

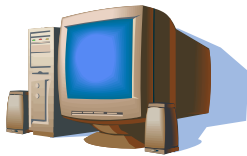
ADO.NET

Nội Dung

- **ADO.NET**
- Quá trình phát triển
- Đặc điểm ADO.NET
- .NET Data Provider
- DataSet
- Data Source Configuration Wizard

Giới thiệu ADO.NET

- ▶ ActiveX Data Object .NET (ADO.NET)
 - Công nghệ của MS
 - Phát triển từ nền tảng ADO
 - Cung cấp các **lớp đối tượng** và **hàm thư viện** phục vụ cho việc kết nối và xử lý dữ liệu



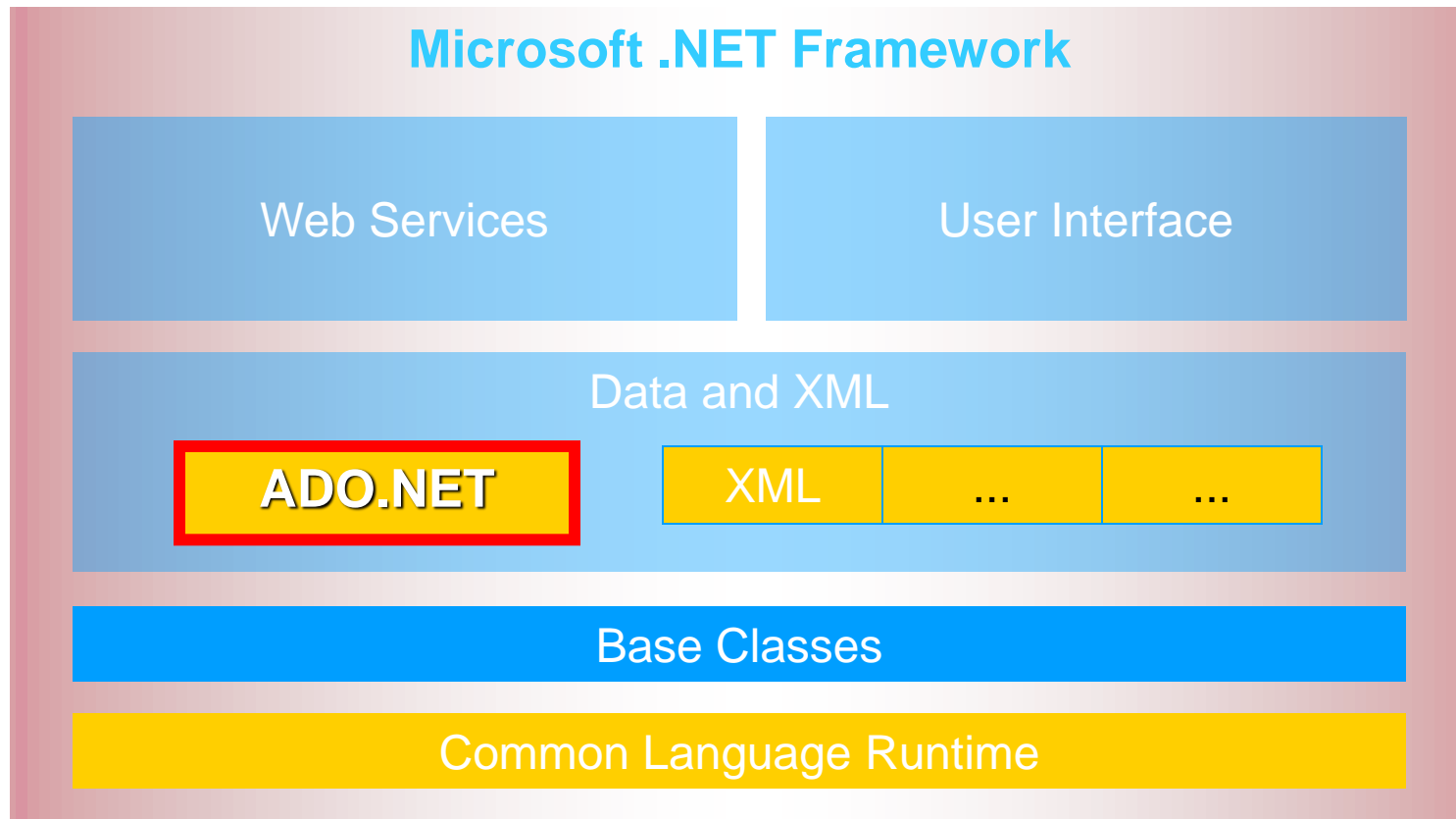
.NET Application

ADO.NET



Giới thiệu ADO.NET

► Mô hình .NET Framework

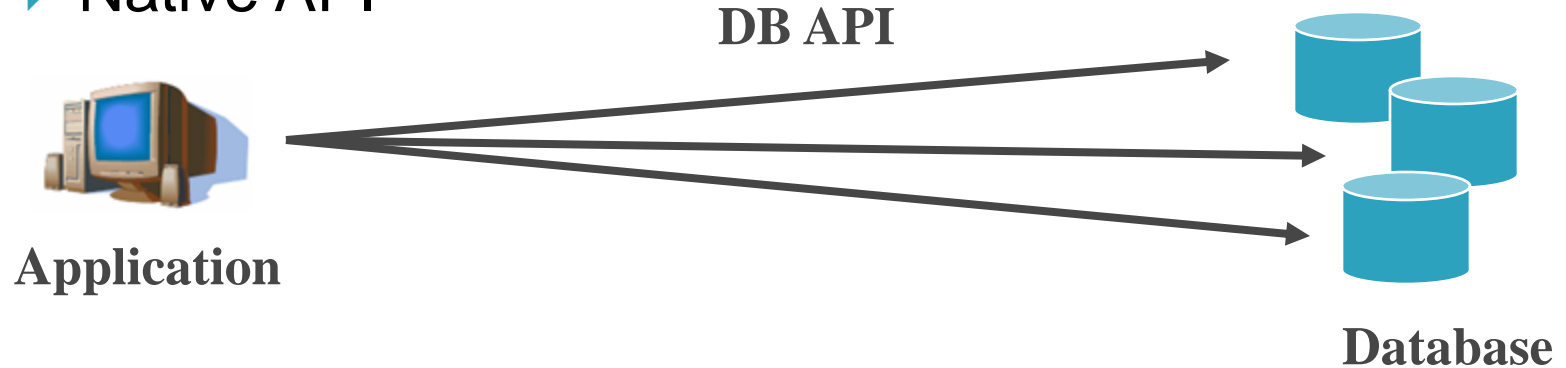


Nội Dung

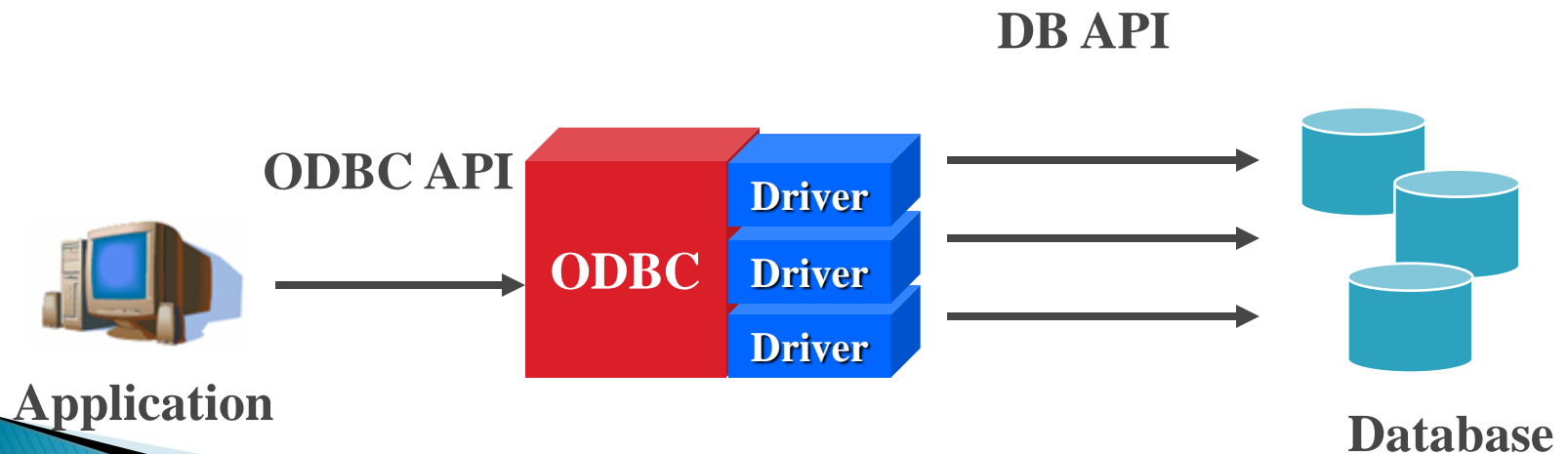
- ADO.NET
- **Quá trình phát triển**
- Đặc điểm ADO.NET
- .NET Data Provider
- DataSet
- Data Source Configuration Wizard

