



ORACLE®

NYOUG

Leveraging Oracle With Microsoft .NET

New York Oracle User Group
Bill Rose
October 28, 2008

Agenda

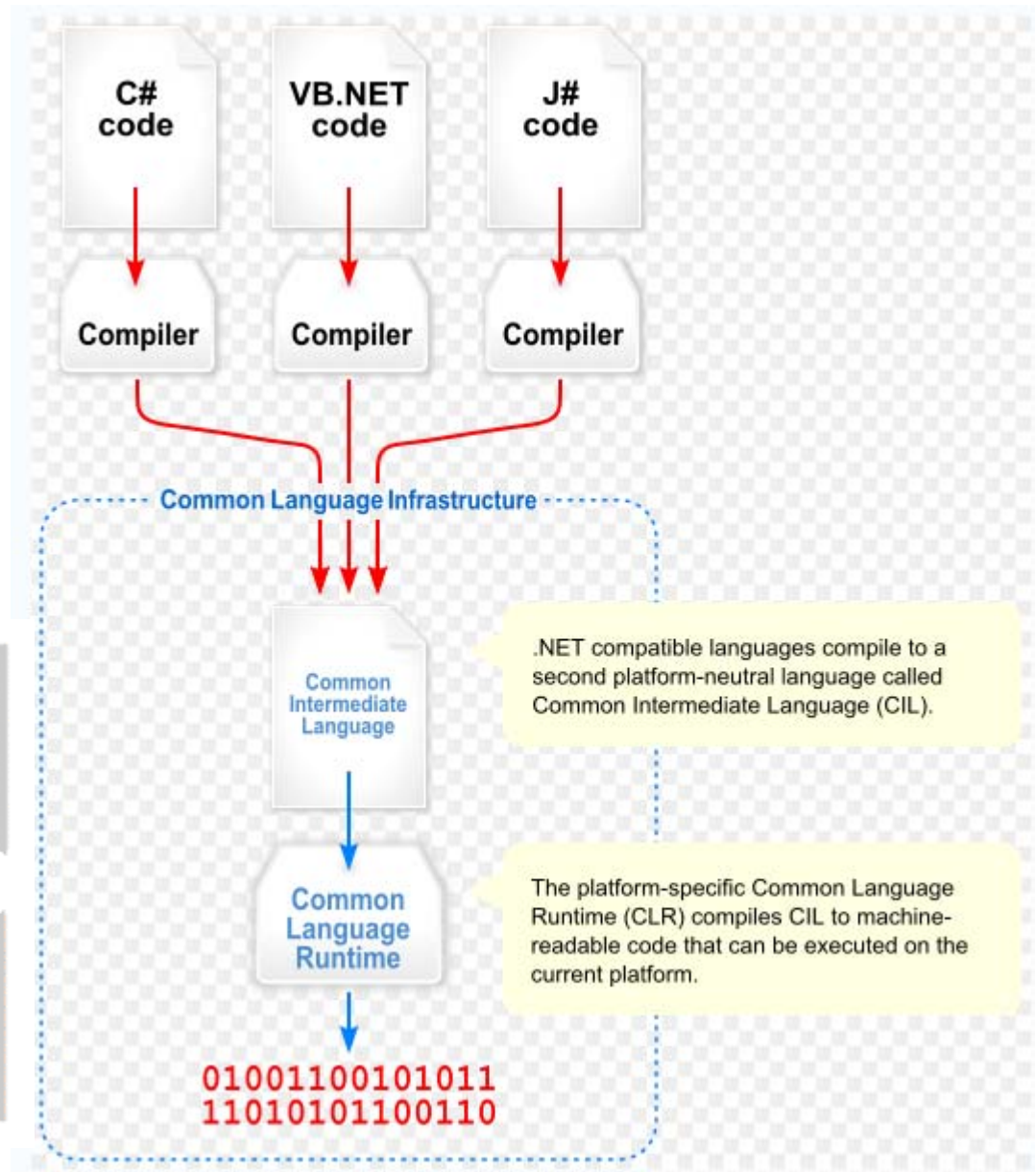
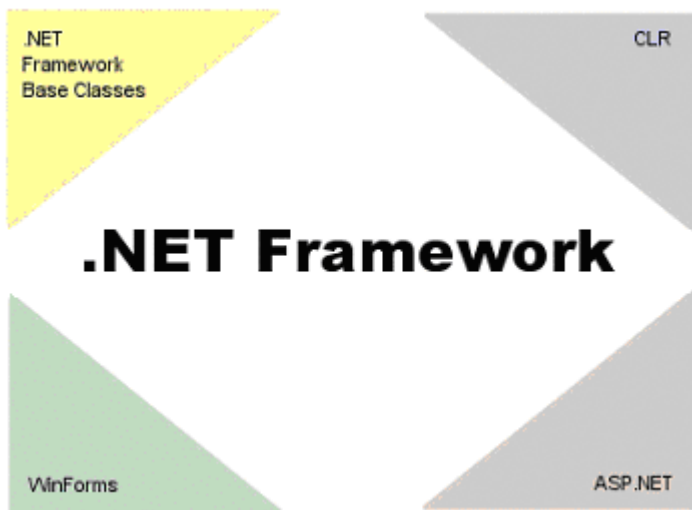
- What is Microsoft .NET
- .NET Expands The Reach Of Oracle
- Oracle and Microsoft
- .NET Helps Developers Follow Oracle Best Practices
- Demonstration
 - DataReader
 - RefCursor
 - Pass / Receive Parameters From Oracle Procedures
 - Disconnected Dataset
 - SQL Injection



