

The background of the slide is a photograph of the Oracle New England Development Centre. It features several modern, cylindrical glass buildings with horizontal white bands. The buildings are situated behind a row of weeping willow trees and a rocky shoreline. In the foreground, there is a calm body of water reflecting the sky and the buildings. The sky is clear and blue.

Oracle Rdb Technical Forums

Connecting to Oracle Rdb from .NET

Jim Murray

Oracle New England Development Centre



Agenda

- .NET
- Connectivity Overview
- ADO.NET Overview
- Oracle Data Provider for .NET
- Oracle Rdb Data Provider for .NET



Microsoft .NET

- Connecting applications over the web
 - W3C standardized XML Web Services
 - Loosely-coupled distributed architecture
 - Core set of technologies
 - Make as much use of CLIENT processing power as possible



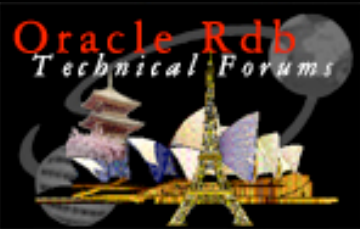
Microsoft .NET (cont.)

- Core components
 - .NET Enterprise Servers
 - .NET Smart Clients
 - XML Web services
 - Developer tools
- Technologies
 - Common Language Runtime (CLR)
 - .NET class libraries
 - ASP.NET

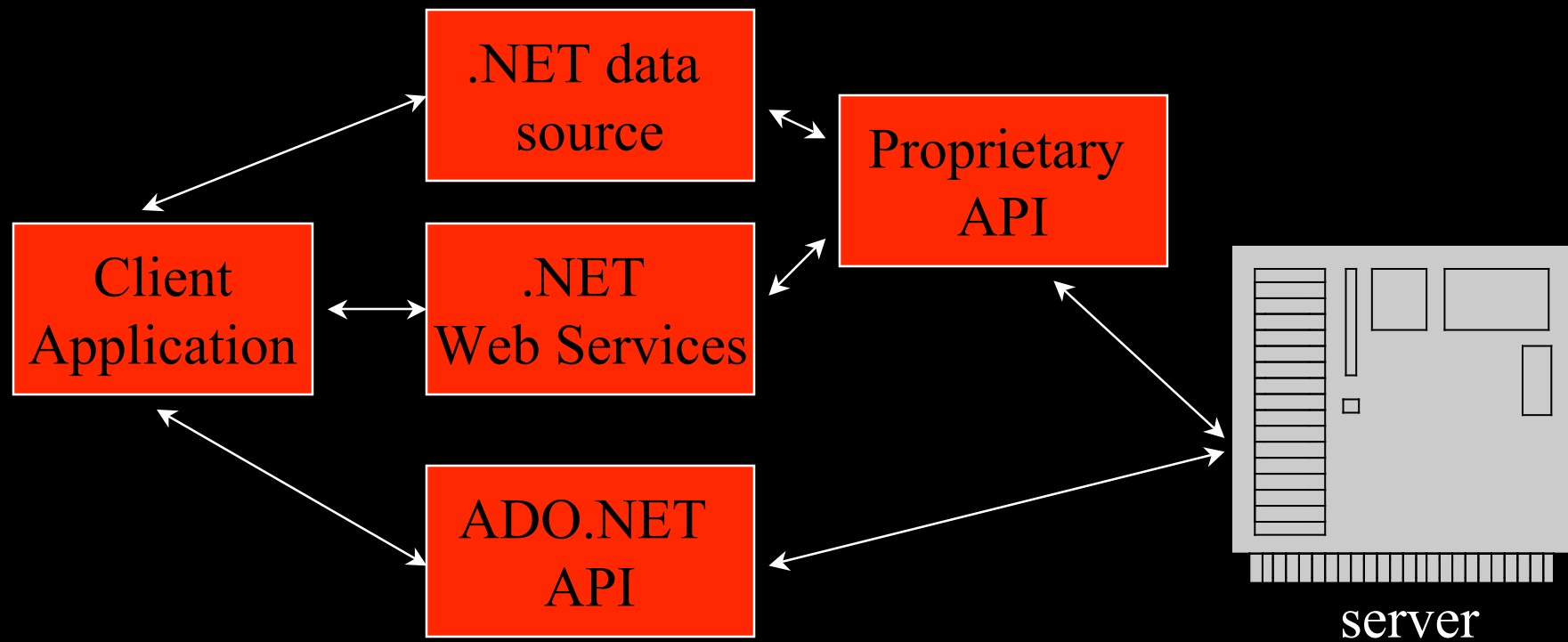


Microsoft .NET (cont.)

- Framework (similar to J2EE)
 - Net(Sockets) , Data (ADO.NET)
 - Namespaces , Collections , Configuration
 - DirectoryServices, EnterpriseServices (COM+)
 - Globalisation ,Reflection, Remoting,
Serialisation,Security,Threading
 - Web,XML



.NET data access





Client Connectivity

- From Open/VMS
 - SQL precompilers/ SQLMOD/Interactive SQL
 - SQL/Services
 - JDBC
- From Desktop
 - SQL/Services
 - ODBC
 - JDBC
 - ADO.NET



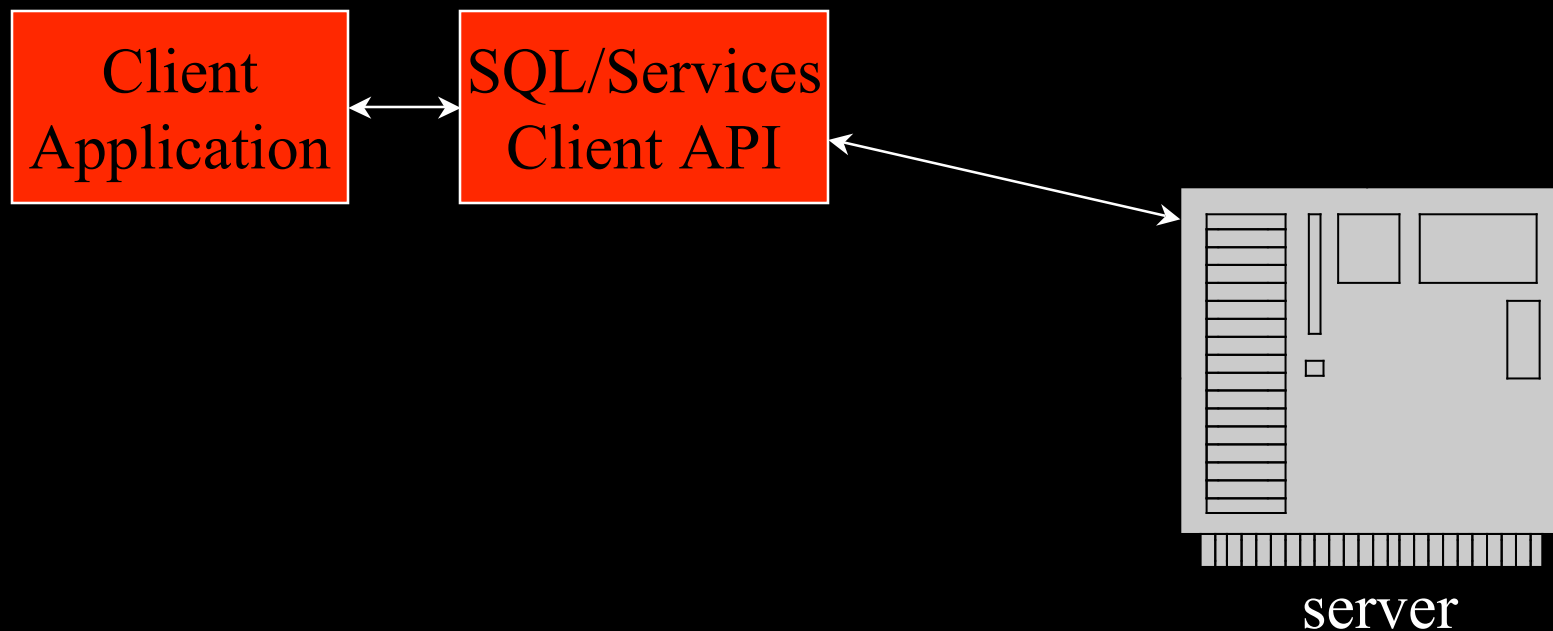
SQL/Services

- SQL/Services API
- ODBC
- OCI Services
- V7.2 now available



SQLSRV Client Architecture

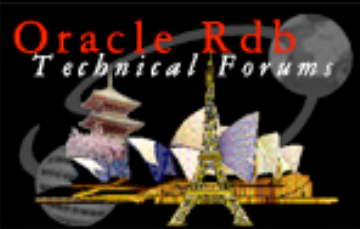
Client using SQL/Services client API directly