Quá trình phát triển²

► Native API

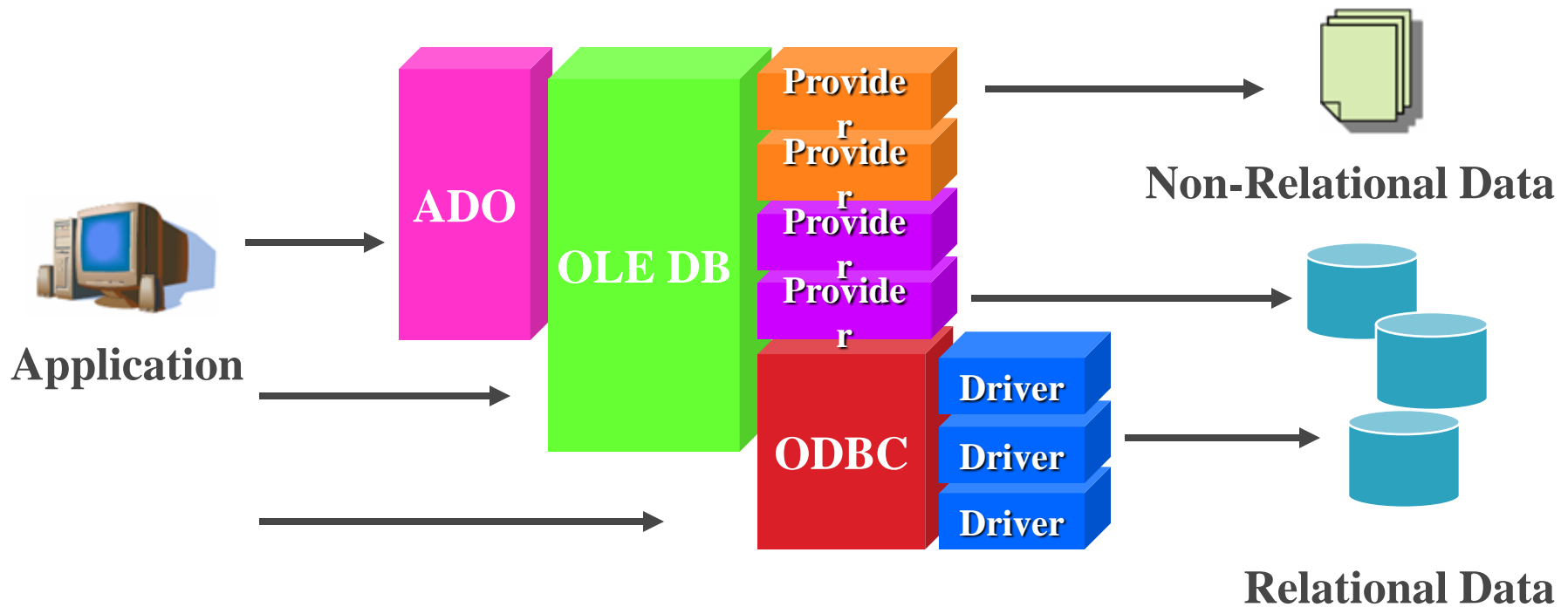


► ODBC



Quá trình phát triển²

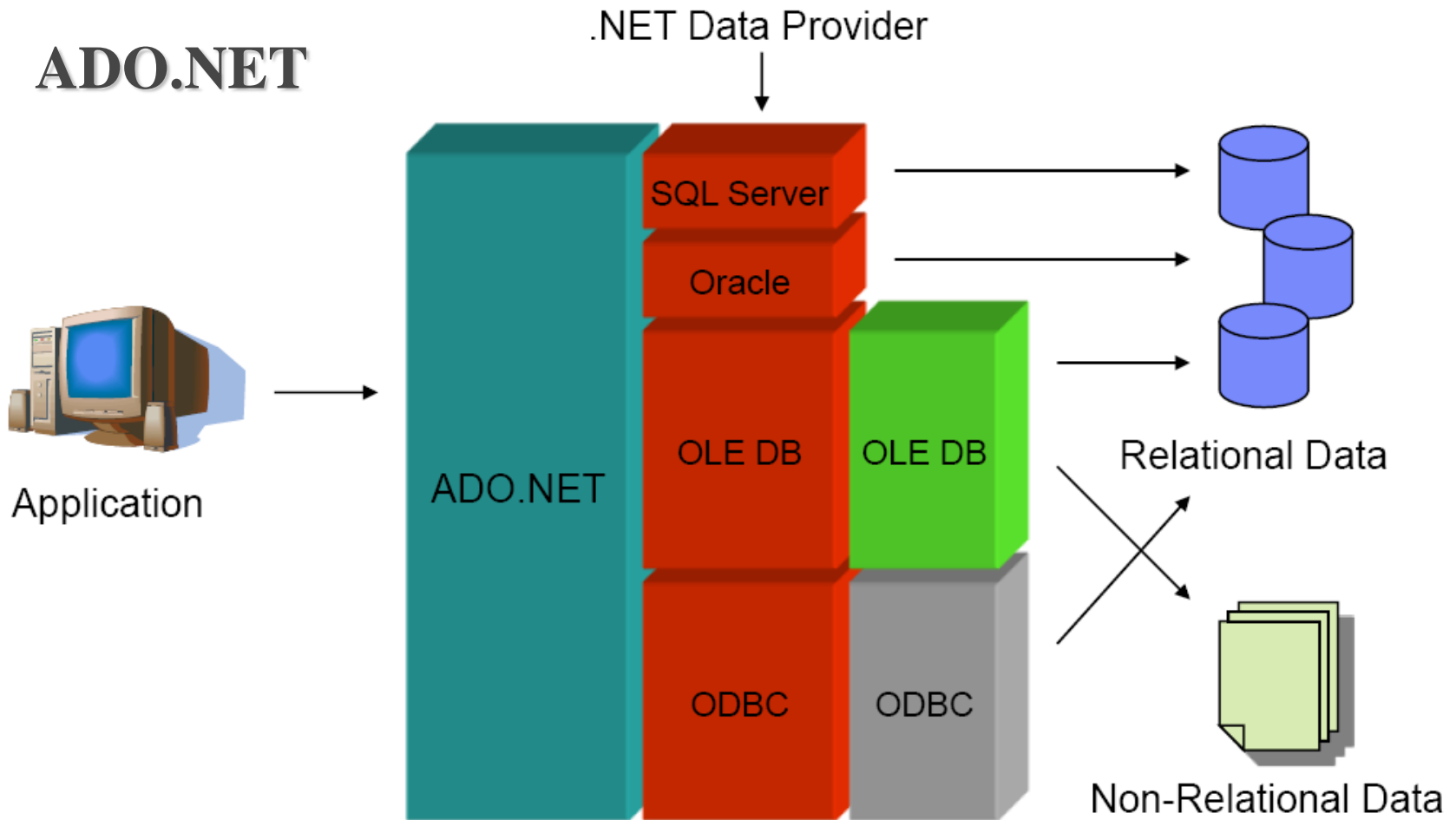
▶ ADO, OLE DB



OLE: Object Linking and Embedding

Quá trình phát triển²

ADO.NET



Nội Dung

- ADO.NET
- Quá trình phát triển
- **Đặc điểm ADO.NET**
- .NET Data Provider
- DataSet
- Data Source Configuration Wizard

ADO.NET

- ▶ ADO.NET là một phần của .NET Framework
 - Thư viện lớp có chức năng **xử lý dữ liệu** trong ngôn ngữ MS.NET
- ▶ ADO.NET là dạng “*Disconnected*”
 - Cho phép lấy cả một cấu trúc phức tạp của DL từ CSDL, sau đó ngắt kết nối rồi mới thực hiện thao tác xử lý
 - Trước đây ADO luôn phải duy trì kết nối trong quá trình thiết kế

ADO.NET

- ▶ ADO.NET mạnh mẽ
 - Kế thừa các ưu điểm của ADO
 - Kết hợp với ý tưởng thiết kế hoàn toàn mới
- ▶ Đặc điểm nổi bật
 - Thiết kế hoàn toàn dựa vào XML
