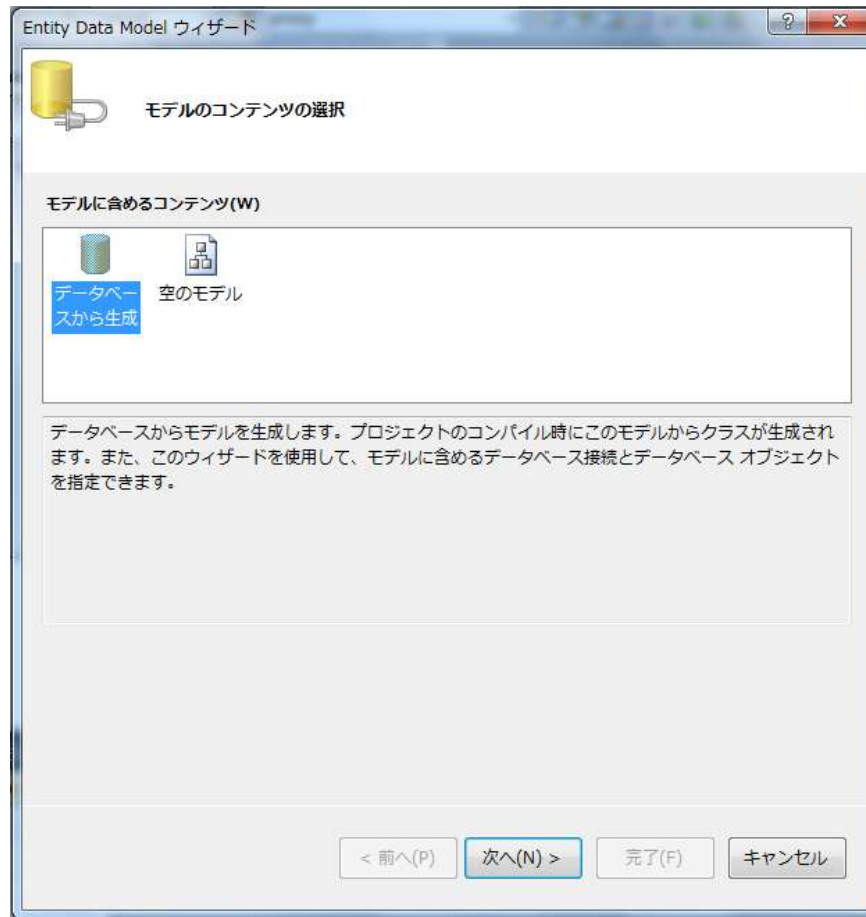
## ADO.NET Entity Data Modelの利用



「新しい項目の追加」→「ADO.NET Entity Data Model」

# Entity Framework with ODP.NET (2)

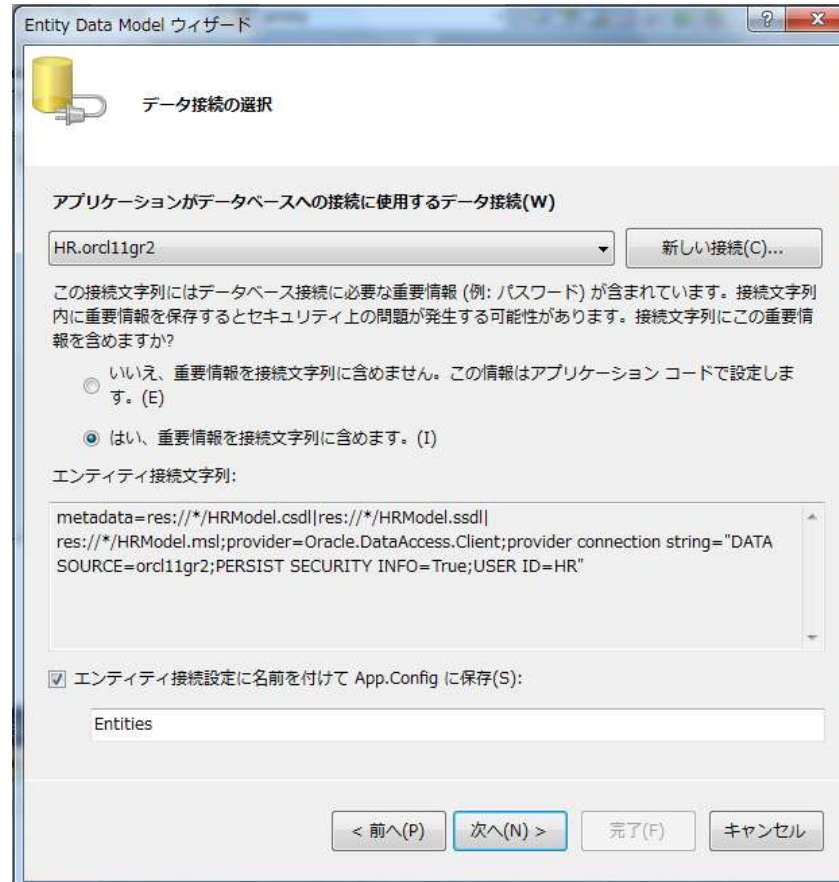
## ADO.NET Entity Data Modelの利用



「モデルに含めるコンテンツの選択」→「データベースから生成」

# Entity Framework with ODP.NET (3)

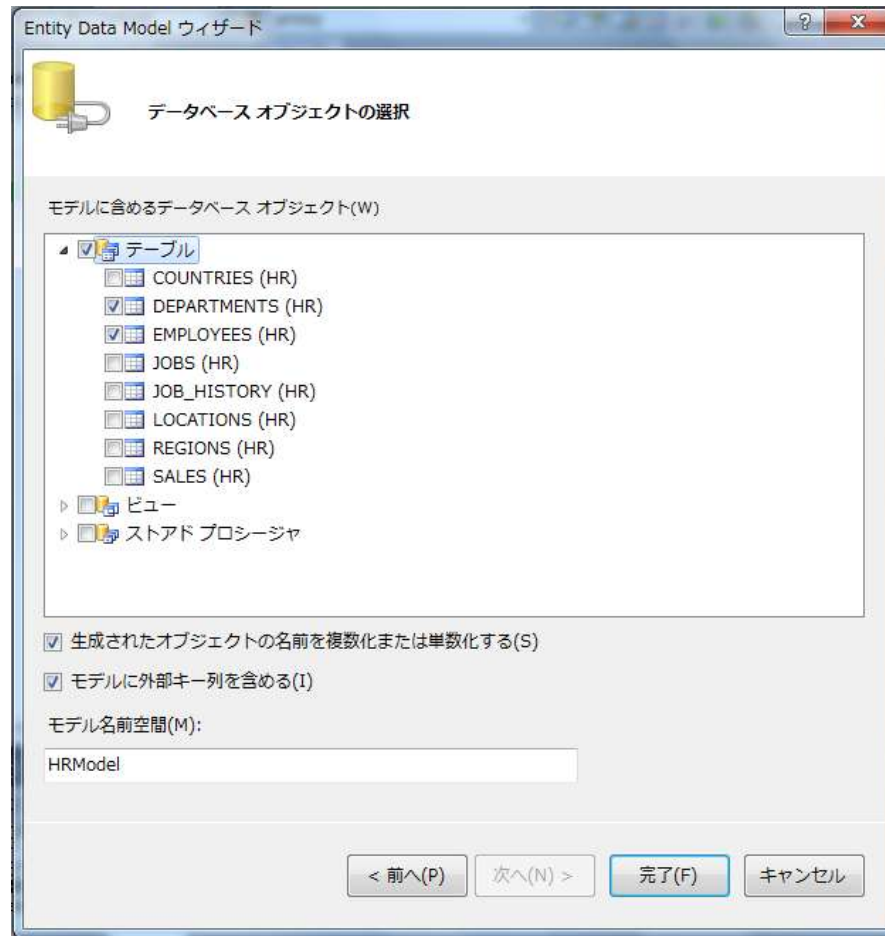
## ADO.NET Entity Data Modelの利用



接続先のデータベースを選択

# Entity Framework with ODP.NET (4)

## ADO.NET Entity Data Modelの利用

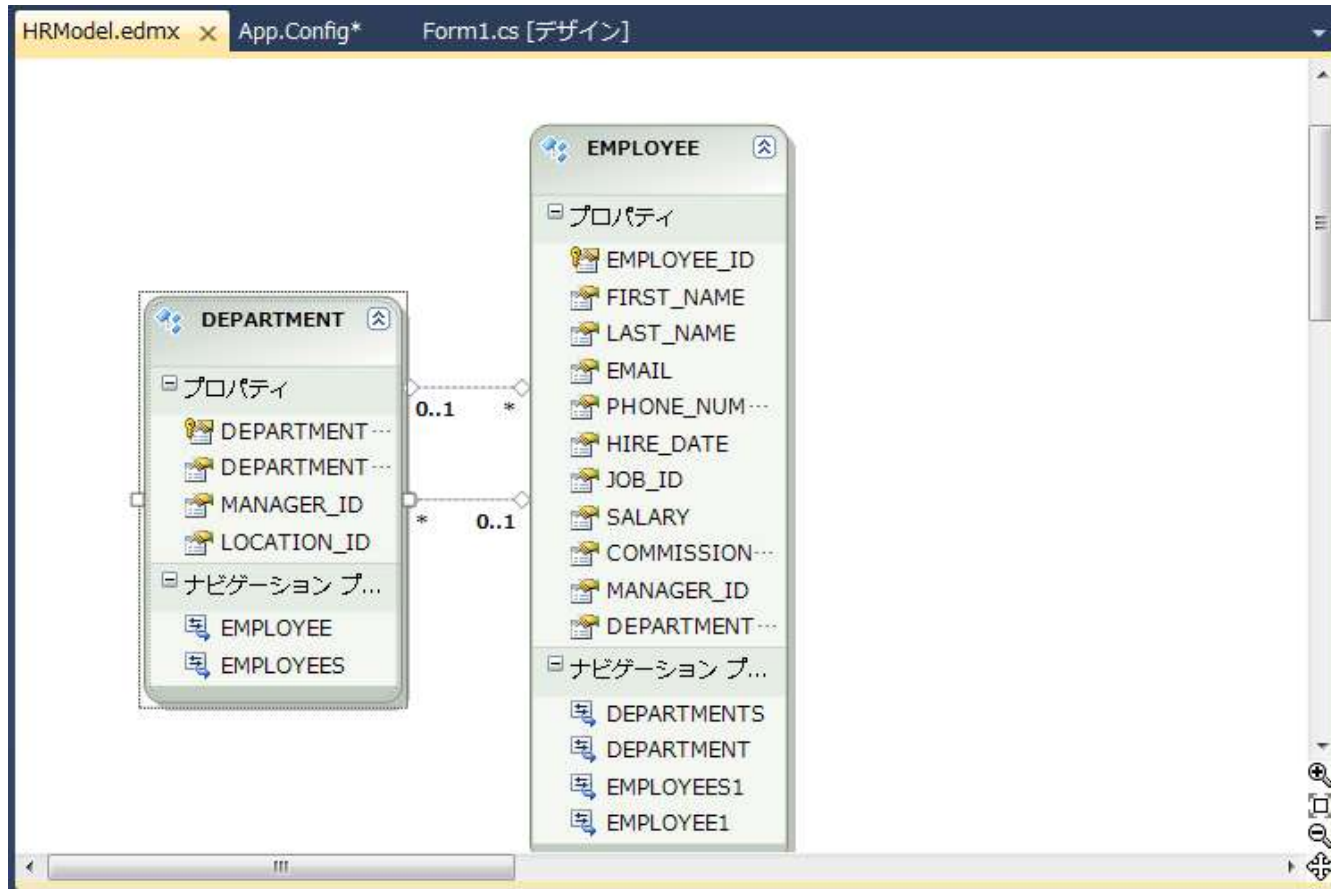


データベース・オブジェクトの選択



# Entity Framework with ODP.NET (5)

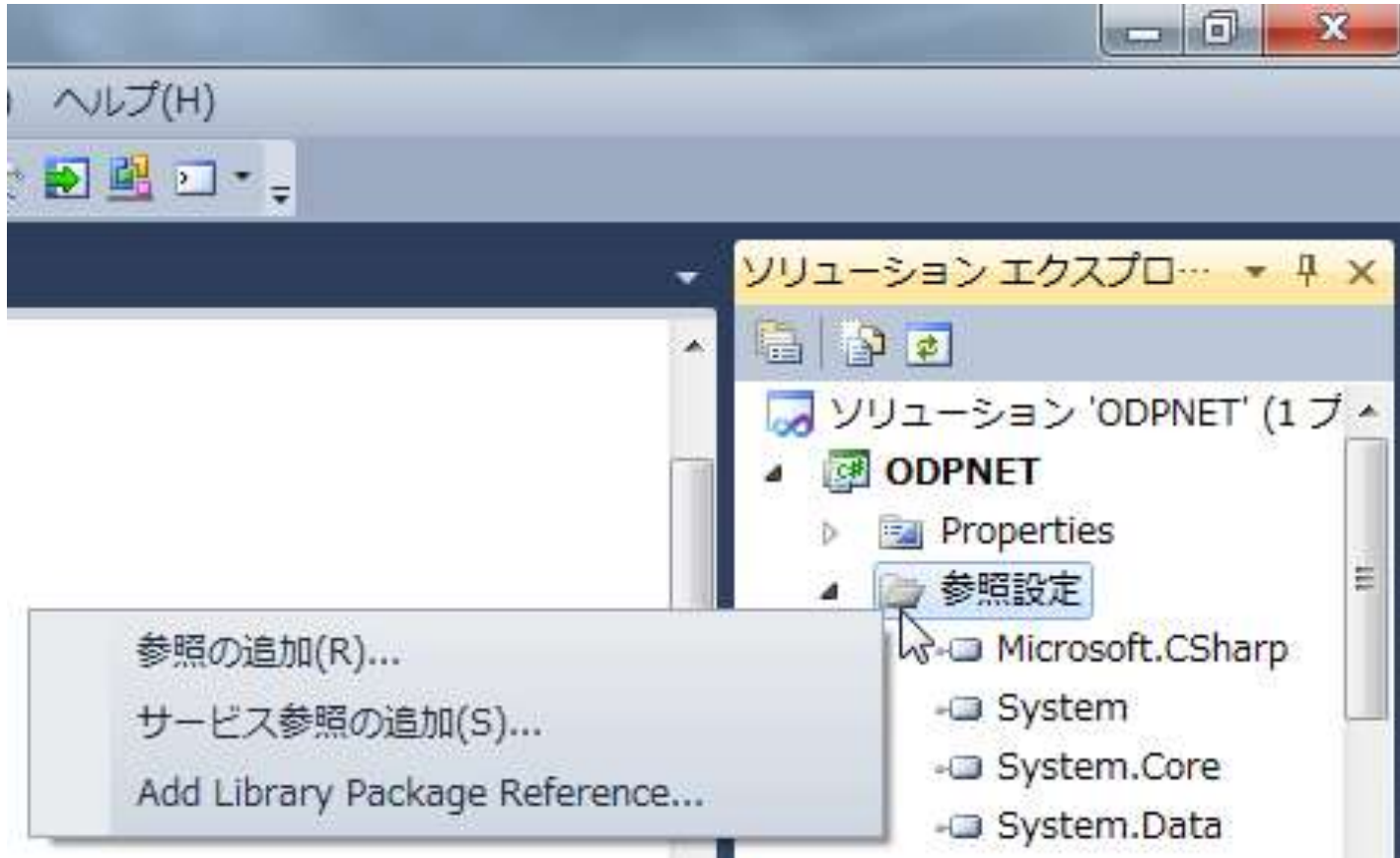
## ADO.NET Entity Data Modelの利用



データベース・オブジェクトの確認