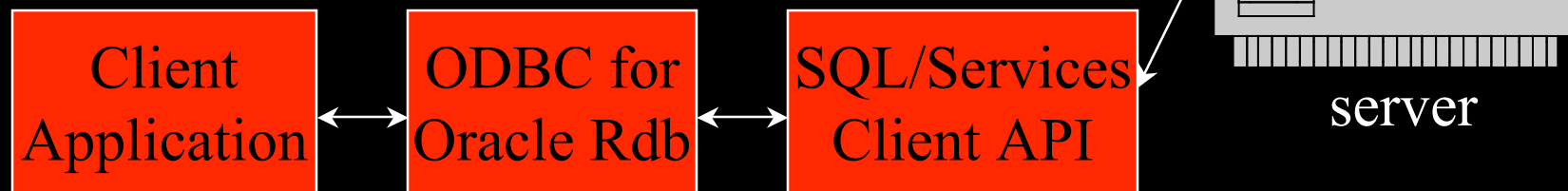
SQL/Services Clients

- OpenVMS
- Microsoft Windows 2000/98/95/NT
- Microsoft Windows XP/ME from V7.2
- Tru64 Unix
- Red Hat Linux from V7.1.6



SQLSRV Client Architecture

Client using ODBC for Oracle Rdb



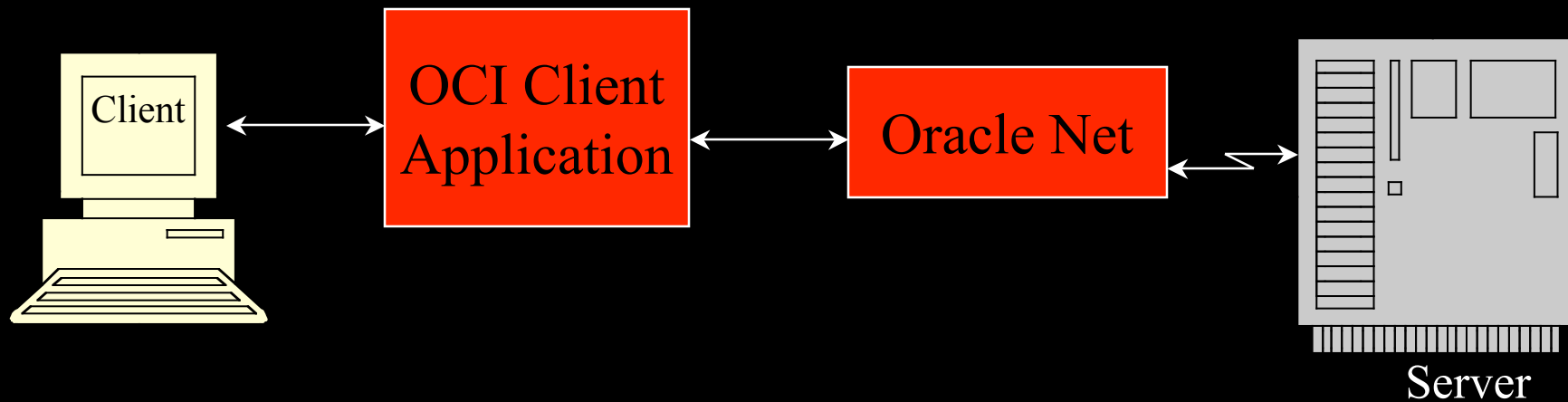


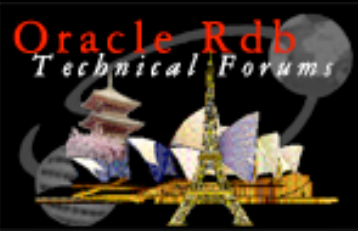
ODBC Clients

- Windows 2000/98/95/NT/XP



OCI Client Architecture



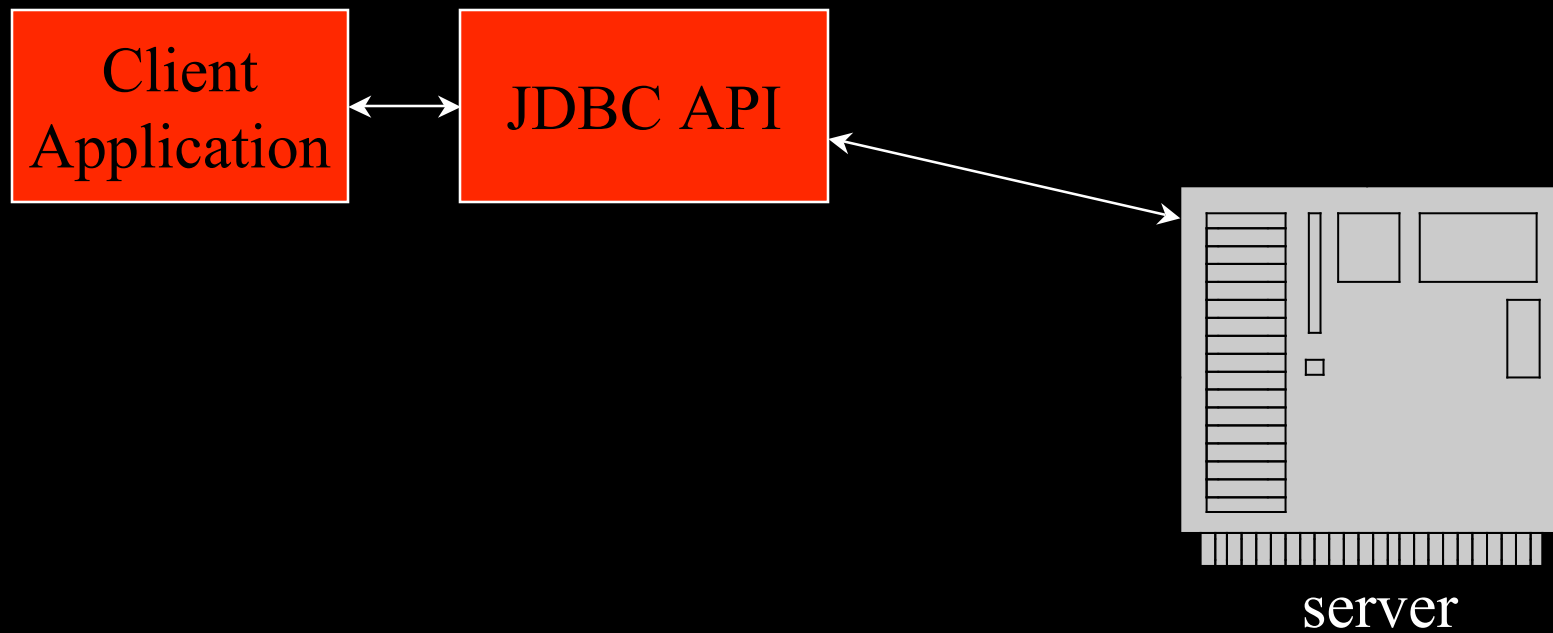


OCI Clients

- Oracle Client
 - IBM AIX-5L
 - HP-UX
 - All Microsoft Windows platforms
 - Linux
 - Solaris
- Oracle Instant Client
 - Clients as above
 - Macintosh OSX
 - Smaller client footprint