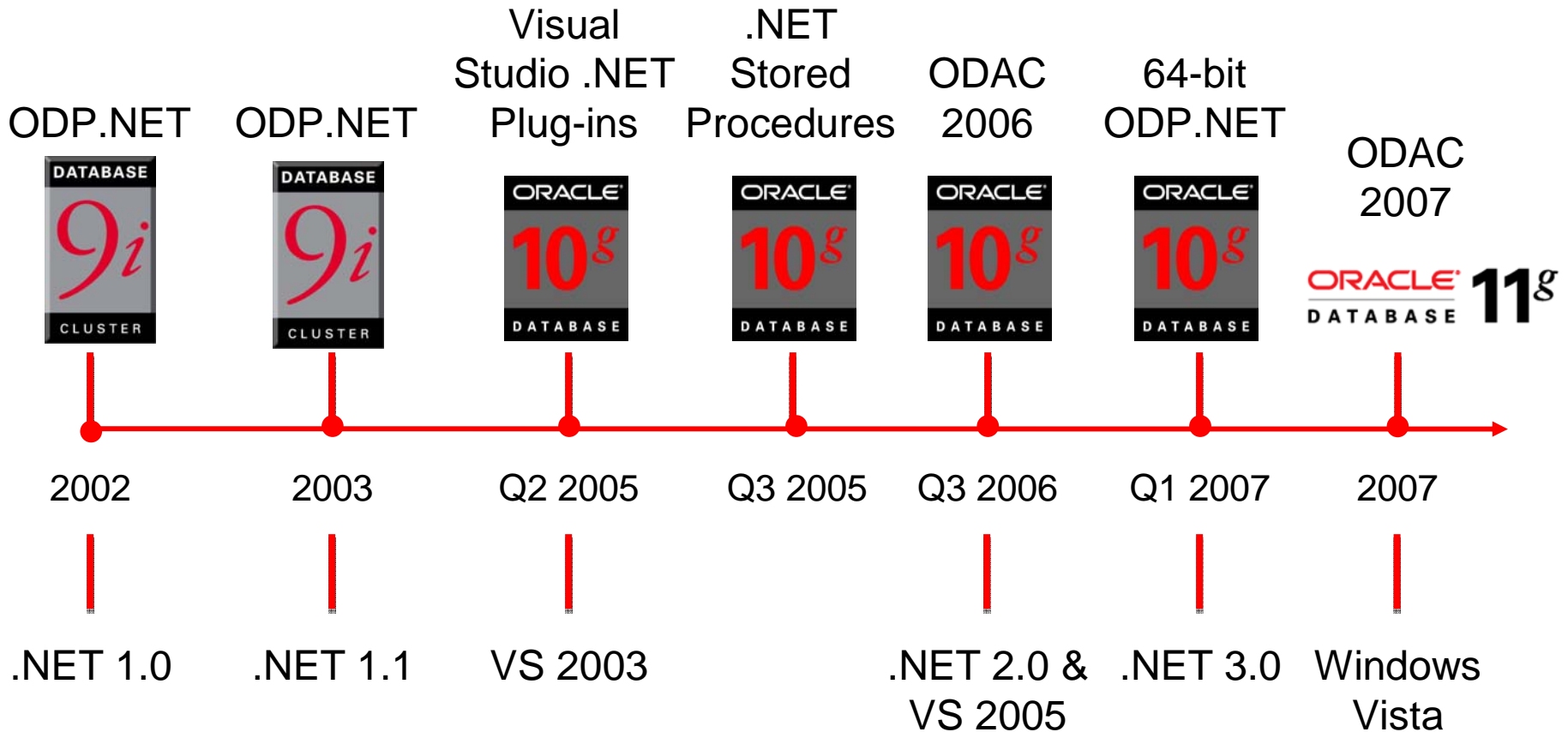
What is Microsoft .NET?