# Entity Framework with ODP.NET (6)

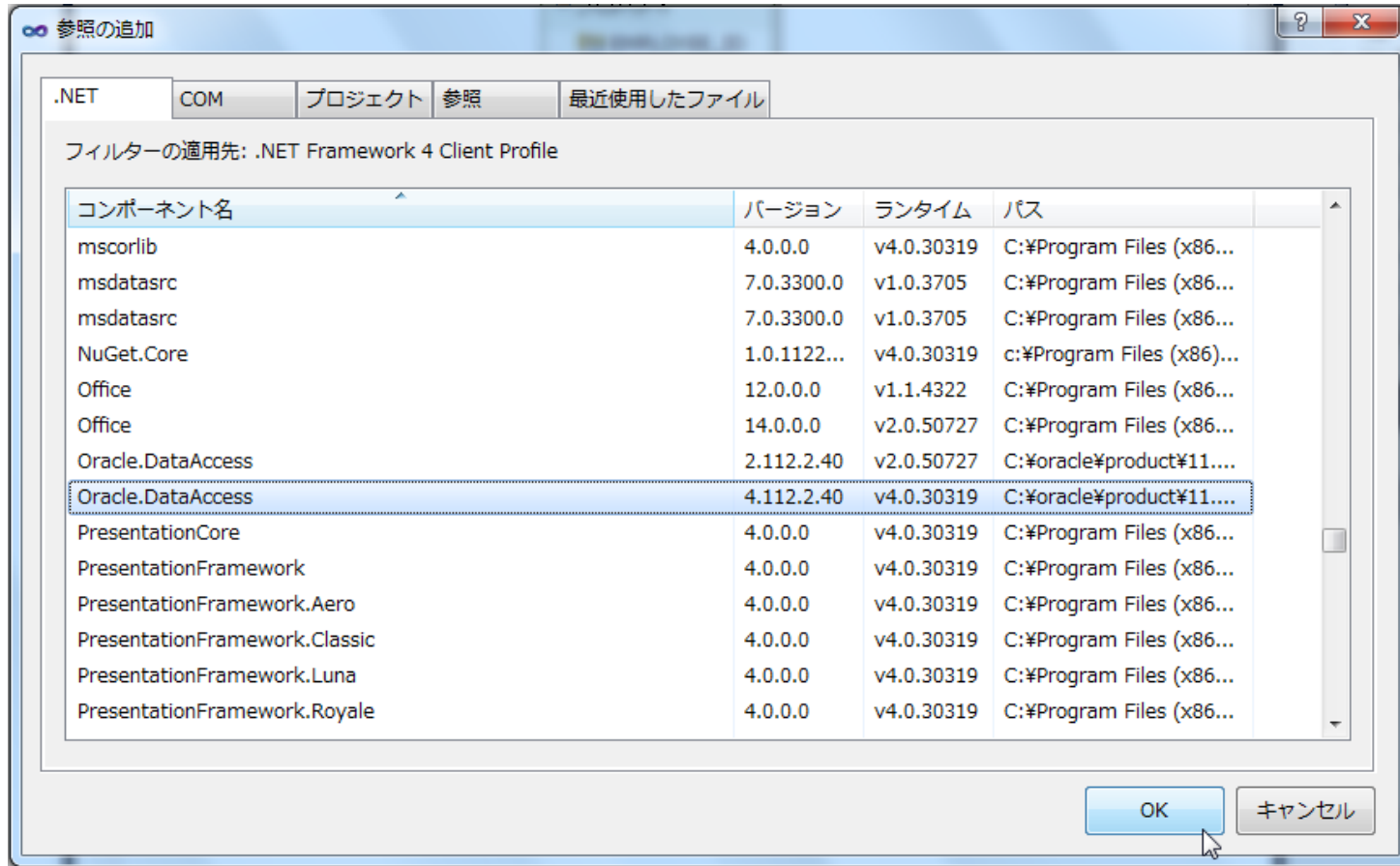
## ADO.NET Entity Data Modelの利用



「参照の追加」

# Entity Framework with ODP.NET (7)

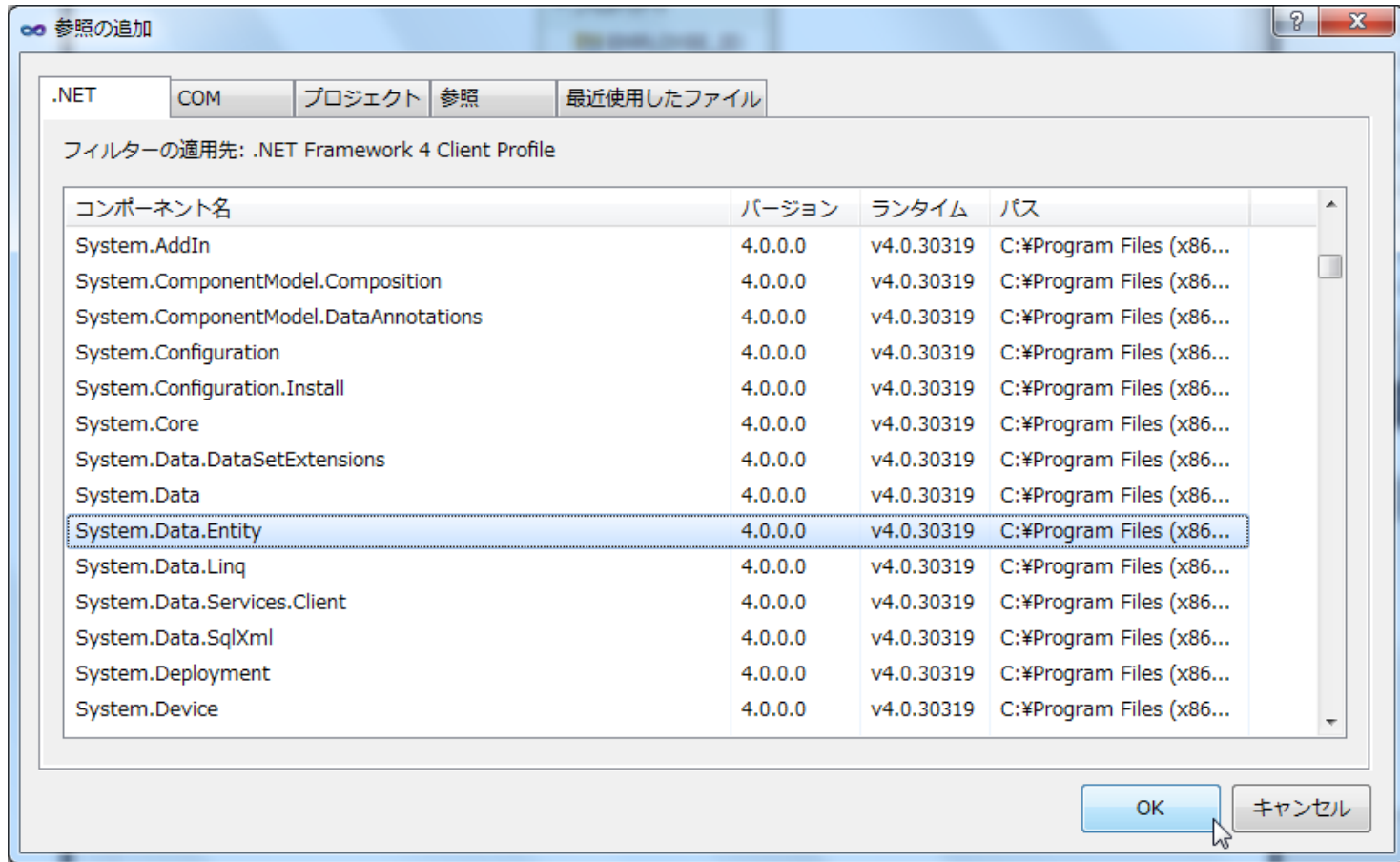
## ADO.NET Entity Data Modelの利用



「Oracle.DataAccess」を選択

# Entity Framework with ODP.NET (8)

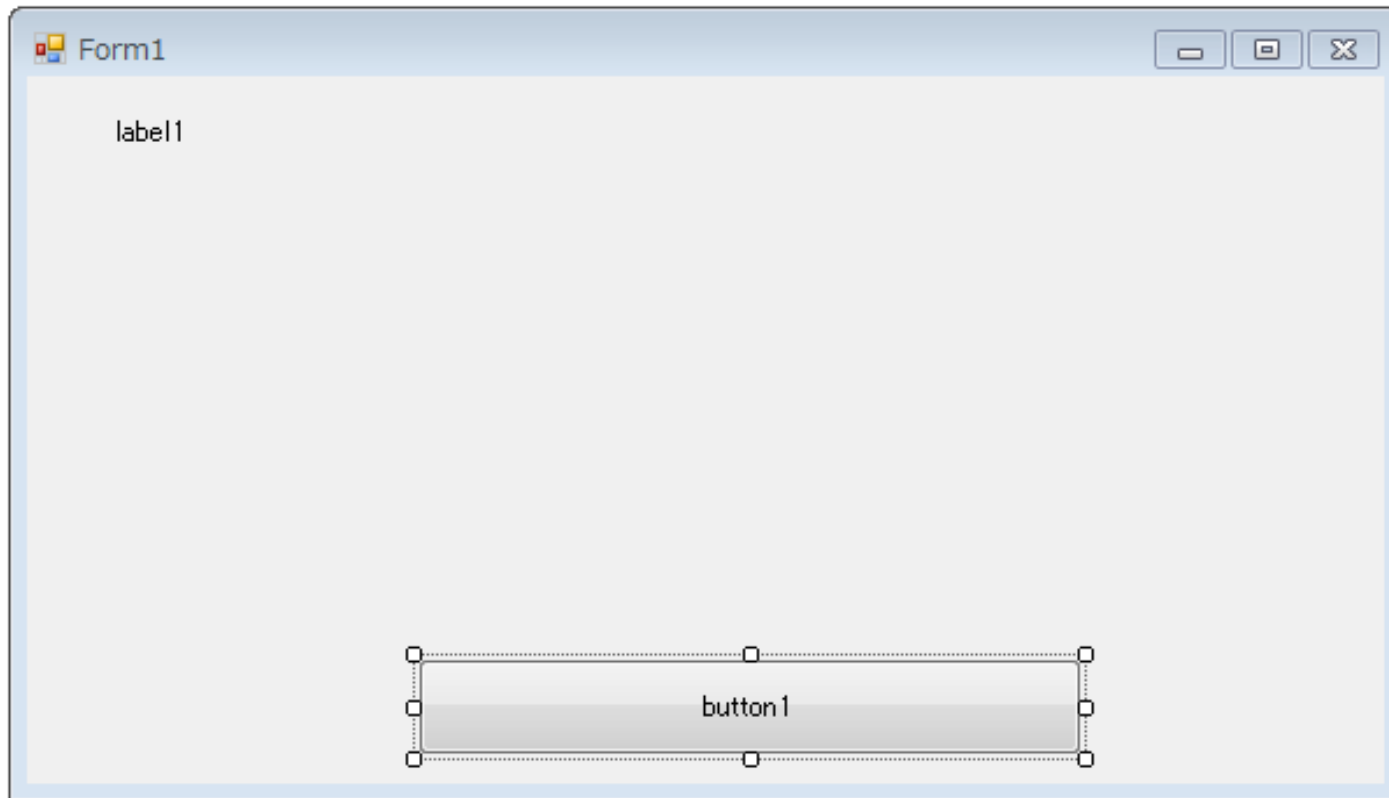
## ADO.NET Entity Data Modelの利用



「System.Data.Entity」を選択

# Entity Framework with ODP.NET (9)

## ADO.NET Entity Data Modelの利用



サンプル・フォームを作成

# Entity Framework with ODP.NET (10)

## ADO.NET Entity Data Modelの利用

```
private void button1_Click(object sender, EventArgs e)
{
    string sOutput = "";

    using (HREntities ctx = new HREntities())