 - Chuẩn giao tiếp dữ liệu tốt nhất trên môi trường Internet hiện nay
 - Thiết kế hoàn toàn hướng đối tượng
 - Đặc trưng của thư viện .NET Framework

ADO.NET vs. ADO

Đặc Điểm	ADO	ADO.NET
DL xử lý được đưa vào bộ nhớ dưới dạng	Recordset : tương đương 1 bảng dữ liệu trong database	Dataset : tương đương 1 database
Duyệt dữ liệu	Recordset chỉ cho phép duyệt tuần tự, từng dòng một.	Dataset : duyệt “tự do, ngẫu nhiên”, truy cập thẳng tới bảng, dòng ,cột mong muốn.
Dữ liệu ngắt kết nối	Recordset thiên về hướng kết nối, nên việc hỗ trợ ngắt kết nối không mạnh	Dataset hỗ trợ hoàn toàn ngắt kết nối
Trao đổi dữ liệu qua Internet	Khả năng trao đổi dữ liệu ADO qua Internet thường có nhiều hạn chế. Do dùng chuẩn COM	ADO.NET trao đổi dữ liệu qua Internet rất dễ dàng vì ADO.NET được thiết kế theo chuẩn XML, là chuẩn dữ liệu chính được sử dụng để trao đổi trên Internet.

