

XML WEBサービス

Part.3

Visual Basic 2008
+
Oracle Database 11g Release 1

2008.06.07

初音 玲



わんくま同盟 東京勉強会 #22

index

WEBサービス

OracleとADO.NETの関係

接続

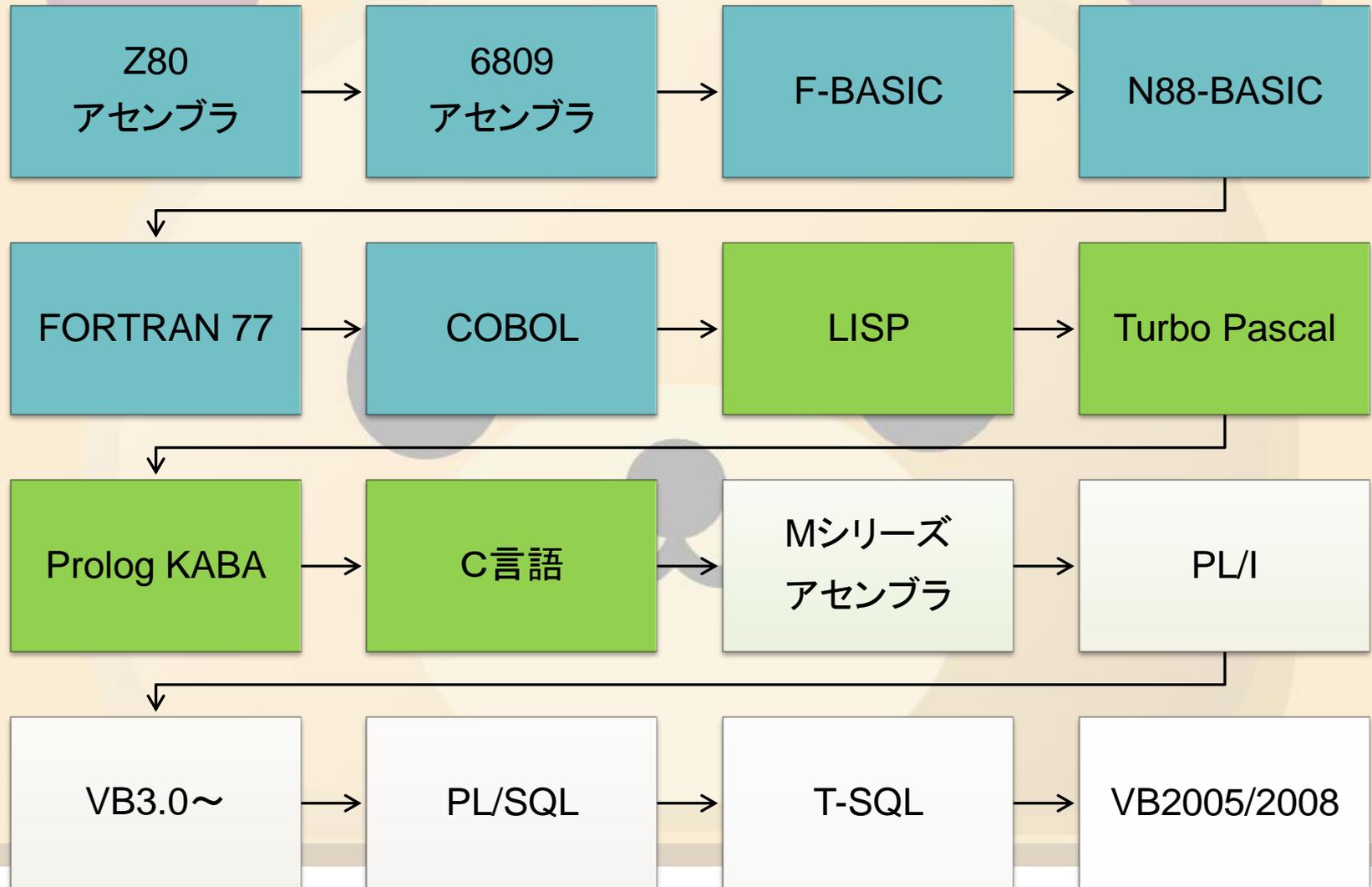
データ取得

データ更新

権限



自己紹介



はじめに

WEBサービス

OracleとADO.NETの関係

接続

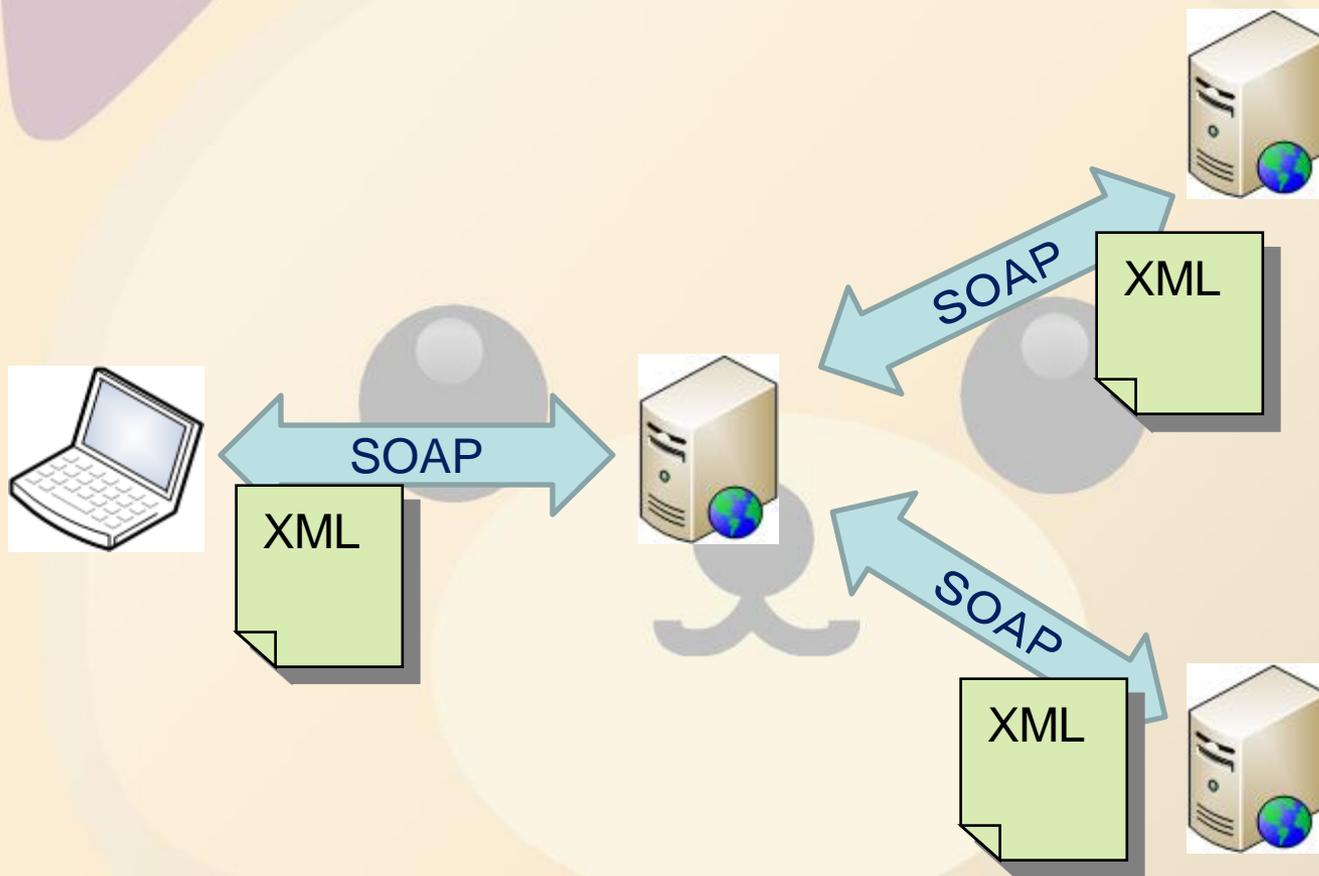
データ取得

データ更新

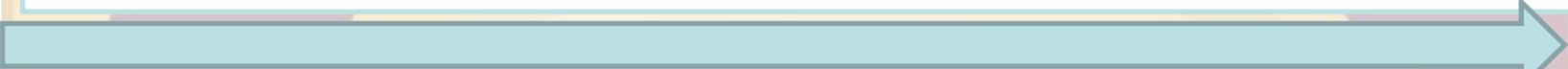
権限



XMLWEBサービスとは



SOAPとは



```
POST /WankumaTokyo22/Connection.asmx HTTP/1.1