JDBC Architecture





JDBC Drivers

Vendor independent

JDBC Class Library

Drivers written by vendor

JavaSoft ODBC

OracleOCI

OracleThin

rdbThin

rdbNative

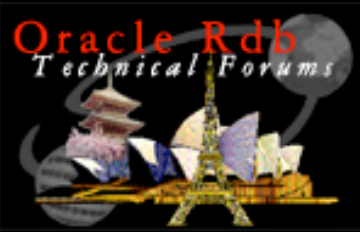
ODBC
C Library

OCI
C Library

Java Sockets

Dynamic SQL

ORACLE



JavaSoftODBC

Vender independent

JDBC Class Library

Drivers written by vender

JavaSoft ODBC

OracleOCI

OracleThin

rdbThin

rdbNative

ODBC
C Library

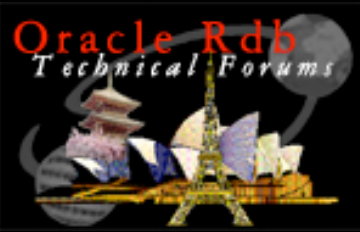
OCI
C Library

Java Sockets

Dynamic SQL

ODBC
SQL/Services

ORACLE



OracleOCI

Vender independent

JDBC Class Library

Drivers written by vender

Javasoft ODBC

OracleOCI

OracleThin

rdbThin

rdbNative

ODBC
C Library

OCI
C Library

Java Sockets

Dynamic SQL

SQL/Services
OCI Services

ORACLE



OracleThin

Vender
independent

JDBC Class Library

Drivers
written by
vender

Javasoft ODBC

OracleOCI

OracleThin

rdbThin

rdbNative

ODBC
C Library

OCI
C Library

Java Sockets

Dynamic SQL

SQL/Services
OCI Services



RdbThin

Vender
independent

JDBC Class Library

Drivers
written by
vender

JavaSoft ODBC

OracleOCI

OracleThin

rdbThin

rdbNative

ODBC
C Library

OCI