.NET Architecture

- Common Language Runtime
- Large Base Class Library
- Development Environment



Oracle's Commitment to .NET



Using .NET To Expand The Reach Of Oracle

- Expose Oracle Functionality as web service
 - Bring Oracle functionality to .Net with Oracle Web Services Manager.
- Windows Communication Foundation (WCF) – Next Generation of Web Services
 - A,B,C's of WCF
- Rich Browser Development Tools
 - SilverLight
 - AJAX
 - Flash
- **Additional Resources:**
 - 'Making Windows Safe For Unix People' – TechNet
 - (<http://blogs.technet.com/jdzions/59825.aspx>)
 - Essential Windows Communication Foundation
 - Steve Resnick – Addison Wesley



.NET Follows Oracle Best Practices

- Makes It Easy To Let The Database Do Its Job
- Inheritance & OOP Make It Easy For Developers To Follow Standards
- Oracle Data Types Built Into .NET
- Try-catch Architecture Integrates With Oracle Exceptions
- XML Integration
- Large Community Of Users

.NET Access Methods

Disconnected Layer

DataSet

Connected Layer (CLR & Oracle Client)

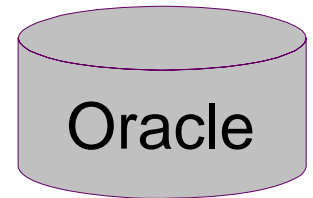
Oracle
DataReader

- Traditional Design (example VB6)
- Very Fast
- Forward Only
- No Row Count

Oracle
DataAdapter

- Simplified Use
- Fill Disconnected Dataset
- Generic Updates

Data Layer



DataReader

```
Private Sub LoadCustomers()  
    Dim cnOra As New OracleConnection(frmMain.strConnectionString)  
    Dim OraCMD As New OracleCommand("", cnOra)  
    Try  
        cmboNames.Items.Clear()  
        OraCMD.CommandType = CommandType.Text  
        OraCMD.CommandText = "Select * from NYOUG_CUSTOMERS Where State ='" _  
                               & cmboStates.SelectedItem & "'"   
  
        cnOra.Open()  
        Dim dr As OracleDataReader = OraCMD.ExecuteReader  
        While dr.Read  
            cmboNames.Items.Add(String.Concat(dr("Lname"), " , ", dr("FNAME")))   
        End While  
        cmboNames.SelectedIndex = 0  
        lblwhatsup.Text = Nothing  
        cnOra.Close()  
    Catch ex As Exception  
        lblwhatsup.Text = ex.Message  
    End Try  
    lblCustomers.Text = "Customers: " & cmboNames.Items.Count  
End Sub
```