Host: localhost Content-Type: application/soap+xml; charset=utf-8
Content-Length: length
<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi=.....>
<soap12:Body>
  :
</soap12:Body>
</soap12:Envelope>
```



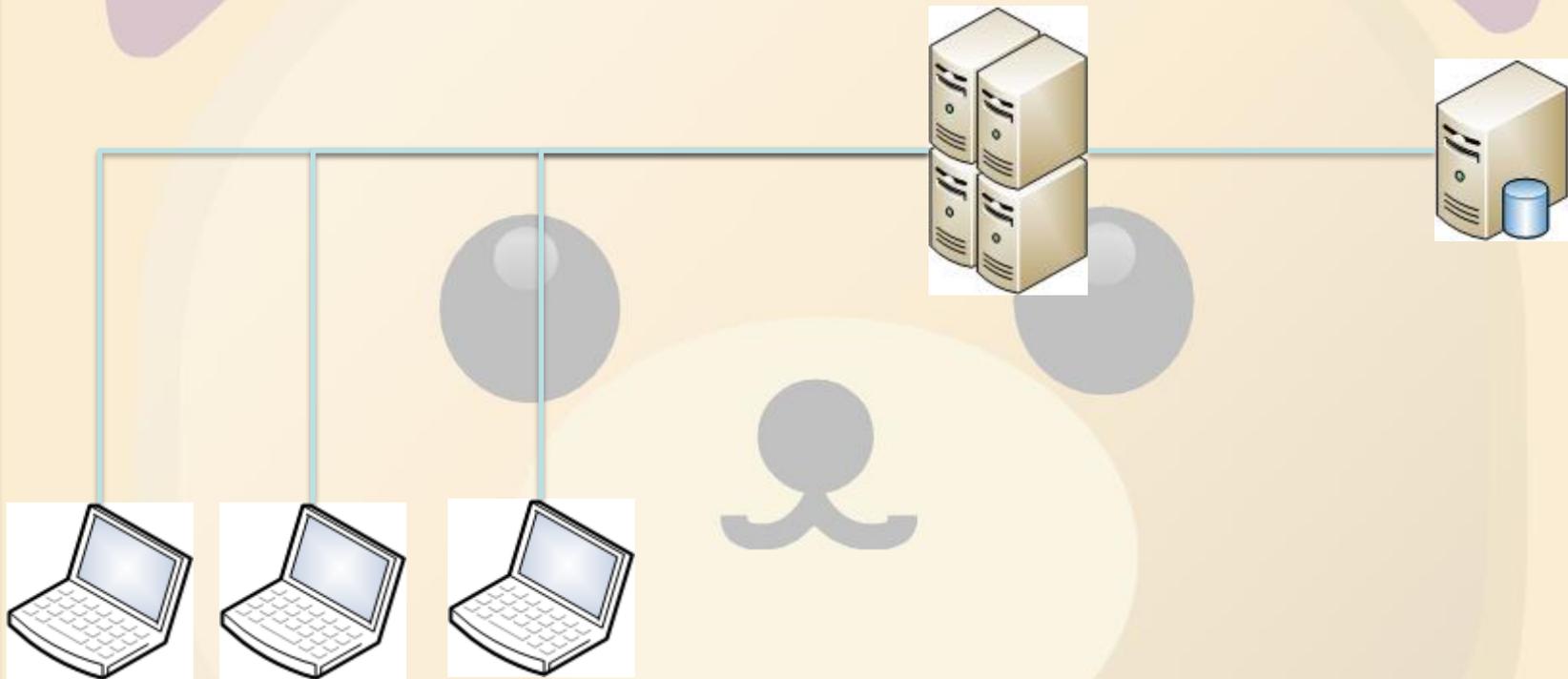
```
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length
<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi=.....>
<soap12:Body>
  :
</soap12:Body>
</soap12:Envelope>
```