C Library

Java Sockets

Dynamic SQL

JDBC ThinServer



RdbNative

Vender independent

JDBC Class Library

Drivers written by vender

Javasoft ODBC

OracleOCI

OracleThin

rdbThin

rdbNative

ODBC
C Library

OCI
C Library

Java Sockets

Dynamic SQL

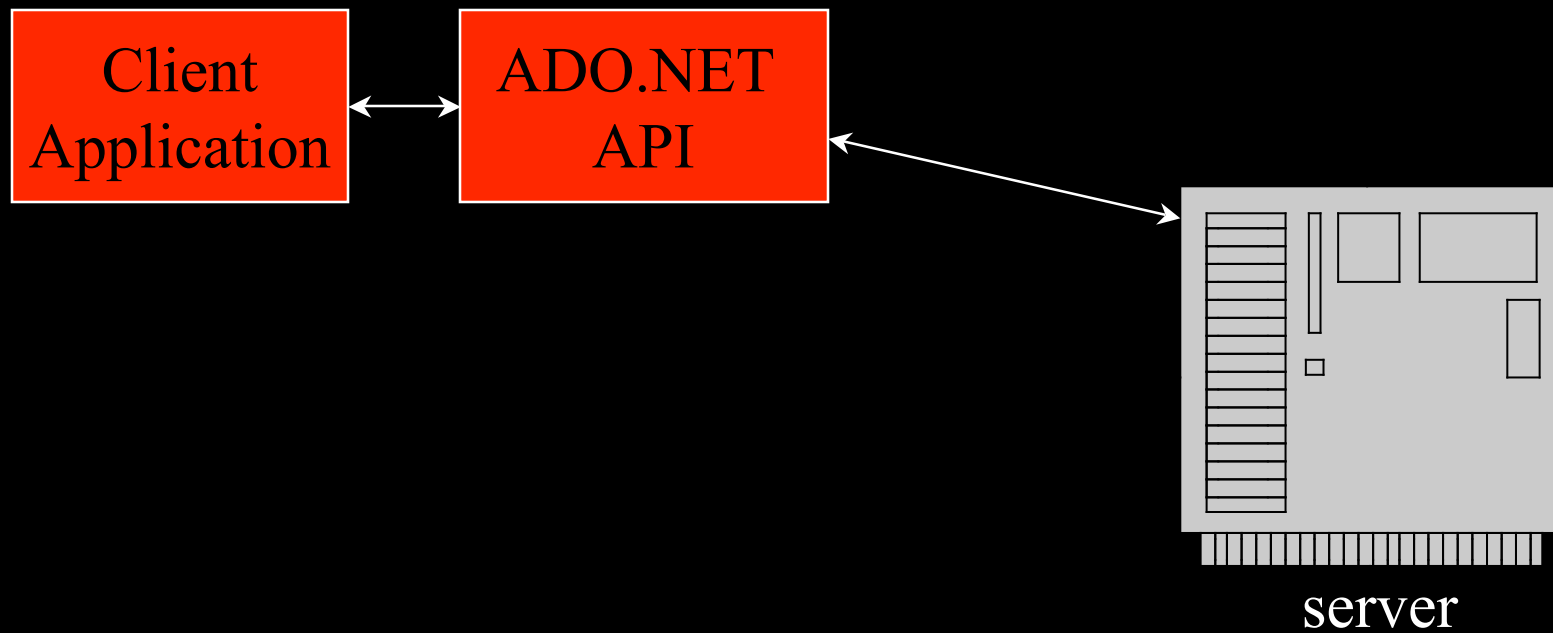


JDBC Clients

- Anywhere JAVA is supported



ADO.NET Architecture





ADO.NET Clients

- Microsoft Windows 2000/98/NT/ME/XP

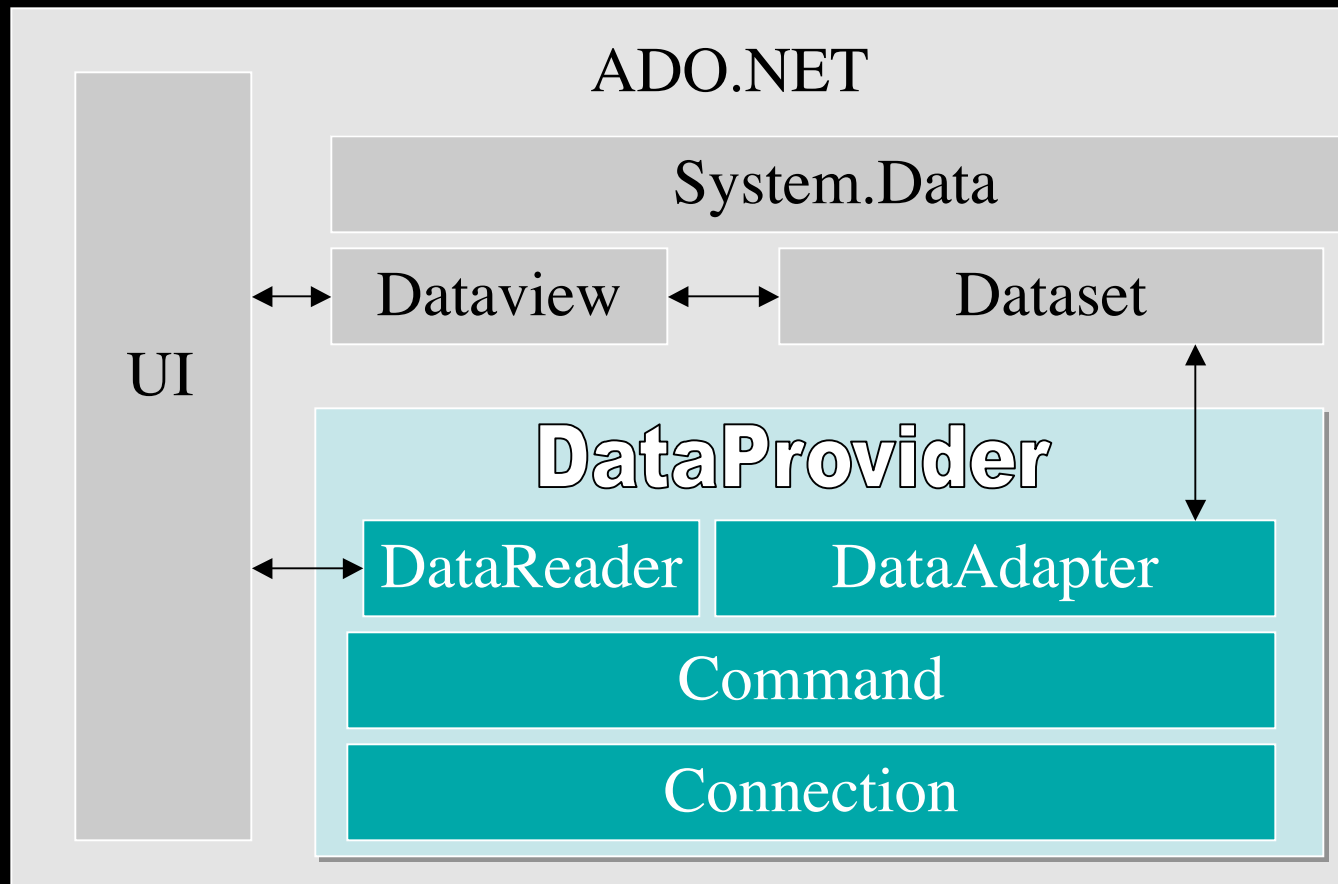


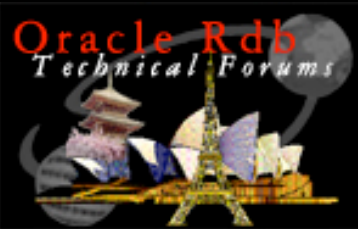
ADO.NET

- Microsoft evolution of the ADO data access model
- Scalable
- Stateless
- XML



ADO .NET





DataSet

- Data Containers (Caches)
- Store data for application use
- Data and Metadata
 - Tables, Rows Columns, Relations
- Populate , Navigate, Update
- Can be disconnected from data source



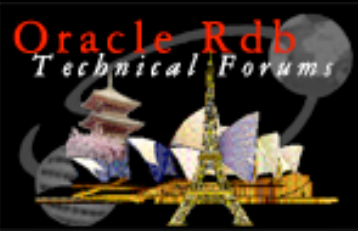
DataView

- designed for use by UI objects
- bound to DataSet
- Customized views of DataSet

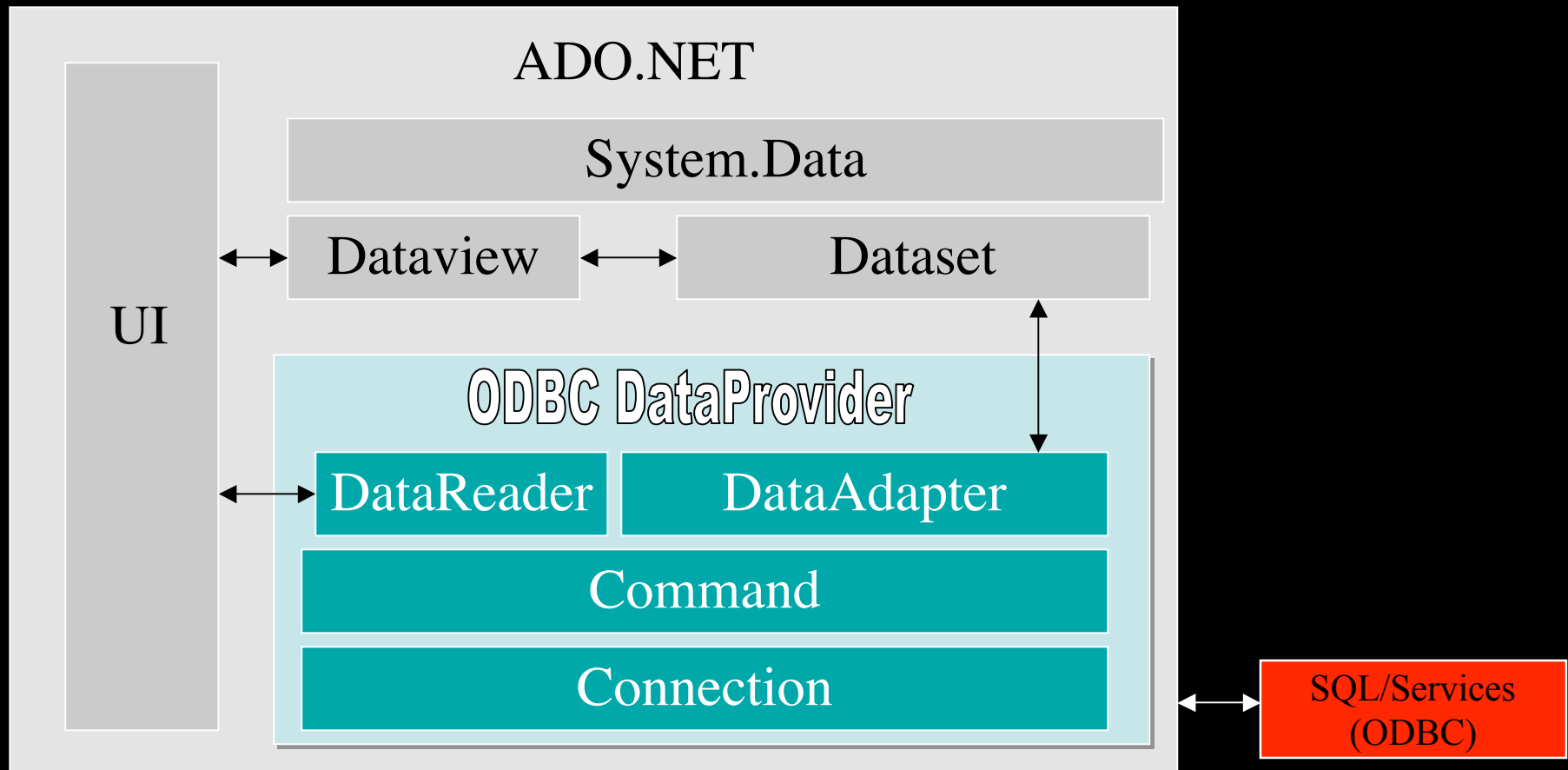


DataProvider

- Connects to the database on behalf of ADO.NET
 - Connection
 - Command
 - DataAdaptor
 - DataReader
- Provided by Microsoft and third party Vendors