    {
        int max_id = 102;

        // LINQ to Entities query -- Retrieve employees with ID
        var OraLINQ1 = from emp in ctx.EMPLOYEES
                       where emp.EMPLOYEE_ID <= max_id
                       select emp;

        foreach (var result in OraLINQ1)
        {
            sOutput += "ID : " + result.EMPLOYEE_ID + "¥r¥n";
            sOutput += "NAME : " + result.FIRST_NAME + "¥r¥n";
            sOutput += "SALARY : " + result.SALARY + "¥r¥n¥r¥n";
        }

        label1.Text = sOutput;
    }
}
```

サンプル・コードの記述

# Entity Framework with ODP.NET (11)

## ADO.NET Entity Data Modelの利用

```
SELECT
  "Extent1"."EMPLOYEE_ID" AS "EMPLOYEE_ID",
  "Extent1"."FIRST_NAME" AS "FIRST_NAME",
  "Extent1"."LAST_NAME" AS "LAST_NAME",
  "Extent1"."EMAIL" AS "EMAIL",
  "Extent1"."PHONE_NUMBER" AS "PHONE_NUMBER",
  "Extent1"."HIRE_DATE" AS "HIRE_DATE",
  "Extent1"."JOB_ID" AS "JOB_ID",
  "Extent1"."SALARY" AS "SALARY",
  "Extent1"."COMMISSION_PCT" AS "COMMISSION_PCT",
  "Extent1"."MANAGER_ID" AS "MANAGER_ID",
  "Extent1"."DEPARTMENT_ID" AS "DEPARTMENT_ID"
FROM
  "HR"."EMPLOYEES" "Extent1"
WHERE
  ("Extent1"."EMPLOYEE_ID" <= :p_linq_0)
```

SQLトレースで実行されたSQLの確認

# Agenda

- .NETとOracle Databaseについて
- Oracle Data Provider for .NET(ODP.NET)の新機能
- **最新のデータアクセス手法について**
- **64bitに対応したODP.NETの利用**