Windowsアプリ+WEBサービス

Winアプリ

XML WEBサービス

DB



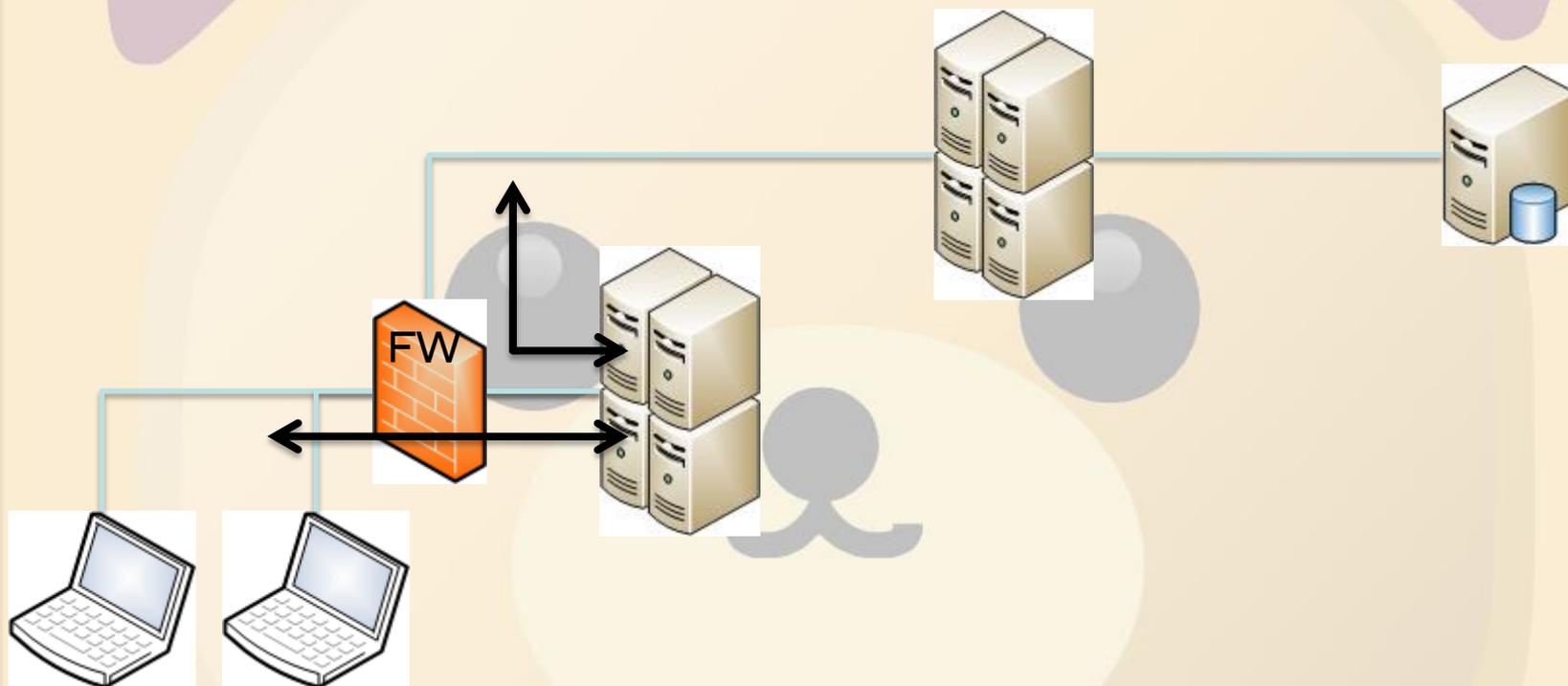
WEBアプリ+WEBサービス

ブラウザ

WEBアプリ

XML WEBサービス

DB



WEBサービスの配置について

開発環境

実行環境

サイトの
発行

MSIL

ソース



MSIL

手動XCOPY

MSIL

初回時

naitive

MSIL

手動XCOPY

MSIL

初回時

naitive

FTP

MSIL

初回時

naitive

HTTP

MSIL

初回時

naitive

ソース

手動XCOPY

初回時

MSIL

初回時

naitive



Webサイトの発行

WEBアプリのときは、aspxファイルの内容をサイト上で更新できるかを決定

Web サイトの発行

ターゲットの場所(L): (http://...、https://... またはドライブ:*)

!Documents\Visual Studio 2008\Projects\WankumaTokyo22\PrecompiledWeb\WankumaTokyo22

このプリコンパイルされたサイトを更新可能にする(A)

固定名およびシングル ページ アセンブリを使用する(E)

プリコンパイル済みアセンブリで厳密な名前を有効にする(S)

◎ 厳密な名前

キー ファイルの場所:

遅延署名する(D)

◎ キー コンテナを使用する(I)

キー コンテナ:

アセンブリを AllowPartiallyTrustedCallerAttribute (A)

ASPX

dll (aspx.vbのMSIL)