Môi trường “connected”

- ▶ Mỗi user có một kết nối cố định tới data source
- ▶ Ưu điểm
 - Môi trường được bảo vệ tốt
 - Kiểm soát được sự đồng bộ
 - Dữ liệu luôn được mới
- ▶ Nhược
 - Phải có một kết nối mạng cố định
 - Scalability

Môi trường “disconnected”

- ▶ Một tập con của dữ liệu trung tâm được sao chép và bổ sung độc lập, sau đó sẽ được merge lại vào dữ liệu trung tâm.
- ▶ Ưu điểm
 - Có thể làm việc bất cứ lúc nào, cũng như có thể kết nối bất kỳ vào Data Source
 - Cho phép user khác có thể kết nối
 - Nâng cao hiệu suất thực hiện của ứng dụng
- ▶ Nhược điểm
 - Dữ liệu không được cập nhật một cách nhanh nhất
 - Sự tranh chấp có thể xuất hiện và phải giải quyết

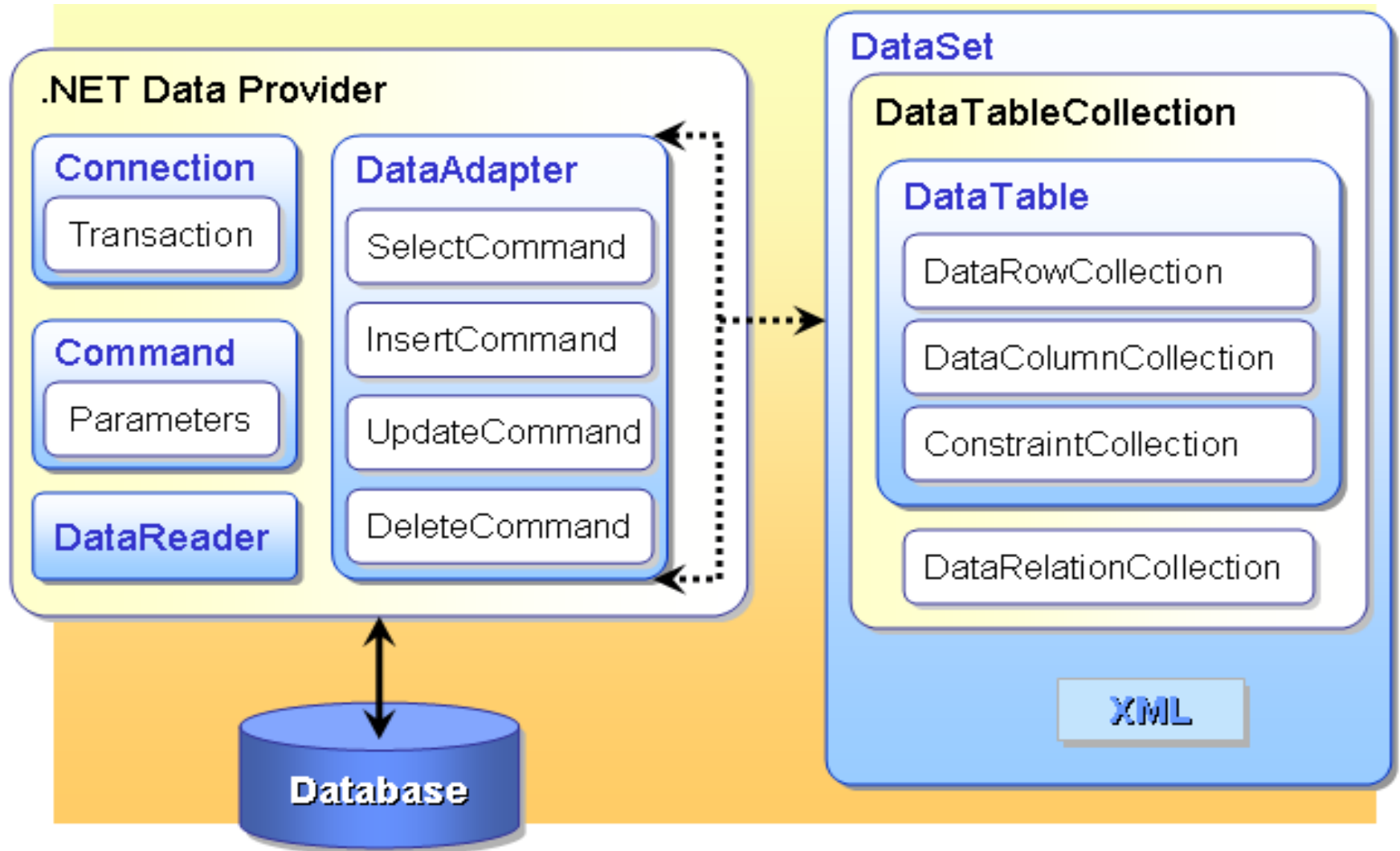
ADO.NET

- ▶ Kiến trúc của ADO.NET gồm 2 phần chính
- ▶ **Phần kết nối**: sử dụng khi kết nối CSDL và thao tác dữ liệu, phải thực hiện kết nối khi thao tác
 - **Connection**: quản lý việc đóng mở DB
 - ???Connection: **SqlConnection**, **OleDbConnection**
 - **Command**: lệnh truy vấn, tương tác dữ liệu khi đang lập kết nối