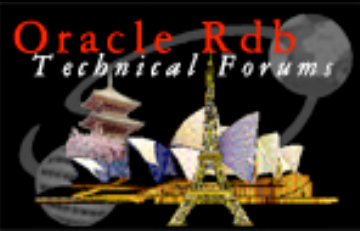
ODBC DataProvider for .NET



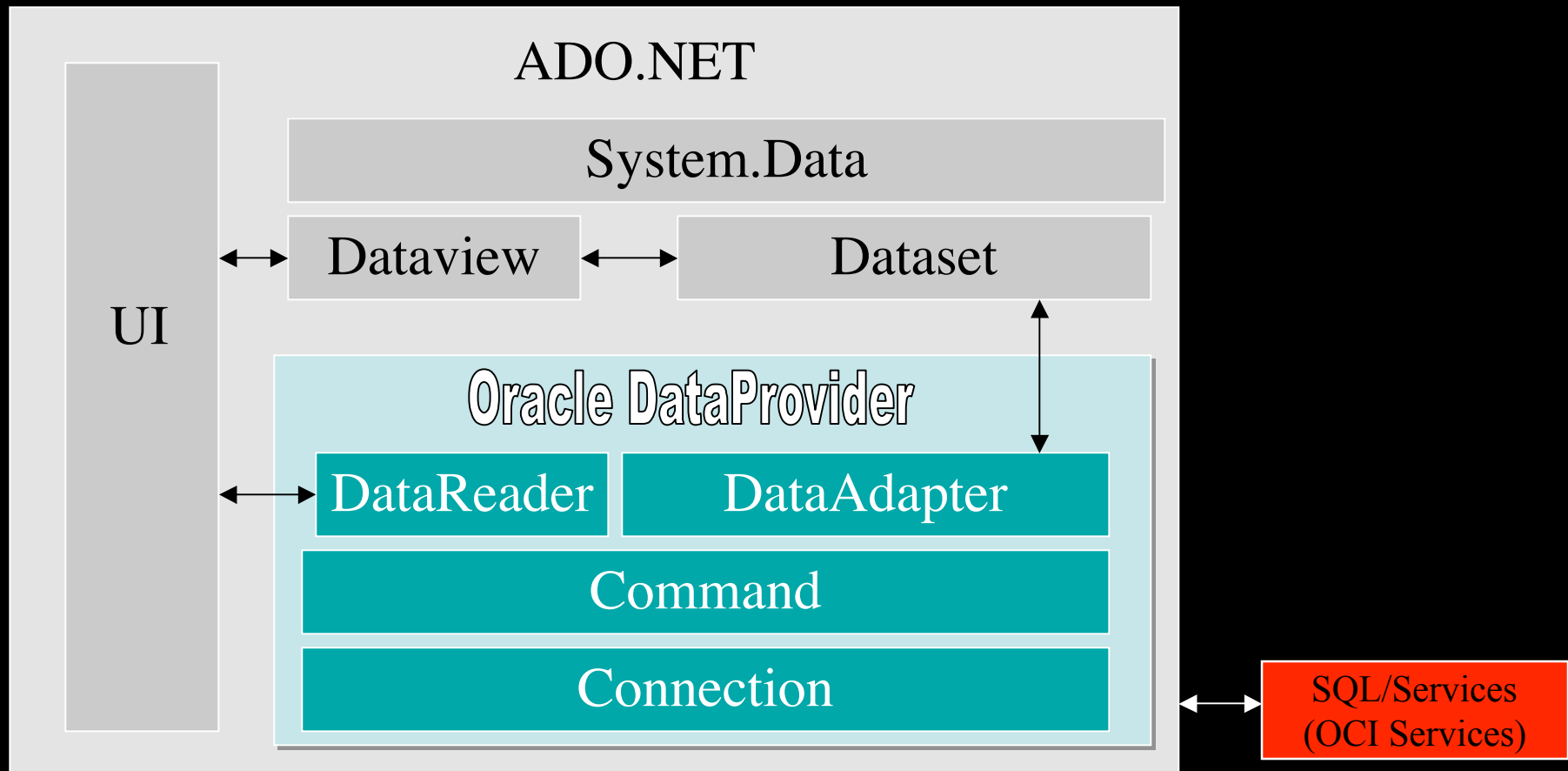


ODBC DataProvider for .NET

- Client Side
 - Standard Microsoft ODBC .NET data provider
- Server Side
 - Access via SQL/Services & ODBC
- Available now



Oracle DataProvider for .NET



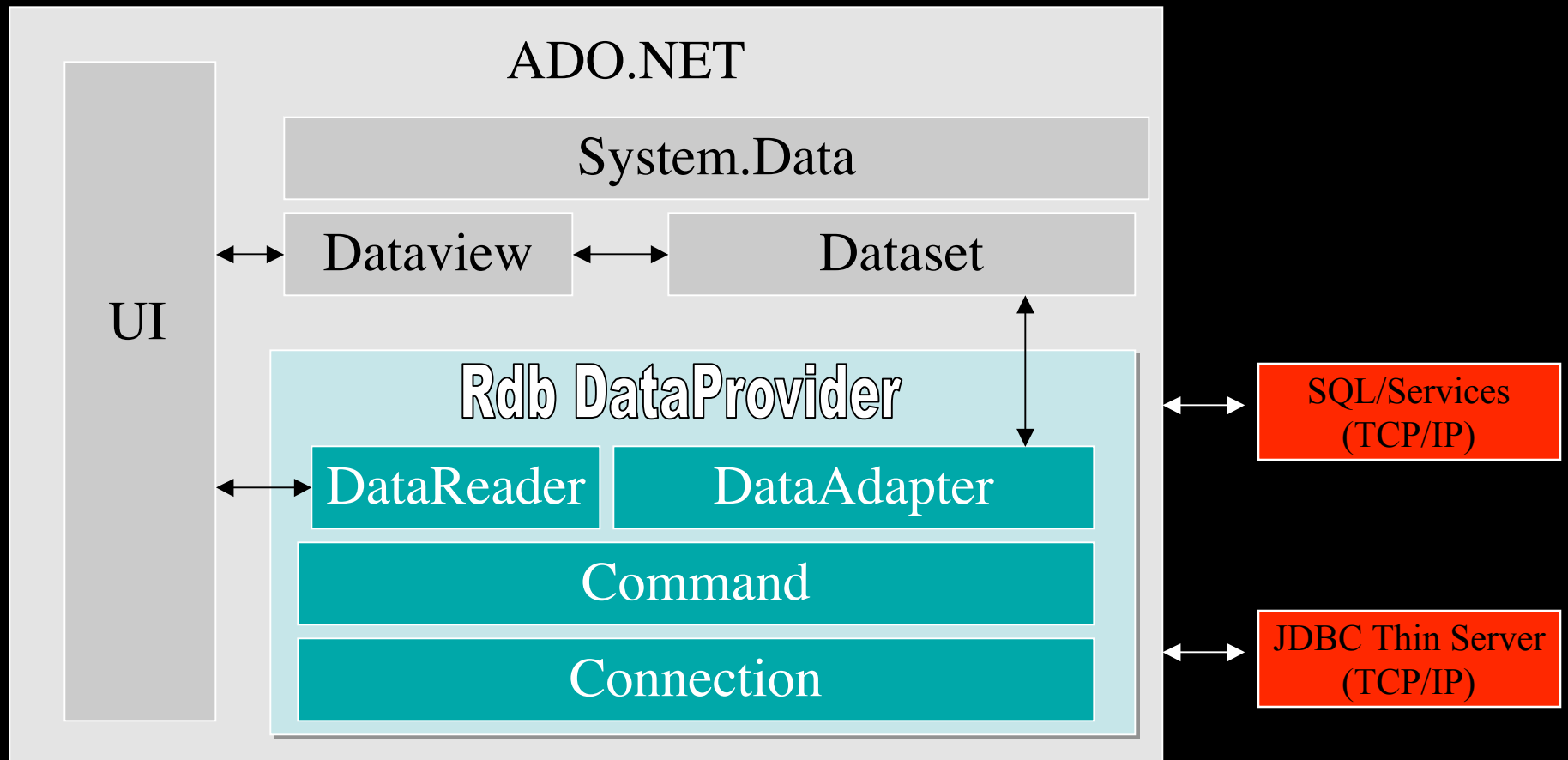


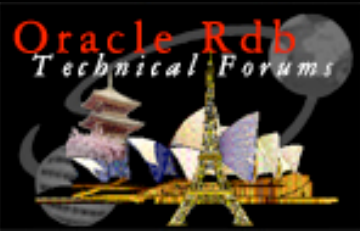
Oracle DataProvider for .NET

- Client Side
 - Standard Oracle .NET data provider
- Server Side
 - Access via SQL/Services OCI Service
- Available now