JITコンパイル

テンポラリアセンブリ



プロジェクトと仮想フォルダの関係について

ソリューション

プロジェクト1

プロジェクト2

プロジェクト3

プロジェクト4

IIS

仮想フォルダ1

仮想フォルダ2

仮想フォルダ3

仮想フォルダ4

WP

WP

WP

WP

はじめに

WEBサービス

OracleとADO.NETの関係

接続

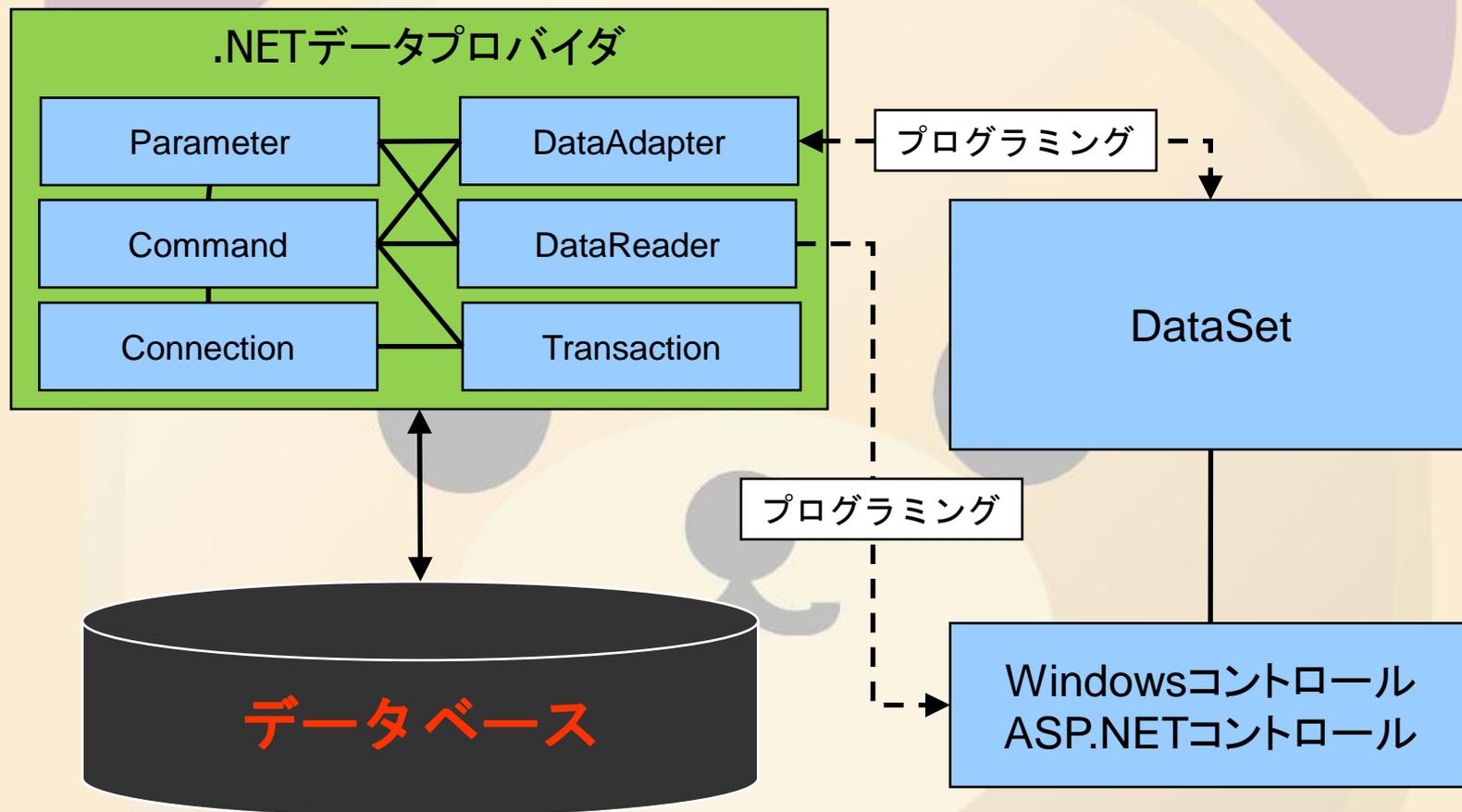
データ取得

データ更新

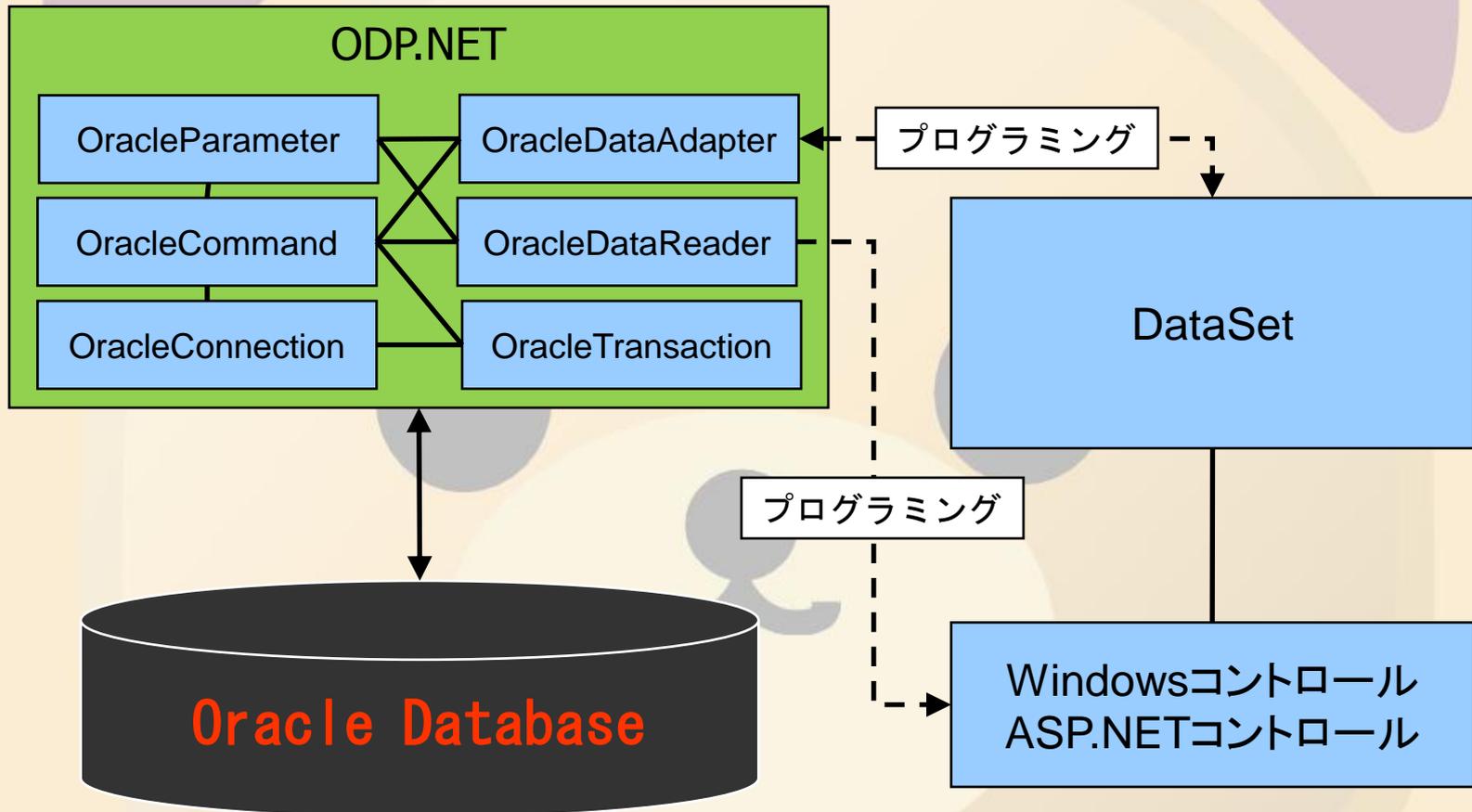
権限



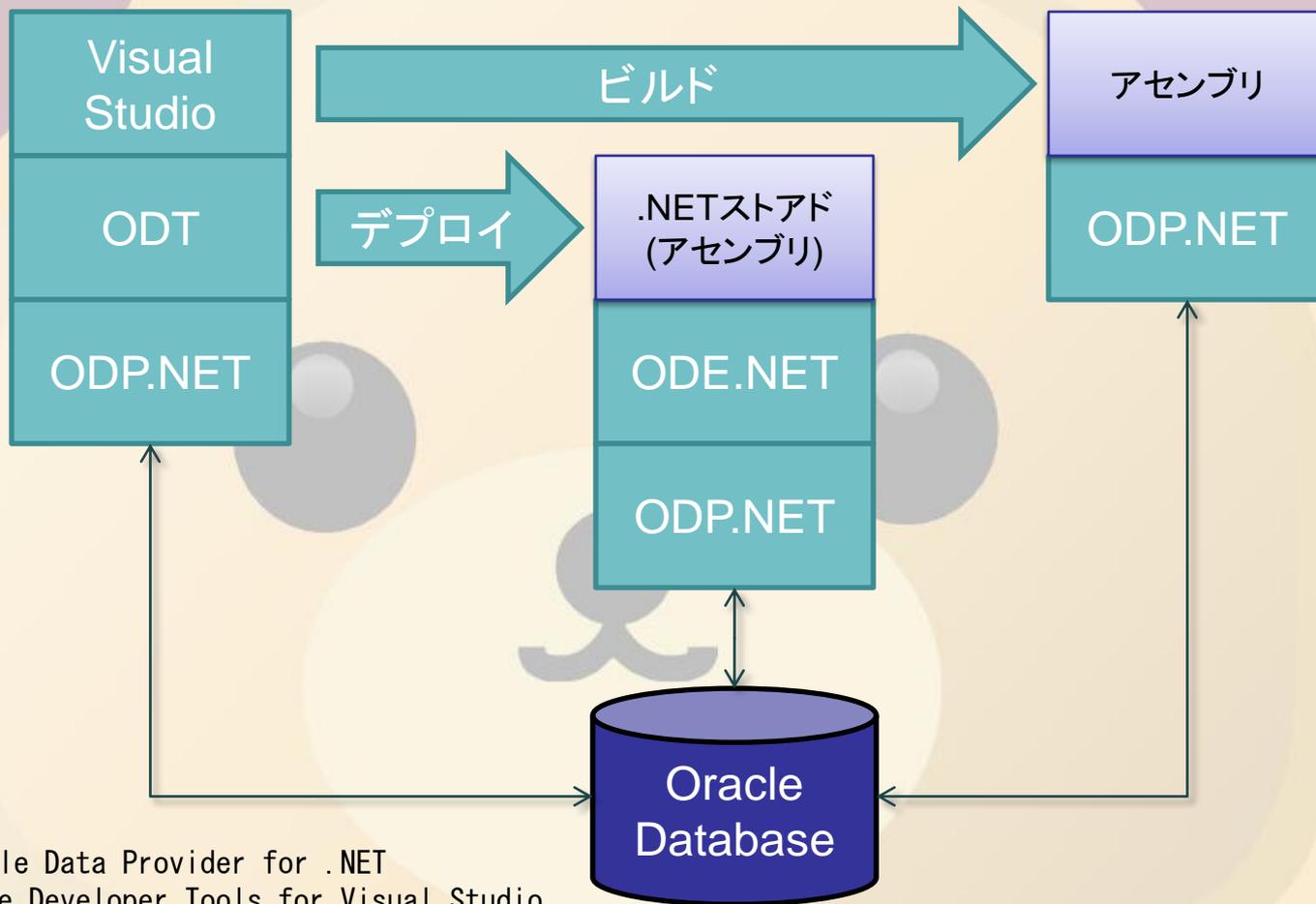
ADO.NETの基本的な構造



ODP.NETの基本的な構造



Visual StudioからOracleに接続



ODP.NET:Oracle Data Provider for .NET
ODT :Orace Developer Tools for Visual Studio
ODE.NET:Oracle Database Extensions for .NET