 - ???Command: **SqlCommand**, **OleDbCommand**
 - **DataReader**: đọc dữ liệu, chỉ xử lý 1 dòng dữ liệu tại một thời điểm
 - ???DataReader: **SqlDataReader**, **OleDbDataReader**
 - **DataAdapter**: cầu nối giữa DB và DataSet

ADO.NET

- ▶ **Phần ngắt kết nối:** là **DataSet**
 - DataSet không quan tâm đến DB thuộc kiểu gì, và lấy dữ liệu từ DataAdapter để xử lý
 - DataSet xem như một DB trong bộ nhớ: bảng, quan hệ...
 - DataSet có các thành phần con như
 - DataTable
 - DataRow
 - DataColumn
 - DataRelation
 - Các đối tượng nhóm: DataTableCollection, DataRowCollection, DataColumnCollection

Mô hình đối tượng ADO.NET



Các lớp thư viện ADO.NET

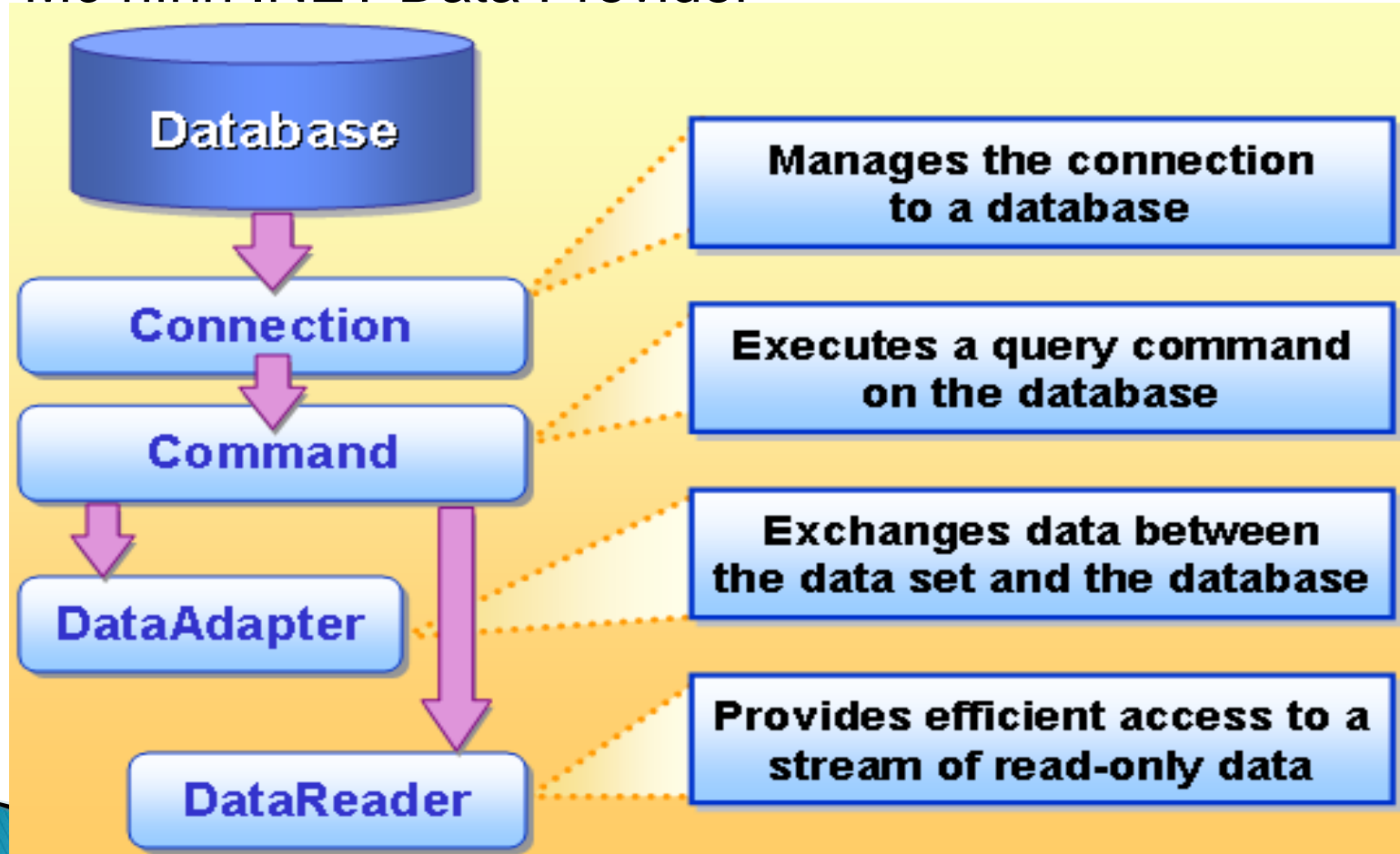
- ▶ System.Data.OleDb: Access, SQL Server, Oracle
- ▶ System.Data.SqlClient: SQL Server
- ▶ System.Data.OracleClient: Oracle
- ▶ Đặc điểm:
 - Cả ba thư viện trên về giao tiếp lập trình là giống nhau
 - Dùng thư viện SqlConnection truy xuất SQL Server nhanh hơn OleDb
 - Tương tự cho OracleClient