Oracle Rdb DataProvider for .NET



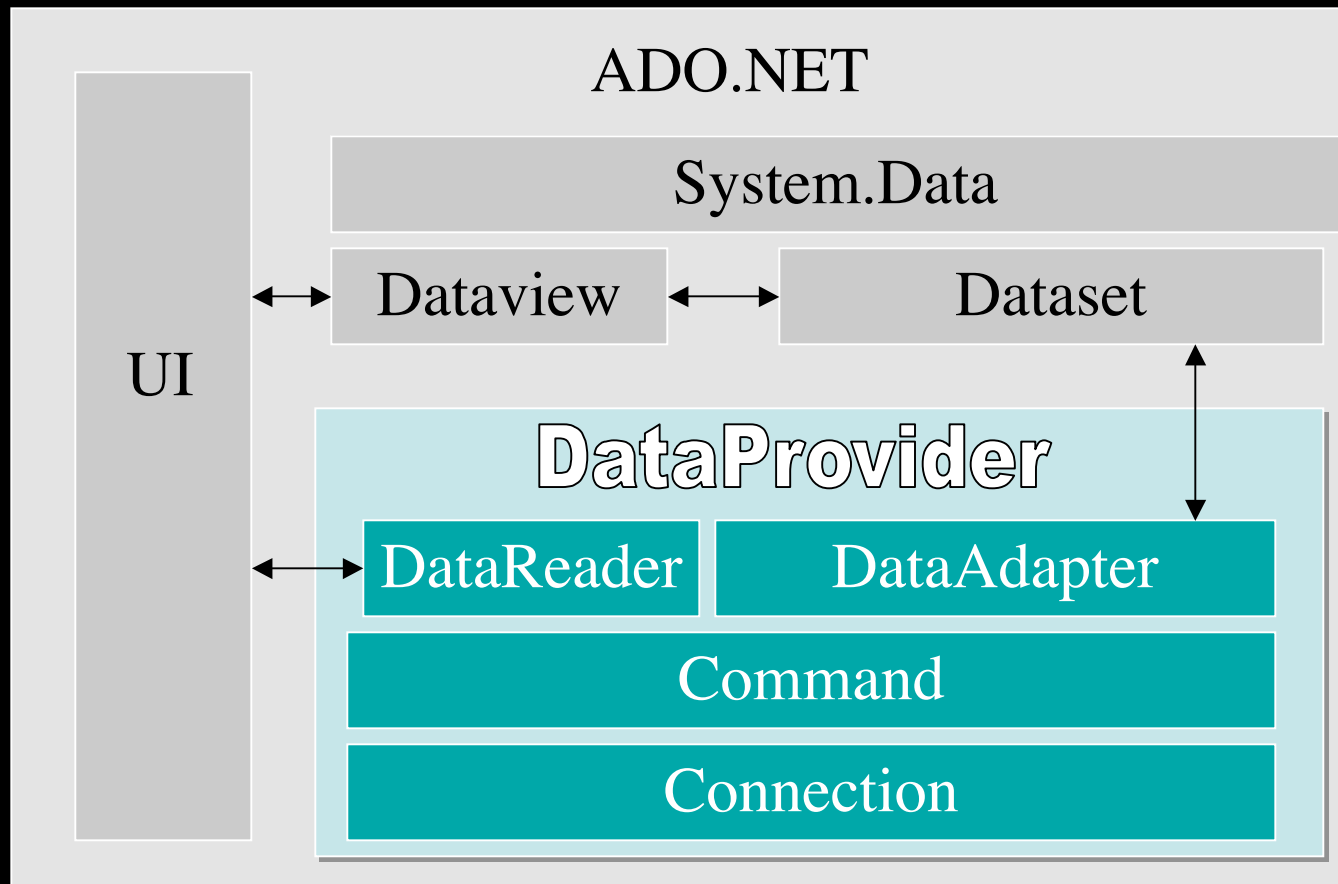


Oracle Rdb DataProvider for .NET

- Client Side
 - Standard .NET data provider API
 - Supplied as Installable library
- Server Side
 - Access via SQL/Services or
 - Access via Oracle JDBC for Rdb Thin Servers
- Beta test Q4 CY06

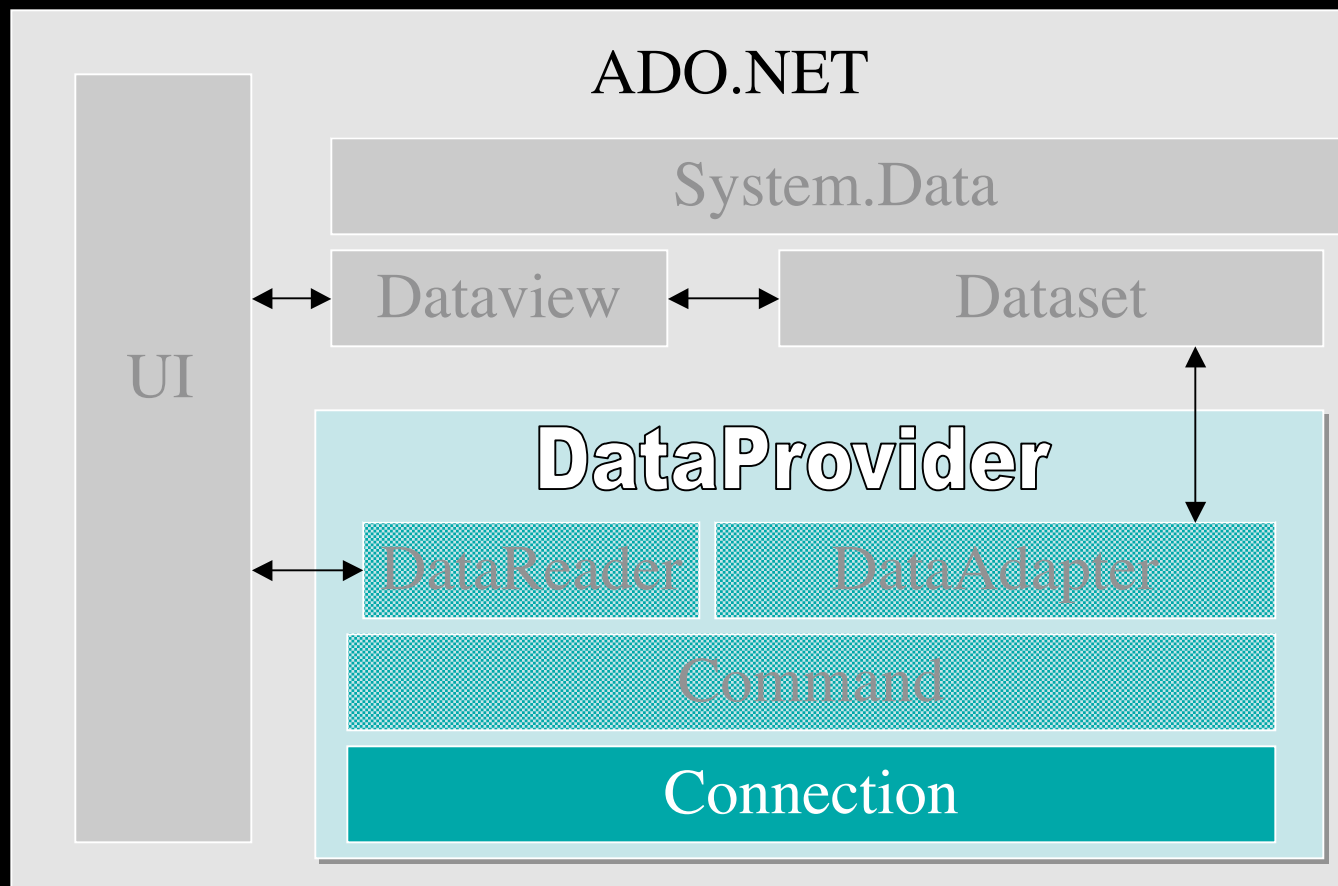


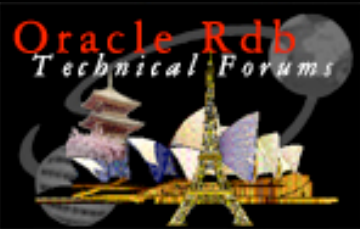
DataProvider for .NET





DataProvider for .NET





CONNECTION

- Opens up a connection to the data source
- Explicit
 - `conn.OPEN()`
- Implicit
 - When using a DataAdapter