ODTとVSのバージョン関連表

	対応DBバージョン	対応Visual Studio
ODT 10.1.x.x	Oracle 9i Database Release 2 Oracle Database 10g Release 1	Visual Studio .NET 2003
ODT 10.2.0.1	Oracle 9i Database Release 2 Oracle Database 10g Release 1 Oracle Database 10g Release 2	Visual Studio .NET 2003
ODT 10.2.0.2	Oracle8i R8.1.7.4以降	Visual Studio .NET 2003 Visual Studio 2005
ODT 11.1.0.6	Oracle 9i Database Release 2～	Visual Studio .NET 2003 Visual Studio 2005 Visual Studio 2008

～ODT 10.2:サーバーエクスプローラとは別
ODT 11.1～:サーバーエクスプローラに統合

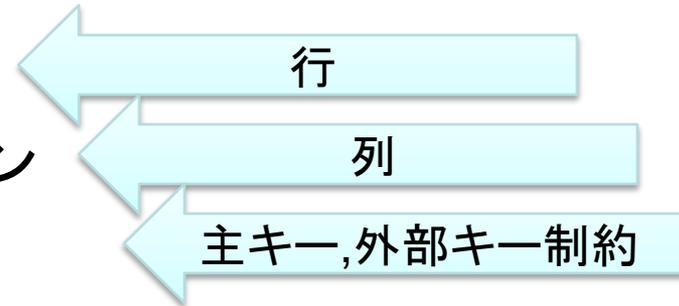
専用ミドルウェアは、やっぱり速い

	OLE DB .NET Data Provider	OLE DB Provider for Oracle	ODP.NET	oo4o
ログオン	2.4	3.0	3.2	1.0
参照	2.0	3.5	3.5	1.0
追加	0.7	1.0	1.0	1.0
更新	0.7	1.2	1.2	1.0

VB2005+oo4oの処理時間を1としたときの相対比
独自プログラムによる測定

DataSetクラス

- メモリ上の仮想データベース
- DataTablesコレクション
 - DataTableクラス
 - DataRowコレクション
 - DataColumnコレクション
 - Constraintsコレクション
- DataRelationsコレクション
 - RDBMSのリレーション定義に相当
 - 親子関係を定義



ODP.NET

OracleConnection
オブジェクト

- 特定のデータソースへの接続を確立

OracleCommand
オブジェクト

- データソースに対してコマンドを実行

OracleDataReader
オブジェクト

- データソースから前方向、読取専用でデータ取得

OracleDataAdapter
オブジェクト

- DataSetを設定し、データソースを使用して更新内容を解決

index

WEBサービス

OracleとADO.NETの関係

接続

データ取得

データ更新

権限



OracleConnectionオブジェクト

```
<WebMethod(Description:="認証チェック")> _  
Public Function IsLogin2(ByVal userID As String, _  
                        ByVal password As String) As Boolean  
    Dim isOK As Boolean = False  
    Dim cn As New OracleConnection()  
    cn.ConnectionString = _  
        String.Format(Setting.ConnectionString, userID, password)  
    cn.Open()  
    isOK = True  
    Try  
        cn.Close()  
    Catch ex As Exception  
    End Try  
End Function
```