# 64-bit ODP.NET

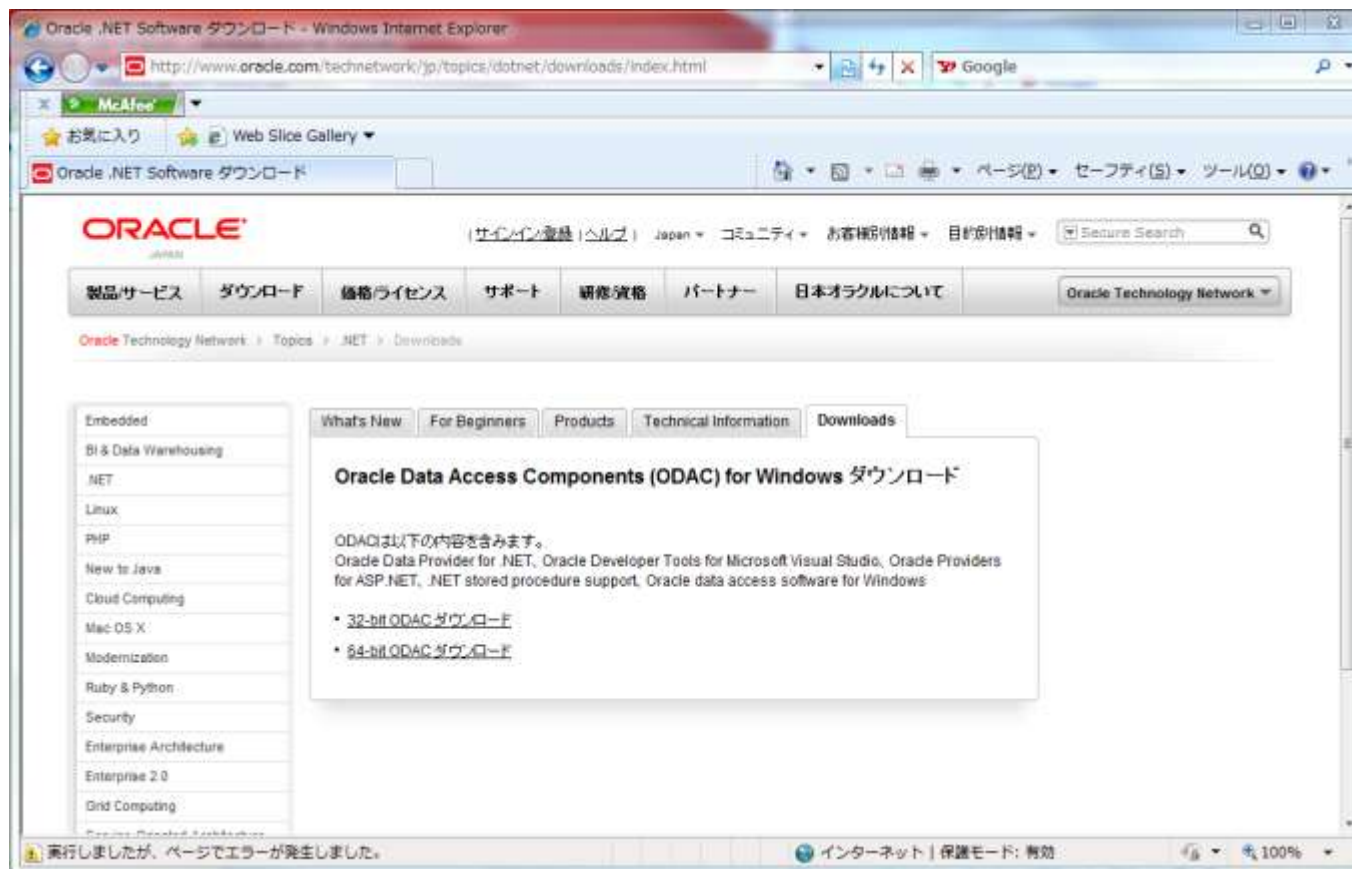
- 64-bit Oracle Data Access Components(ODAC)に含まれます。
- 以下の2つのバージョンで利用可能
  - ODAC 11.2.0.2.1
  - ODAC 10.2.0.3
- OTNからダウンロード可能
  - <http://www.oracle.com/technetwork/jp/database/windows/downloads/index-214820-ja.html>
- 利用方法
  - CodeZine 64-bitに対応した「ODP.NETを使ってみよう！」
    - <http://codezine.jp/article/detail/5914>

# 64-bit ODP.NET

- 64-bit ODP.NETのメリット
  - 広大なメモリ空間の利用
    - Datasetに大量のデータを取り込む際に効果大
  - 64-bit OS環境で最適なパフォーマンス
    - 32-bitアプリケーションを64-bit OS上で動作させた場合、Windows on Windows(WOW)上で動作。ボトルネックが発生

# 64-bit ODP.NET

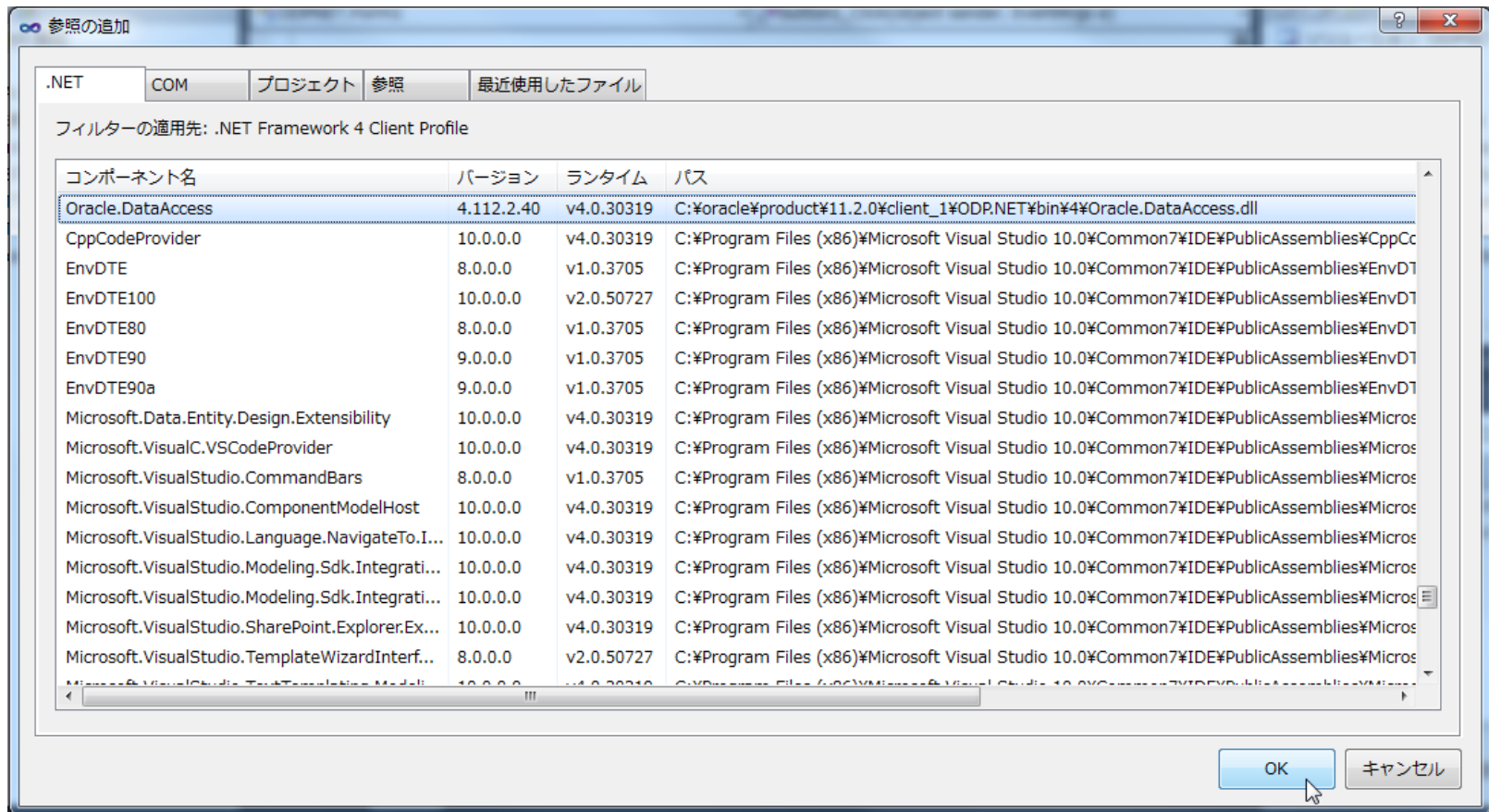
## ダウンロード



ダウンロードURL : <http://www.oracle.com/technetwork/jp/topics/dotnet/downloads/index.html>