Nội Dung

- ADO.NET
- Quá trình phát triển
- Đặc điểm ADO.NET
- **.NET Data Provider**
- DataSet
- Data Source Configuration Wizard

NET Data Provider

► Mô hình .NET Data Provider

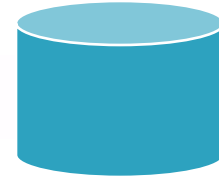


NET Data Provider - Connection



Application

Connection



Database

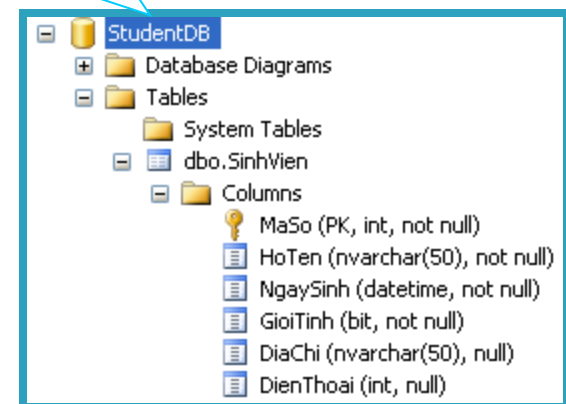
Thuộc tính & Phương thức

- **ConnectionString**: chuỗi kết nối DataSource
- **Open()**: thiết lập kết nối đến DS
- **Close()**: đóng kết nối với DS

NET Data Provider - Connection

- Mô tả CSDL sử dụng minh họa

Đặc điểm	Thông tin
DBMS	MS SQL Server Express 2005
DB 1	CSDL StudentDB
DB 2	CSDL Northwind



NET Data Provider - Connection

SQL Connection


```
SqlConnection cnn = new SqlConnection();  
cnn.ConnectionString = “server=.\\SQLEXPRESS;  
database=Northwind; Trusted_Connection=true”;
```

```
cnn.Open();
```

```
//xử lý trong quá trình kết nối
```

```
...
```

```
cnn.Close();
```



Thay bằng “localhost”
trong SQL Server 2005

NET Data Provider - Connection

Database	ODBC/OLE DB Connection (*)
MS Access	<p>Driver = {Microsoft Access Driver (*.mdb)}; DBQ = <đường dẫn file access> Provider=Microsoft.Jet.OLEDB.4.0; Data Source = <đường dẫn file access></p>
SQL Server	<p>Driver = {SQLServer}; Server = ServerName; Database= DatabaseName; Uid=Username; Pwd=Password;</p> <p>Provider= SQLOLEDB; Data Source=ServerName; Initial Catalog=DatabaseName; UserId=Username; Password=Password</p>