CONNECTION C# example

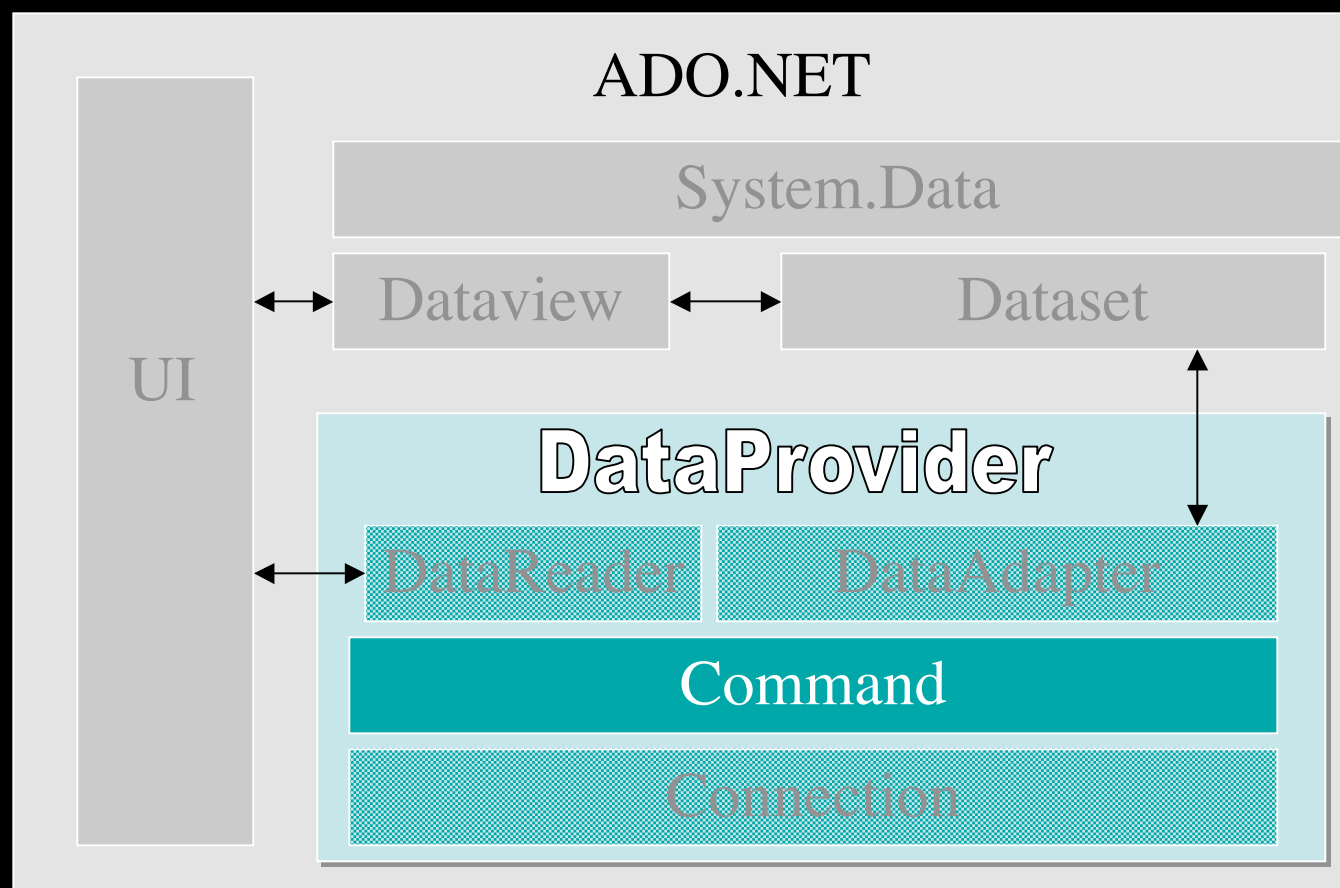
```
OracleConnection myConn = new OracleConnection  
    ("DataSource=myEMP; UserId=netUser ; Password=  
    netUser",
```

```
RdbConnection myConn = new RdbConnection  
    ("Server=localhost:1701;  
    Database=diske:[regtest]mf_personnel;  
    User ID=netUser; Pwd=netUser;");
```

```
myConn.Open();
```



DataProvider for .NET





COMMAND

- Issues command to the underlying data source
- Creates results stream
- methods
 - ExecuteReader
 - ExecuteNonQuery
 - ExecuteScalar



COMMAND C# example

```
String sql = "select * from employees";
```

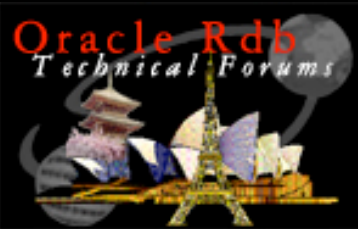
```
OracleCommand cmd = new  
    OracleCommand(sql,myConn) ;
```

```
RdbCommand cmd = new  
    RdbCommand(sql,myConn) ;
```

```
OracleDataReader dr = cmd.ExecuteReader();
```

```
RdbDataReader dr = cmd.ExecuteReader();
```

```
// process the resultset(s) here
```



ExecuteReader C#

```
String sql =
```

```
    "select lastname from employees where employee_id = :id",conn);
```

```
OracleCommand command = new OracleCommand(sql, conn);
```

```
RdbCommand command = new RdbCommand(sql, conn);
```

```
command.Parameters.Add("id", "00165");
```

```
command.Prepare();
```

```
OracleDataReader reader = command.ExecuteReader();
```

```
RdbDataReader reader = command.ExecuteReader();
```

```
while(reader.Read()){
```

```
    Console.WriteLine("name "+ reader.GetString(0) );
```

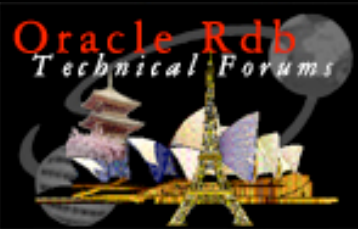
```
}
```

```
reader.Close();
```



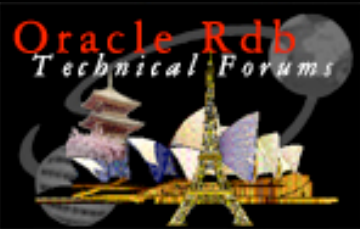
ExecuteNonQuery C++

```
void insertDepartment( RdbConnection* conn, String* deptNbr, String*
    deptName )
{
    RdbCommand* cmd = new RdbCommand();
    cmd->Connection = conn;
    try {
        cmd->CommandText = String::Format( S"insert into  departments (
            department_id, department_name ) values( {0}, '{1}' )", deptNbr,
            deptName );
        cmd->CommandType = CommandType::Text;
        cmd->ExecuteNonQuery();
    } catch(RdbException* e1 )
    { Console::WriteLine( "Insert error: {0}",e1->Message->ToString() ); }
    cmd->Dispose();
}
```