# 64-bit ODP.NET

## Visual Studio からの利用

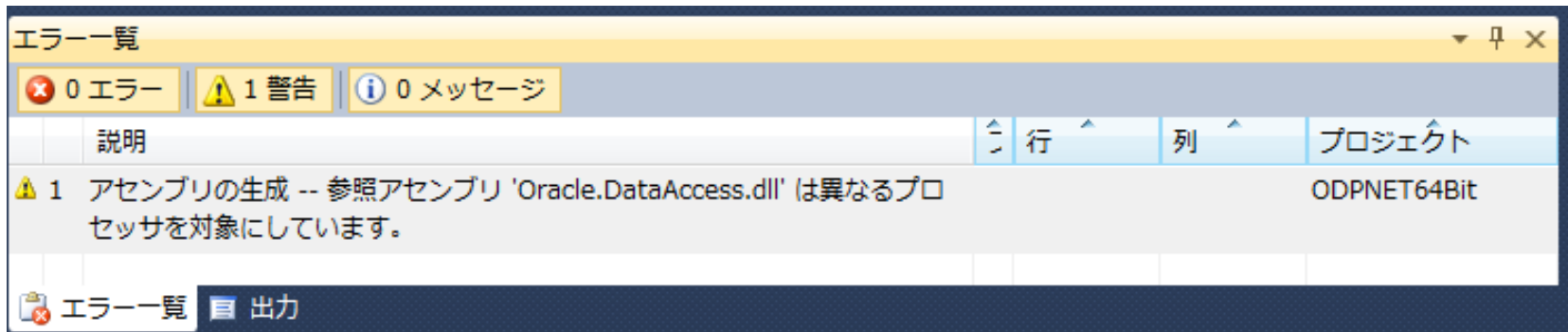


「参照の追加」→「OracleDataAccess」

# 64-bit ODP.NET

## Visual Studio からの利用

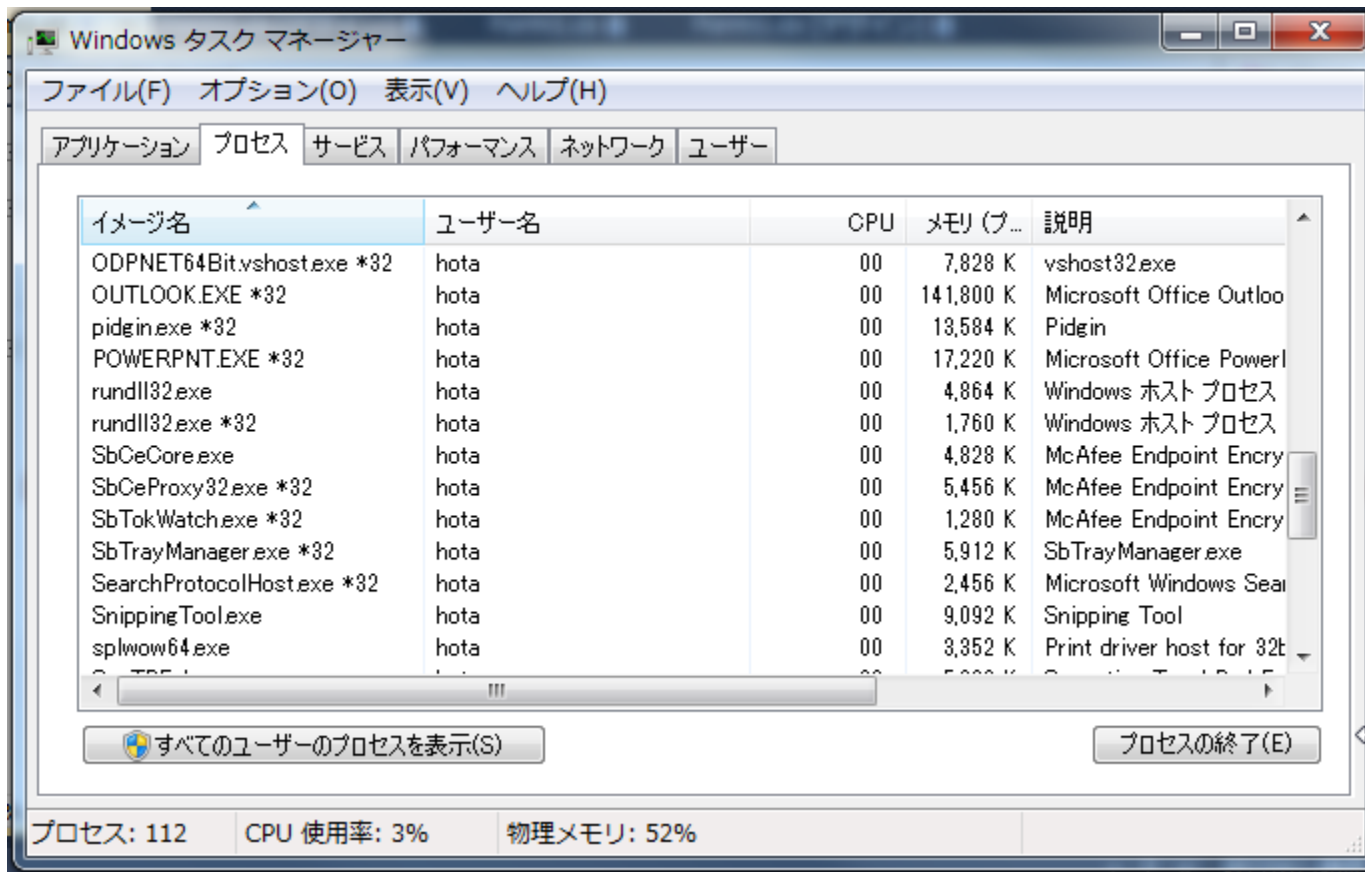
アプリケーションをそのまま実行すると以下のエラーが発生します。



# 64-bit ODP.NET

## Visual Studio からの利用

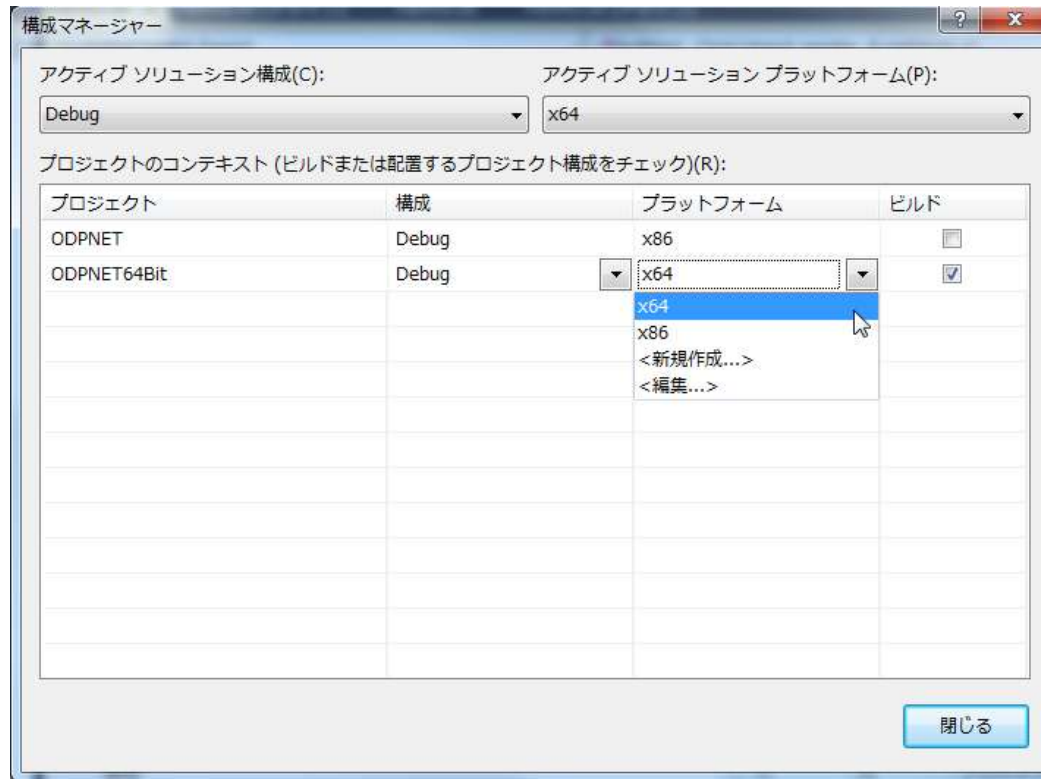
アプリケーションの動作環境をタスクマネージャーから確認



# 64-bit ODP.NET

## Visual Studio からの利用

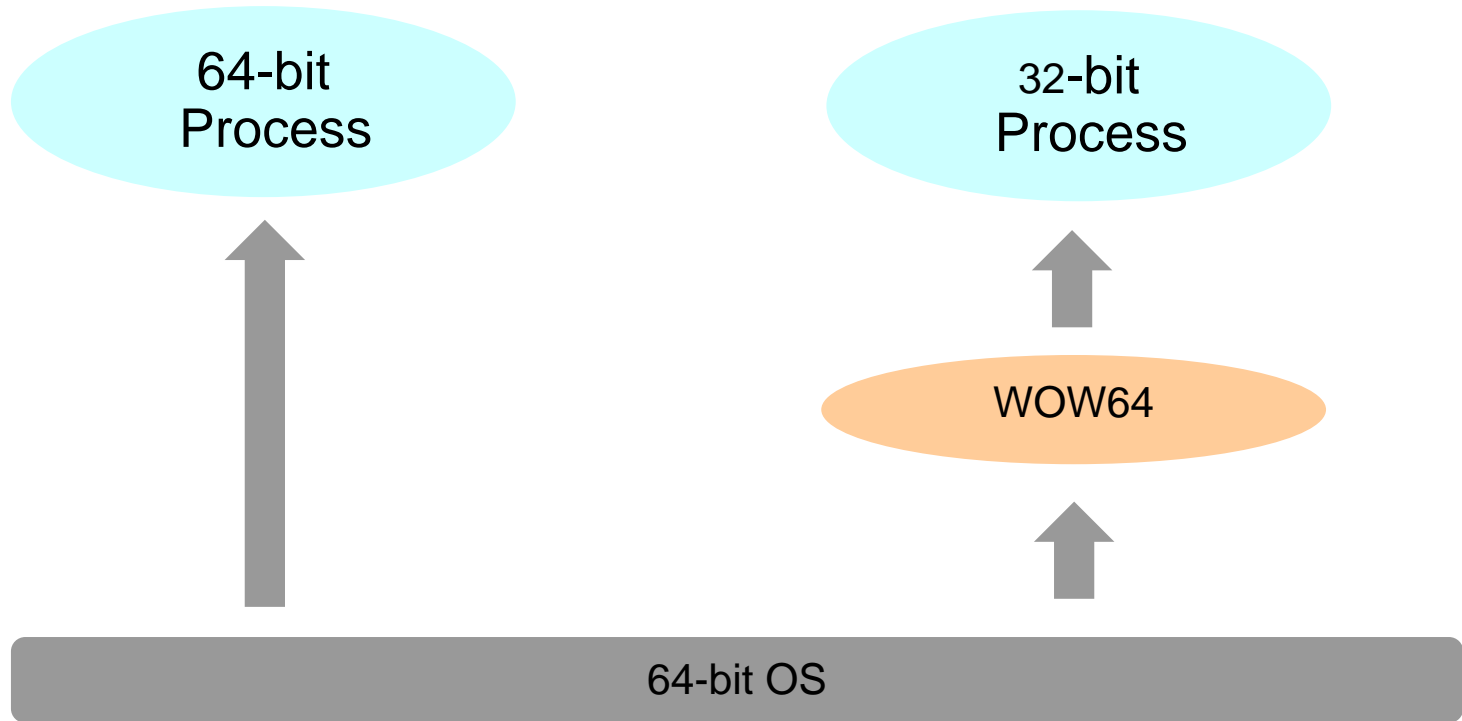
プラットフォームをx64に変更して再実行すると正常に実行されます。



デフォルトの「Any CPU」の利用は推奨しません。

# 64-bit ODP.NET

Visual Studio からの利用



32-bitアプリケーションは「WOW64」上で実行