ADO.NETからのエラーの取得

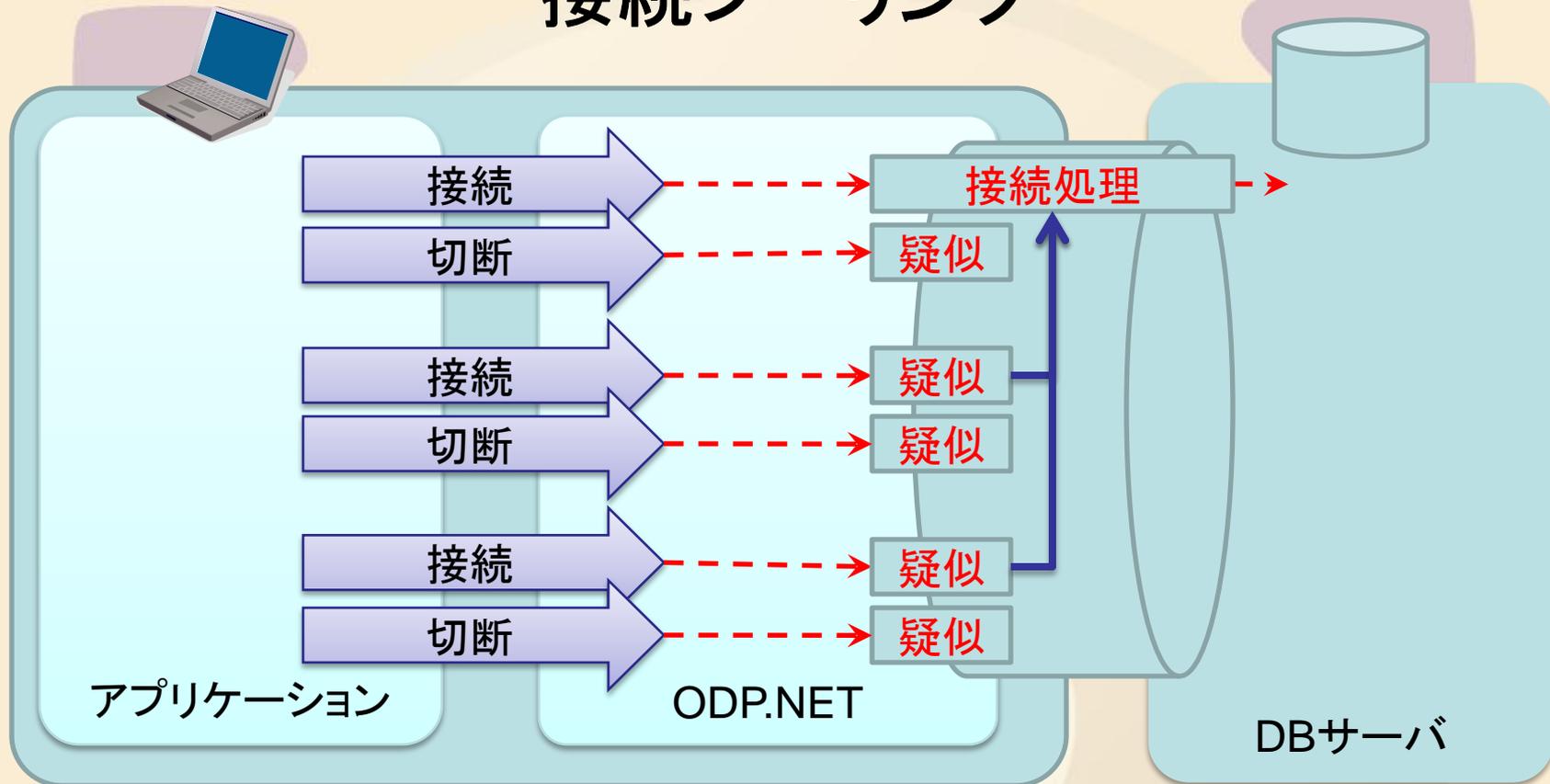
Try~Catch

- エラー処理をおこなって呼び出し元に戻る
- cn.OpenをTry~Catchで囲む
- CatchにはException

Global.asax

- エラーは例外として呼び出し元に戻る
- エラーログを一元取得したいとき
- Try~Catchでは囲まない

接続プーリング



User Id={0};Password={1};Data Source=ホスト文字列;

- ODP.NETのデフォルト動作はPooling=True

index

WEBサービス

OracleとADO.NETの関係

接続

データ取得

データ更新

権限



わんくま同盟 東京勉強会 #22

OracleDataReader

```
<WebMethod(Description:="EMPテーブルからEMPNO,ENAMEを取得する")> _
Public Function GetRecords(ByVal userID As String, _
                           ByVal password As String) As (Of TStatus)
    Using _cn As New Oracle.DataAccess.Client.OracleConnection()
        :
        Using _cmd As New OracleCommand(sqlString, _cn)
            Dim rd As OracleDataReader = Nothing
            rd = _cmd.ExecuteReader
            Do While rd.Read
                Dim item As New TStatus
                item.EmpNo = rd.Item("EMPNO").ToString
                item.Ename = rd.Item("ENAME").ToString
                emp.Add(item)
            Loop
        End Using
        _cn.Close()
    End Using
    Return emp
End Function
```



Parameterオブジェクト

以下の条件を与えたときの実行結果は？

HTTP POST プロトコルを使って操作をテストするには、[起動] ボタンをクリックしてください。

パラメータ	値
userID:	<input type="text" value="scott"/>
password:	<input type="text" value="tiger"/>
ename:	<input type="text" value="' OR '1'='1'"/>

index

WEBサービス

OracleとADO.NETの関係

接続

データ取得

データ更新

権限



データソースとDataSetクラスの対応付け

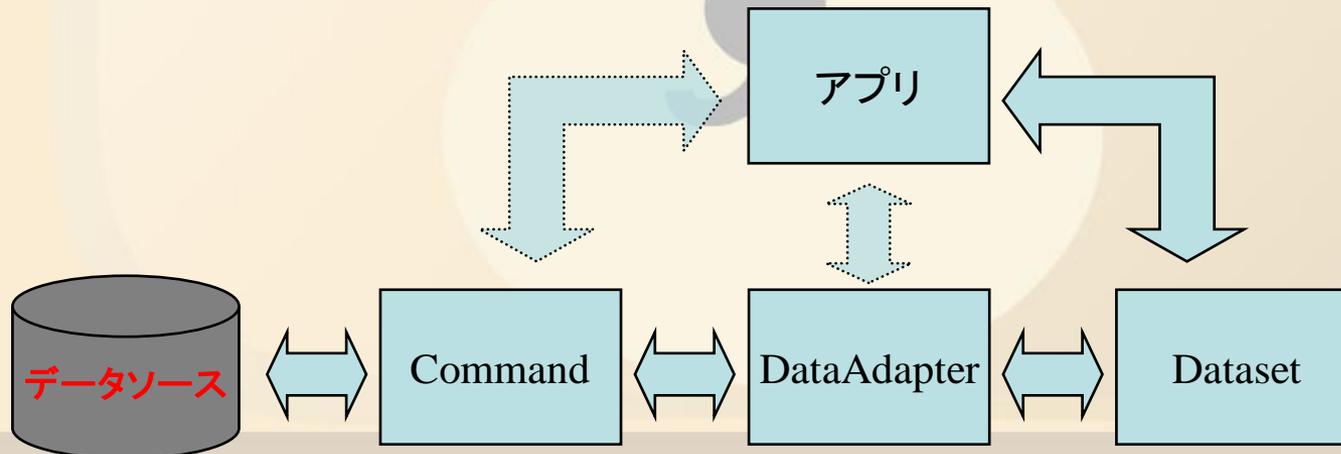
.NETデータプロバイダには、更新可能セットがない

DataSetは仮想的なデータベース

- もちろん更新も可能

.NETデータプロバイダとDataSetの相互乗り入れ

- DataSetによるデータソースの更新可能セットを実現



OracleDataAdapterオブジェクト

```
<WebMethod(Description:="EMPテーブルを取得する")> _  
    Public Function GetRecords(ByVal userID As String, _  
                               ByVal password As String) As System.Data.DataSet  
        Const sqlString As String = "SELECT * FROM EMP"  
        Dim Ds As New System.Data.DataSet  
  
        Using _cn As New OracleConnection()  
            _cn.ConnectionString =  
                String.Format(Setting.ConnectionString, userID, password)  
            _cn.Open()  
            Using _cmd As New OracleCommand(sqlString, _cn)  
                Using _da As New OracleDataAdapter(_cmd)  
                    _da.Fill(Ds, "EMP")  
                End Using  
            End Using  
            _cn.Close()  
        End Using  
        Return Ds  
    End Function
```