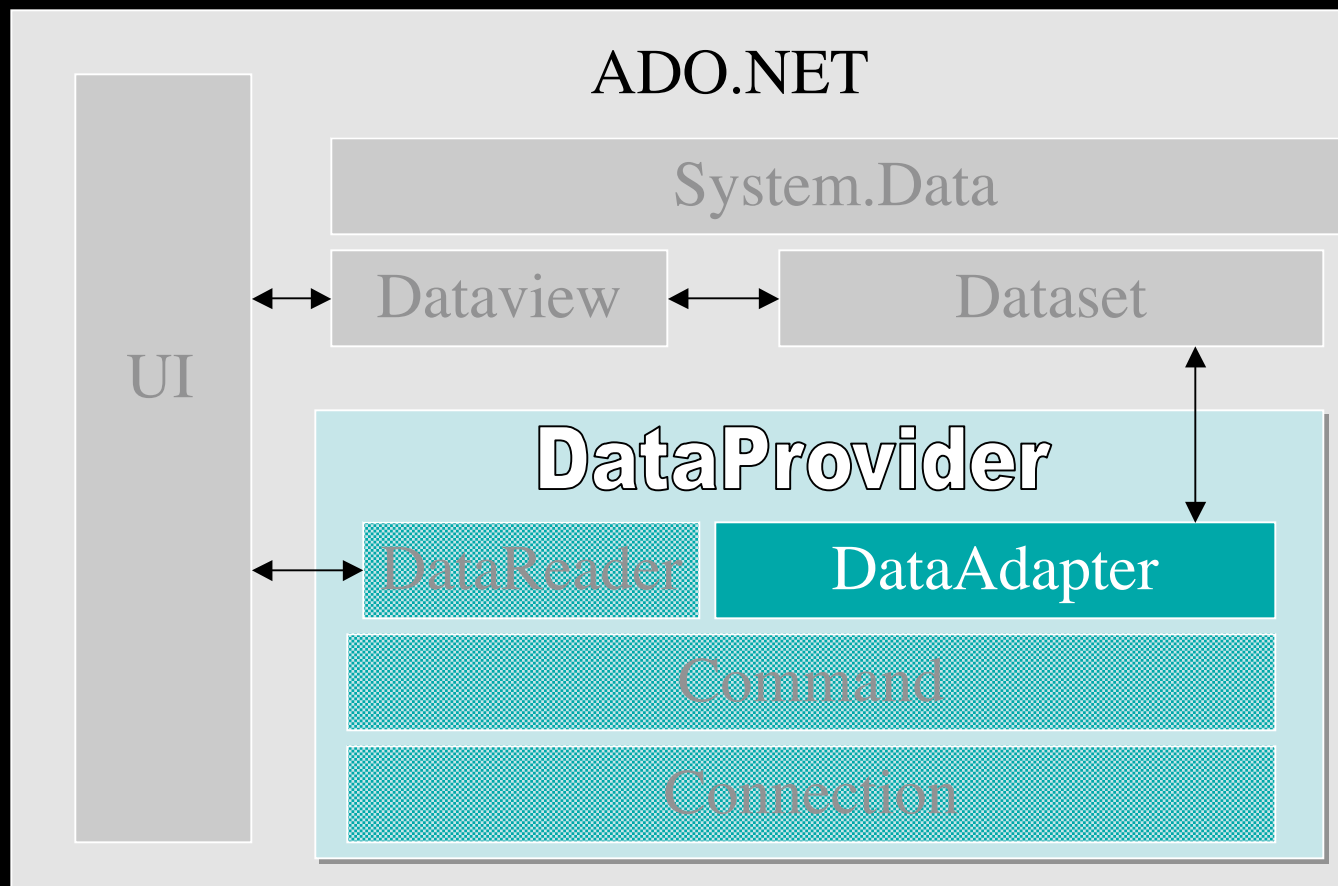


ExecuteScalar C++

```
void countDepartments( OracleConnection* conn )
{
    RdbCommand * cmd = new OracleCommand();
    cmd->Connection = conn;
    try {
        cmd->CommandText = S"select count(*) from
departments";
        cmd->CommandType = CommandType::Text;
        Object* numberOfDepartments = cmd->ExecuteScalar();
        Console::WriteLine( S"Number of Departments: {0}",
            numberOfDepartments );
    } catch ( OracleException* e1 )
    { ...
```



DataProvider for .NET



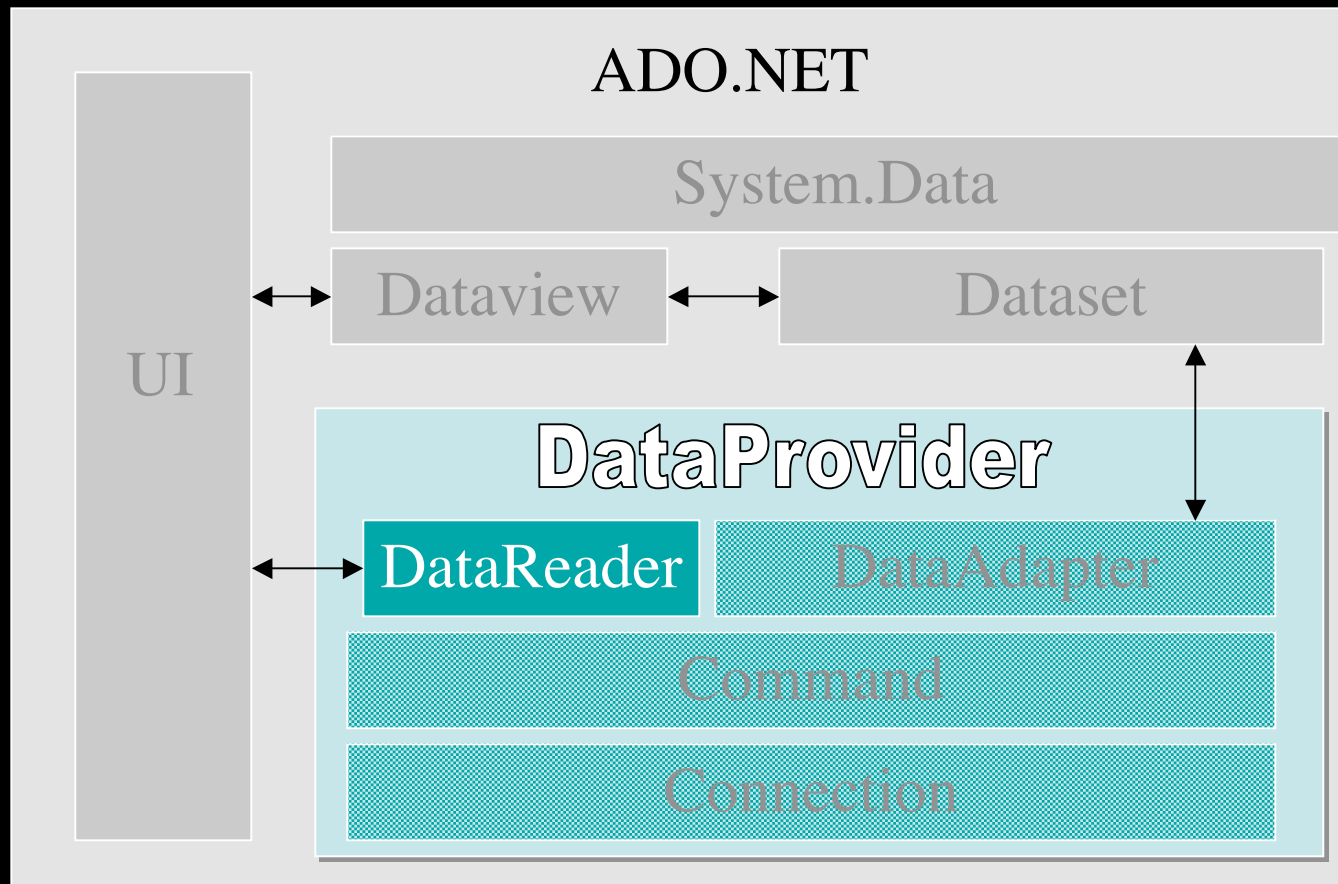


DataAdapter

- Methods + Properties to associate DataSet with data source
- Retrieve data from data source
- Save data into data source
- Connect to DB when it needs to fill DataSet or update DB from DataSet changes



DataProvider for .NET





DataReader

- For large amounts of read only data
- read-only, forward-only stream returned from the database
- Only one record at a time is ever present in memory



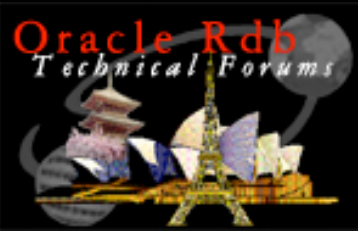
Using Oracle Data Provider

- **SERVER SIDE**
 - Install and setup **Oracle SQL/Services**
 - Create OCI Service
- **CLIENT SIDE**
 - Install **Oracle 10g Data Provider for .NET**
 - Define OCI service in **TNSNAMES.ORA**



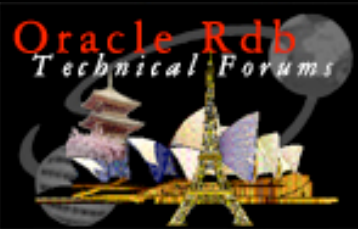
TNSNAMES.ORA

```
MY_MFPER =  
(DESCRIPTION =  
  (ADDRESS =  
    (PROTOCOL = TCP)  
    (HOST = MYAXP.MYCOMPANY.COM)  
    (PORT = 1521) )  
  (CONNECT_DATA =  
    (SERVER = DEDICATED)  
    (SERVICE_NAME = MFPER) )  
)
```



Using Oracle Rdb Data Provider

- **SERVER SIDE – SQL/Services**
 - Install and setup Oracle SQL/Services
 - Create Service
- **SERVER SIDE – Thin Server**
 - Install Oracle JDBC for Rdb
 - Start server
- **CLIENT SIDE**
 - Install Oracle Rdb Data Provider for .NET



For More Information

- [White paper on Rdb and .NET](#)
- www.oracle.com/rdb
- metalink.oracle.com
- <http://www.hp.com/products/openvms>
- Jim.murray@oracle.com

Q U E S T I O N S
&
A N S W E R S

ORACLE®

ORACLE®



.NET example

```
void insertDepartment( OracleConnection* conn, String* deptNbr, String*
    deptName )
{
    OracleCommand* cmd = new OracleCommand();
    cmd->Connection = conn;
    try {
        cmd->CommandText = String::Format( S"insert into departments (
            department_id, department_name ) values( {0}, '{1}' )", deptNbr,
            deptName );
        cmd->CommandType = CommandType::Text;
        cmd->ExecuteNonQuery();
    } catch(OracleException* e1 )
    { Console::WriteLine( "Insert error: {0}",e1->Message->ToString() ); }
    cmd->Dispose();
}
```