# 64-bit ODP.NET

## Visual Studio からの利用

### 大量データにアクセスするサンプルコードを記述

```
// Oracle Databaseへの接続設定とデータ取得SQLの設定
OracleConnection conn = new OracleConnection("user id=sh;password=oracle:data source=orcl11gr2");
OracleCommand cmd = new OracleCommand("select * from sales", conn);
OracleDataAdapter da = new OracleDataAdapter(cmd);
DataSet dt = new DataSet();

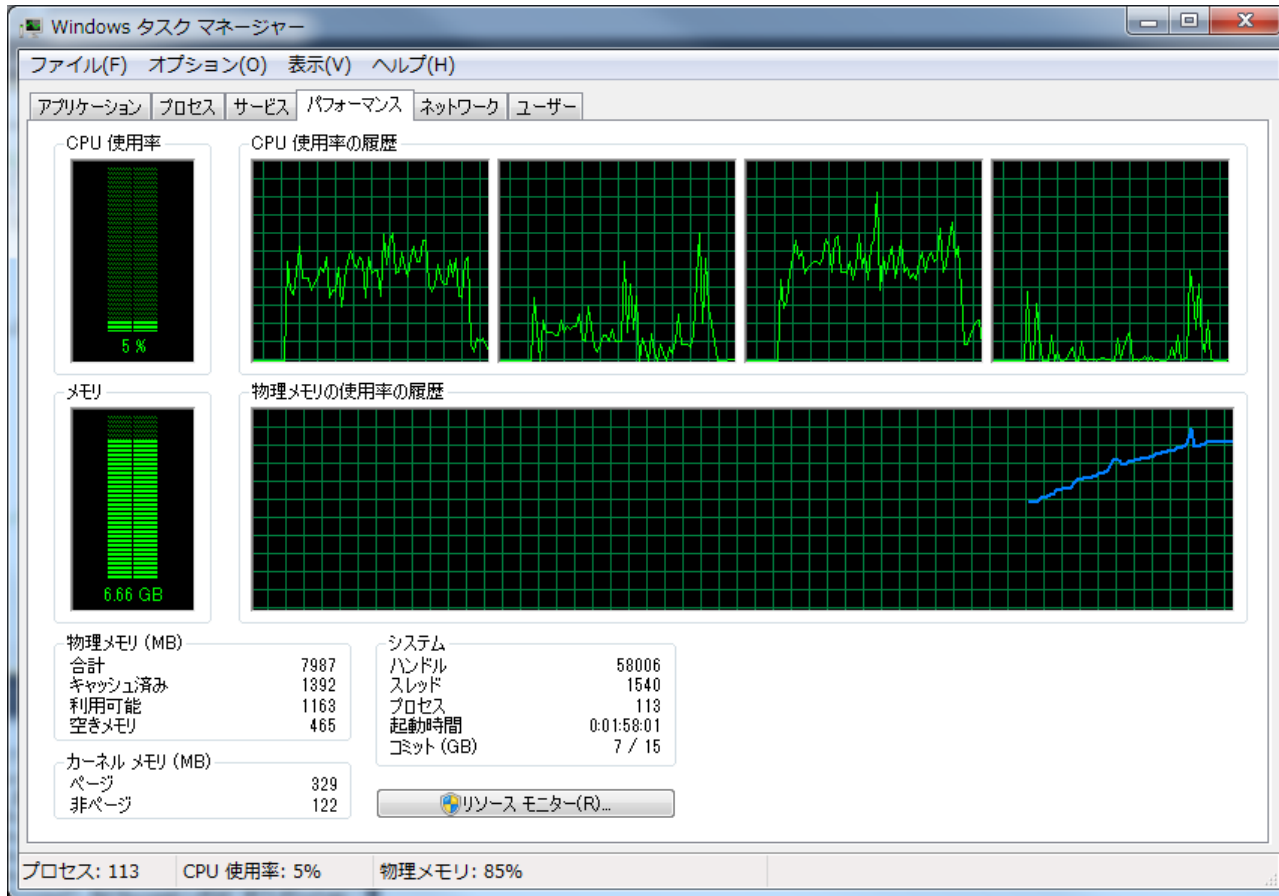
// 約100万行データの取得を10回繰り返す
for (int iCnt = 0; iCnt < 10; iCnt++)
{
    da.Fill(dt);
}
MessageBox.Show("データ取得が完了しました");

// 終了処理
conn.Close();

da.Dispose();
cmd.Dispose();
conn.Dispose();
```