OracleCommandBuilderでSQL作成

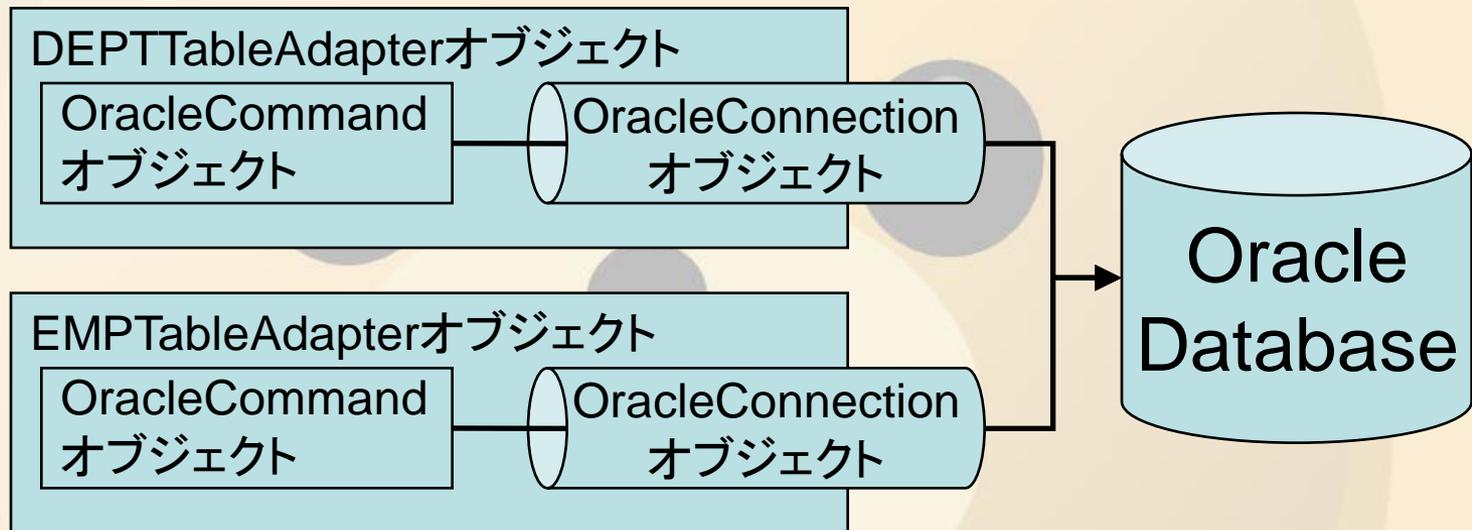
```
Using _tr As OracleTransaction = _cn.BeginTransaction()  
  Using _cmd As New OracleCommand(sqlString, _cn)  
    _cmd.Transaction = _tr    '###重要###  
    Using _da As New OracleDataAdapter(_cmd)  
      Using cb As New OracleCommandBuilder(_da)  
        _da.UpdateCommand = cb.GetUpdateCommand()  
        _da.InsertCommand = cb.GetInsertCommand()  
        _da.DeleteCommand = cb.GetDeleteCommand()  
        _da.Update(ds, "EMP")  
        isOK = True  
        If isOK Then  
          _tr.Commit()  
        Else  
          _tr.Rollback()  
        End If  
      End Using  
    End Using  
  End Using  
End Using
```



暗黙的なトランザクション

TableAdapterごとにConnectionが存在

– Connectionに対するトランザクションでは無理



MS-DTC (マイクロソフト分散トランザクションコーディネータ)

System.Transactionsクラスを使う

暗黙的なトランザクション

Try

Me.Validate()

Using _trn As New System.Transactions.TransactionScope

Me.DEPTBindingSource.EndEdit()

Me.DEPTTableAdapter.Update(Me.DataSet1.DEPT)

,

Me.EMPBindingSource.EndEdit()

Me.EMPTableAdapter.Update(Me.DataSet1.EMP)

,

_trn.Complete() ' トランザクション完了

End Using

Catch ex As **System.Transactions.TransactionAbortedException**

MessageBox.Show(ex.Message)

Catch ex As Exception

MessageBox.Show(ex.Message)

End Try

ADO.NET 2.0からは
COM+カタログ登録不要
GACへの登録不要

System.Transactionsの参照設定を忘れないように



index

WEBサービス

OracleとADO.NETの関係

接続

データ取得

データ更新

権限



権限

接続は、アプリ固有ユーザID

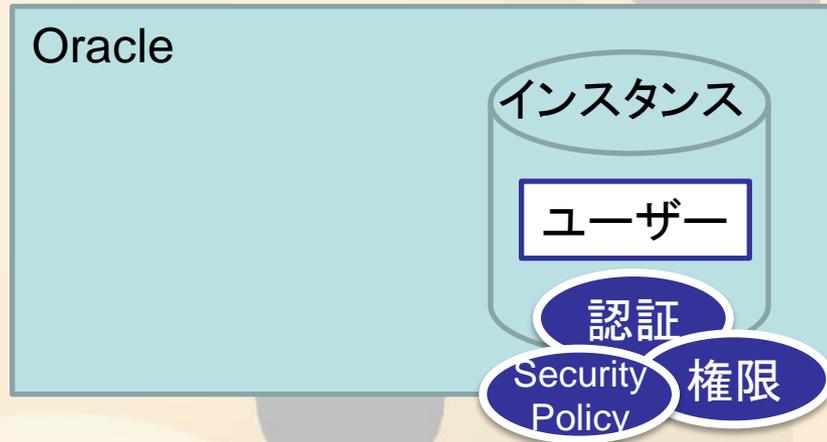
- コネクションプーリング
- アプリ側でアクセス制御
- 直接ツールで接続されたら？
- アプリ側にバグがあったら？

接続は、利用者固有ユーザID

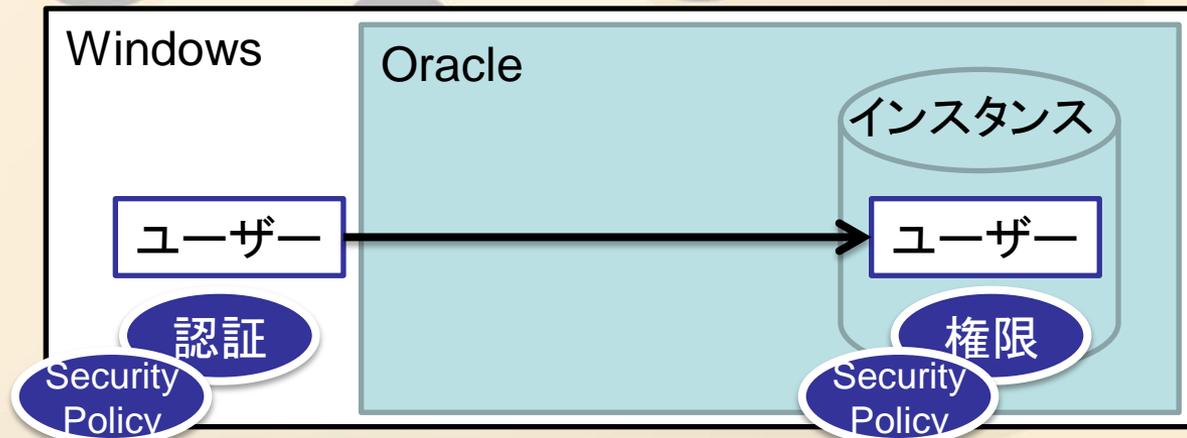
- コネクションプーリング
- DBの設定でアクセス制御
- 直接ツールで接続されても安全
- アプリ側にバグがあっても安全