Call Stored Procedure

```
Private Sub btnUpdate_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnUpdate.Click
    Try
        Dim cnORA As New OracleConnection(frmMain.strConnectionString)
        Dim oraCMD As New OracleCommand("", cnORA)
        cnORA.Open()
        oraCMD.CommandType = CommandType.StoredProcedure
        oraCMD.CommandText = "affinity.NYOUG_ADDCUST"
        oraCMD.Parameters.Add("inFNAME", OracleType.Char, 30).Value = txtFname.Text
        oraCMD.Parameters.Add("inLNAME", OracleType.Char, 30).Value = txtLname.Text
        oraCMD.Parameters.Add("inADDR1", OracleType.Char, 30).Value = txtAddr1.Text
        oraCMD.Parameters.Add("inCITY", OracleType.Char, 30).Value = txtCity.Text
        oraCMD.Parameters.Add("inSTATE", OracleType.Char, 30).Value = txtState.Text
        oraCMD.Parameters.Add("inZIP", OracleType.Char, 10).Value = txtZip.Text
        oraCMD.Parameters.Add("OutResult", OracleType.Char, 3000).Direction = ParameterDirection.Output
        oraCMD.ExecuteReader()
        lblResult.Text = oraCMD.Parameters("outResult").Value
        cnORA.Close()
    Catch ex As Exception
        MessageBox.Show(ex.Message, "Error")
    End Try
End Sub
```

Oracle Data Adaptor

```
Private Sub SetupAdapter()
```

```
    Dim uCMD As New OracleCommand
    uCMD.Connection = cnOra
    uCMD.CommandType = CommandType.Text
    uCMD.Parameters.Add(":inFNAME", OracleType.Char, 30, "FNAME")
    uCMD.Parameters.Add(":inLNAME", OracleType.Char, 30, "LNAME")
    uCMD.Parameters.Add(":inADDR1", OracleType.Char, 30, "ADDR1")
    uCMD.Parameters.Add(":inCITY", OracleType.Char, 30, "CITY")
    uCMD.Parameters.Add(":inSTATE", OracleType.Char, 30, "STATE")
    uCMD.Parameters.Add(":inZIP", OracleType.Char, 30, "ZIP")
    uCMD.Parameters.Add(":inID", OracleType.Int32, 12, "ID")
    uCMD.CommandText = "Update NYOUG_CUSTOMERS SET FNAME = :inFNAME, LNAME = :inLNAME, ADDR1 = :inADDR1, " & _
                        "CITY = :inCITY, STATE = :inSTATE, ZIP = :inZIP WHERE ID = :inID"
    daCustomers.UpdateCommand = uCMD
```

RefCursor

```
Private Sub btnGetData_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnGetData.Click
    Try
        Dim daOra As New OracleDataAdapter("", cnOra)
        daOra.SelectCommand.CommandType = CommandType.StoredProcedure
        daOra.SelectCommand.CommandText = "AFFINITY.NYOUG_GETCUSTS.GETCUSTS"
        daOra.SelectCommand.Parameters.Add(New OracleParameter("outCursor", OracleType.Cursor)).Direction = _
                                                    ParameterDirection.Output

        daOra.Fill(ds, "Customers")
        dgv.DataSource = (ds.Tables("Customers"))
        dgv.AutoSizeColumnsMode = DataGridViewAutoSizeColumnsMode.DisplayedCells
        dgv.AllowUserToAddRows = False
    Catch ex As Exception
        MessageBox.Show(ex.Message)
    End Try
End Sub
```

SQL Injection

```
Private Sub LoadData(ByVal inFname As String, ByVal inLname As String)
    Dim cmd As New OracleCommand( _
        "Select * from NYOUG_CUSTOMERS Where fname like '" & inFname & _
            "' and lname like '" & inLname & "'", cnOra)

    Try
        cnOra.Open()
        Dim dr As OracleDataReader = cmd.ExecuteReader
        While dr.Read
            dgv.Rows.Add(dr("ID"), dr("FNAME"), dr("LNAME"), dr("ADDR1"), dr("CITY"), _
                dr("STATE"), dr("ZIP"))
        End While
    Catch ex As Exception
        MessageBox.Show(ex.Message)
    End Try
    If cnOra.State = ConnectionState.Open Then
        cnOra.Close()
    End If
End Sub
```



Goals

- Understand What Microsoft .NET Is
- See How .NET Expands The Reach Of Oracle
- Consider Using .NET To Enhance Your Oracle Solutions