# 64-bit ODP.NET

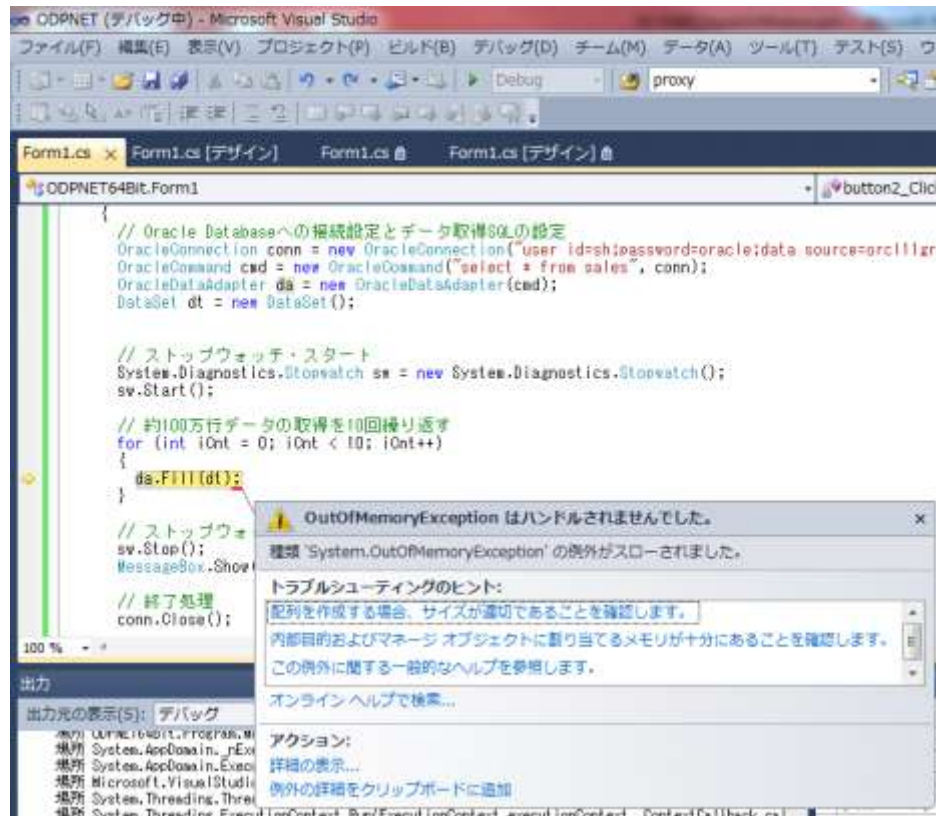
## Visual Studio からの利用



「Windows タスク マネージャー」からメモリ使用量の確認

# 64-bit ODP.NET

## Visual Studio からの利用



32-bitで実行した場合は上記エラーが発生

# まとめ

- .NETからOracle Databaseへの接続
  - 最新のODP.NETによるパフォーマンスの向上
  - 「Entity Framework」にも間もなく対応予定
  - 64-bit ODP.NETによる大量データアクセス

# OTNセミナーオンデマンド

コンテンツに対する  
ご意見・ご感想を是非お寄せください。

OTNオンデマンド 感想



[http://blogs.oracle.com/oracle4engineer/entry/otn\\_ondemand\\_questionnaire](http://blogs.oracle.com/oracle4engineer/entry/otn_ondemand_questionnaire)

上記に簡単なアンケート入力フォームをご用意しております。

セミナー講師/資料作成者にフィードバックし、  
コンテンツのより一層の改善に役立てさせていただきます。

是非ご協力をよろしくお願いいたします。

# OTNセミナーオンデマンド

日本オラクルのエンジニアが作成したセミナー資料・動画ダウンロードサイト

## 掲載コンテンツカテゴリ(一部抜粋)

Database 基礎

Database 現場テクニック

Database スペシャリストが語る

Java

WebLogic Server/アプリケーション・グリッド

EPM/BI 技術情報

サーバー

ストレージ



超入門! Oracle データベースって何  
再生時間: 60分

100以上のコンテンツをログイン不要でダウンロードし放題

データベースからハードウェアまで充実のラインナップ

毎月、旬なトピックの新作コンテンツが続々登場

## 例えばこんな使い方

- 製品概要を効率的につかむ
- 基礎を体系的に学ぶ/学ばせる
- 時間や場所を選ばず(オンデマンド)に受講
- スマートフォンで通勤中にも受講可能



毎月チェック!



コンテンツ一覧 はこちら

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

新作&おすすめコンテンツ情報 はこちら

<http://oracletech.jp/seminar/recommended/000073.html>

OTNオンデマンド



# オラクルエンジニア通信

オラクル製品に関わるエンジニアの方のための技術情報サイト

## オラクルエンジニア通信 - 技術資料、マニュアル、セミナー

Oracleエンジニアのための技術情報サイト by Oracle Japan

**新着情報を知りたい**

**技術資料を探したい**

**セミナーを受けたい**

### About

Oracleエンジニアの方がスキルアップしていただくために、厳選した情報をお届けしています

<b>技術資料</b>	<b>インストールガイド・設定チュートリアルetc. 欲しい資料への最短ルート</b>	<b>アクセスランキング</b>	<b>他のエンジニアは何を見ているのか？人気資料のランキングは毎月更新</b>
<b>特集テーマ Pick UP</b>	<b>性能管理やチューニングなど月間テーマを掘り下げて詳細にご説明</b>	<b>技術コラム</b>	<b>SQLスクリプト、索引メンテナンスetc. 当たり前運用/機能が見違える!?</b>

<http://blogs.oracle.com/oracle4engineer/>

オラクルエンジニア通信





製品/技術  
情報



Oracle Databaseっていくら？オプション機能も見積れる簡単ツールが大活躍

セミナー



基礎から最新技術までお勧めセミナーで自分にあった学習方法が見つかる

スキルアップ



ORACLE MASTER ! 試験頻出分野の模擬問題と解説を好評連載中

Viva!  
Developer



全国で活躍しているエンジニアにスポットライト。きらりと輝くスキルと視点を盗もう

<http://oracletech.jp/>

oracletech





あなたにいちばん近いオラクル



# Oracle Direct

まずはお問合せください

Oracle Direct



システムの検討・構築から運用まで、ITプロジェクト全般の相談窓口としてご支援いたします。  
システム構成やライセンス/購入方法などお気軽にお問い合わせ下さい。

## Web問い合わせフォーム

専用お問い合わせフォームにてご相談内容を承ります。  
[http://www.oracle.co.jp/inq\\_pl/INQUIRY/quest?rid=28](http://www.oracle.co.jp/inq_pl/INQUIRY/quest?rid=28)

※フォームの入力にはログインが必要となります。  
※こちらから詳細確認のお電話を差し上げる場合がありますので  
ご登録の連絡先が最新のものになっているかご確認下さい。

## フリーダイヤル

0120-155-096

※月曜～金曜  
9:00～12:00、13:00～18:00  
(祝日および年末年始除く)

ORACLE®

# **Hardware and Software Engineered to Work Together**

**ORACLE®**