Oracleにおけるユーザ管理

Oracle認証

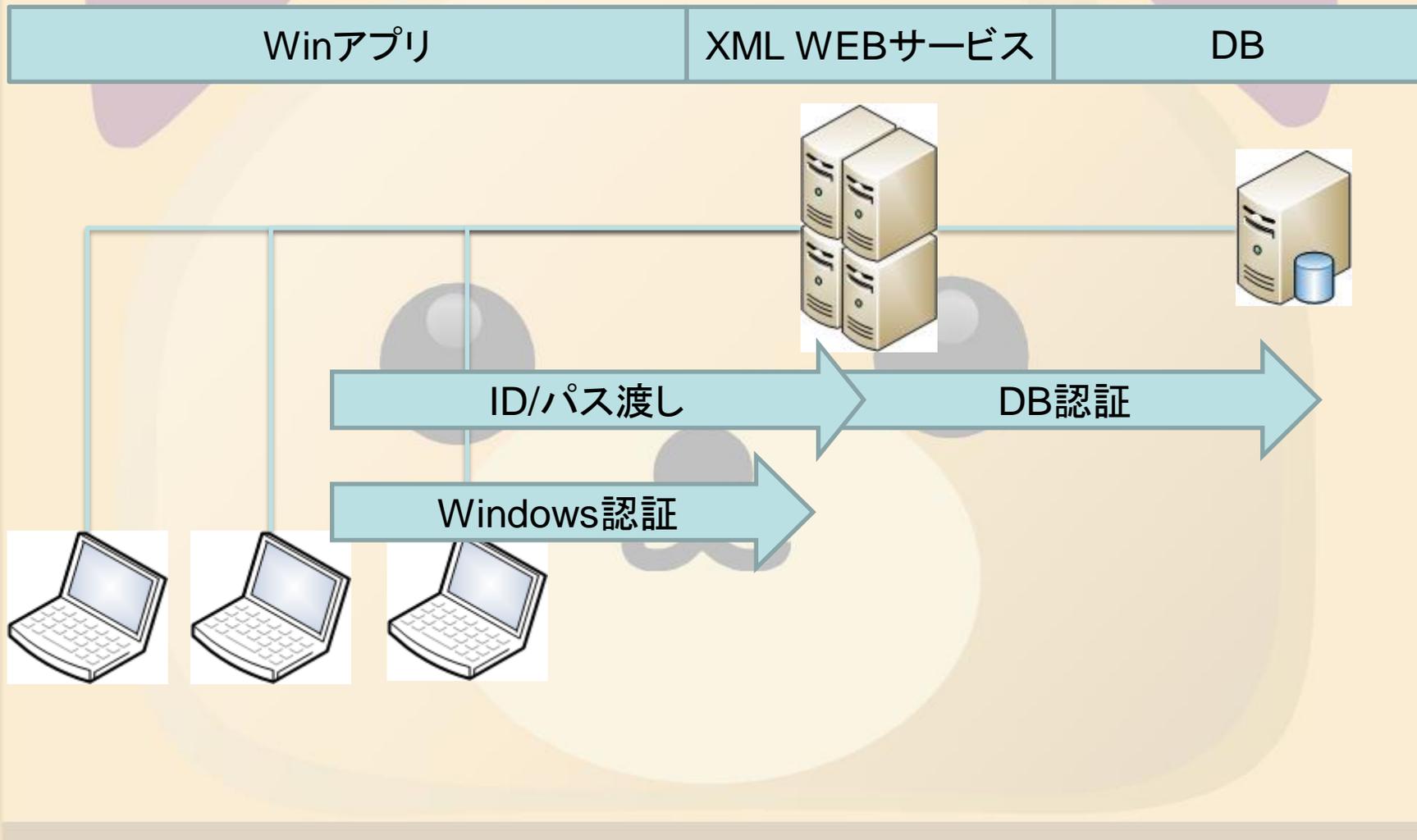


Windows認証

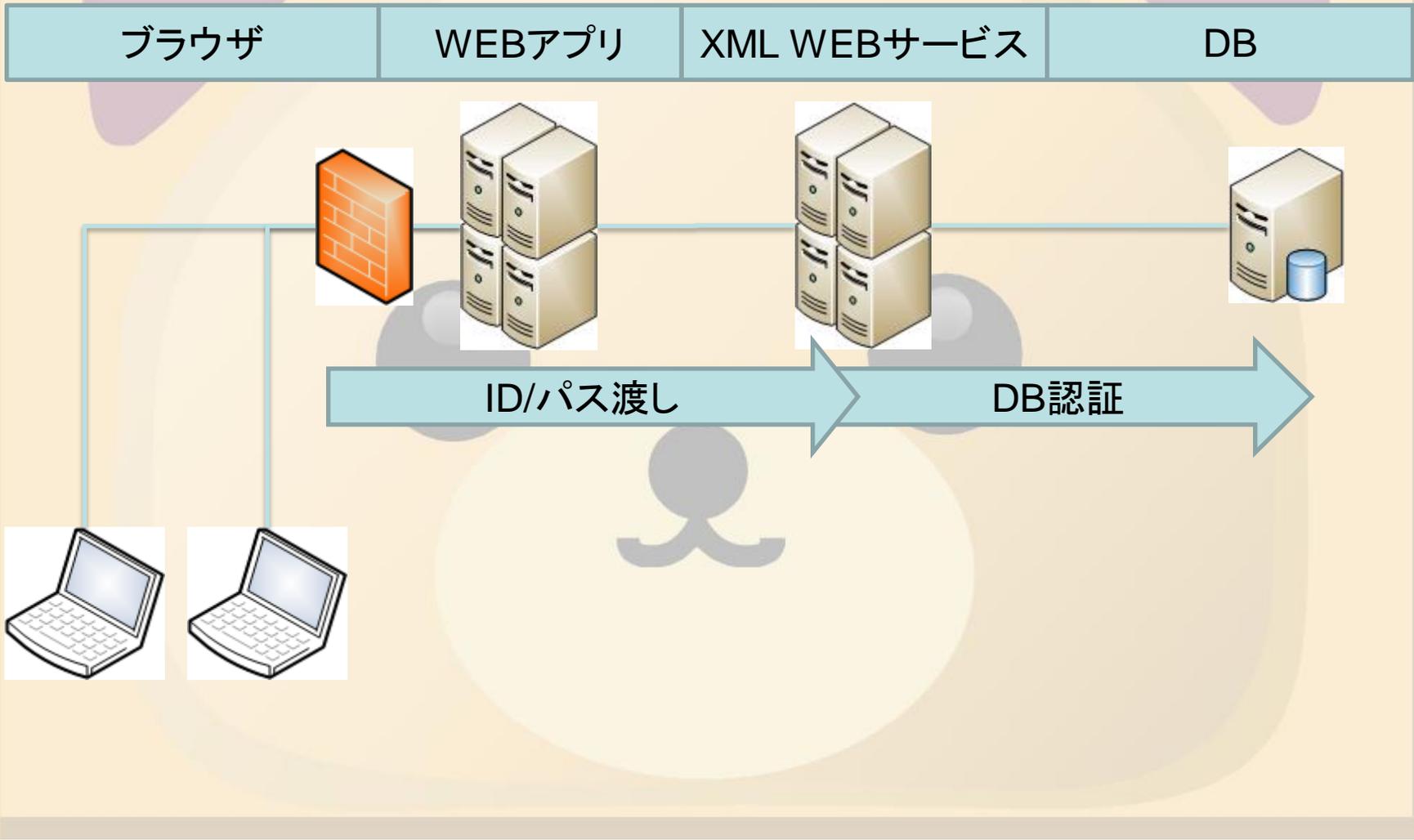


ローカル認証だと4万人くらいが限界なのでAD認証も考慮

Windowsアプリにおけるお勧め認証構造



WEBアプリにおけるお勧め認証構造



まとめ

- 接続
 - 接続プーリング
- データ取得
 - コード記述開発
 - GUI操作開発
- データ更新
 - CommandBuilder
 - トランザクション
- 権限
 - Oracle認証
 - Windows認証



わんくま同盟 東京勉強会 #22