(*): Xem thêm chuỗi kết nối trong: <http://www.connectionstrings.com/?carrier=sqlserver2005>

NET Data Provider - Connection

OLEDB Connection

```
OleDbConnection cnn = new OleDbSqlConnection();  
cnn.ConnectionString = “Provider=Microsoft.Jet.OLEDB.4.0;  
                          Data Source =Sinhvien.mdb”;
```

```
cnn.Open();
```

```
//xử lý trong quá trình kết nối
```

```
...
```

```
cnn.Close();
```

NET Data Provider - Command



Thuộc tính & Phương thức

- **Connection**: kết nối để thực hiện lệnh
- **CommandText**: câu lệnh cần thực hiện
- **CommandType**: loại câu lệnh (Text, TableDirect, StoredProc)
- **ExecuteScalar()**: thực hiện câu lệnh và trả về giá trị đơn
- **ExecuteNonQuery()**: gọi các lệnh SQL, store, trả về số row bị tác động (Insert, Update, Delete...)
- **ExecuteReader()**: thực hiện lệnh và trả về **DataReader**

NET Data Provider - Command

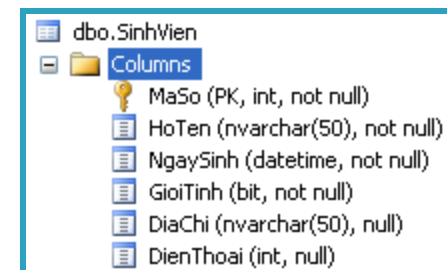
SQL Command

```
SqlConnection cnn = new SqlConnection();  
cnn.ConnectionString = "server=.\SQLEXPRESS;  
database=Northwind; Trusted_connection=true";  
  
SqlCommand cmd = new SqlCommand("",cnn);  
cmd.CommandText = "Select COUNT(*) From Customers";  
  
cnn.Open();  
  
int count = (int) cmd.ExecuteScalar();  
cnn.Close();
```

NET Data Provider - Command

SQL Command

```
SqlConnection conn = new SqlConnection();  
conn.ConnectionString = "Data source=.\SQLEXPRESS;" +  
    "Initial Catalog=StudentDB; Integrated security=yes";  
  
SqlCommand cmd = new SqlCommand();  
cmd.CommandText =  
    "Insert into Sinhvien values(007,'Le Nam','1/1/1980',1,null,null)";  
cmd.Connection = conn;  
  
conn.Open();  
  
cmd.ExecuteNonQuery();  
  
conn.Close();
```



NET Data Provider - Command

- ▶ Tham số hóa câu lệnh
 - Một câu lệnh được sử dụng nhiều lần

SQL Command - Param

...

```
cmd.CommandText = "Insert into Sinhvien
```

```
values(@MS,@HT,@NS,@GT,@DC,@DT)";
```

```
cmd.Parameters.Add("@MS", SqlDbType.Int);
```

```
cmd.Parameters.Add("@HT", SqlDbType.NVarChar);
```

```
cmd.Parameters.Add("@NS", SqlDbType.DateTime);
```

```
cmd.Parameters.Add("@GT", SqlDbType.Bit);
```

```
cmd.Parameters.Add("@DC", SqlDbType.NVarChar);
```

```
cmd.Parameters.Add("@DT", SqlDbType.Int);
```

Câu lệnh
được tham
số hóa

Khai báo
tham số

NET Data Provider - Command

SQL Command - Param

```
...  
cmd.Parameters["@MS"].Value = 999;  
cmd.Parameters["@HT"].Value = "Nguyen Ha Giang";  
cmd.Parameters["@NS"].Value = new  
DateTime(1978,12,4);  
cmd.Parameters["@GT"].Value = 1;  
cmd.Parameters["@DC"].Value = "Tan Binh";  
cmd.Parameters["@DT"].Value = 5120791;  
  
conn.Open();  
int count = (int) cmd.ExecuteNonQuery();  
  
conn.Close();
```

Truyền các
giá trị của
tham số
tương ứng

NET Data Provider - DataReader



Thuộc tính & Phương thức

- **HasRow**: cho biết câu truy vấn có trả về dữ liệu
- **Read()**: đọc một mẫu tin
- **[i]**: truy xuất đến cột i của mẫu tin được đọc
- **Close()**: đóng

Truy xuất tuần tự

Chỉ đọc dữ liệu

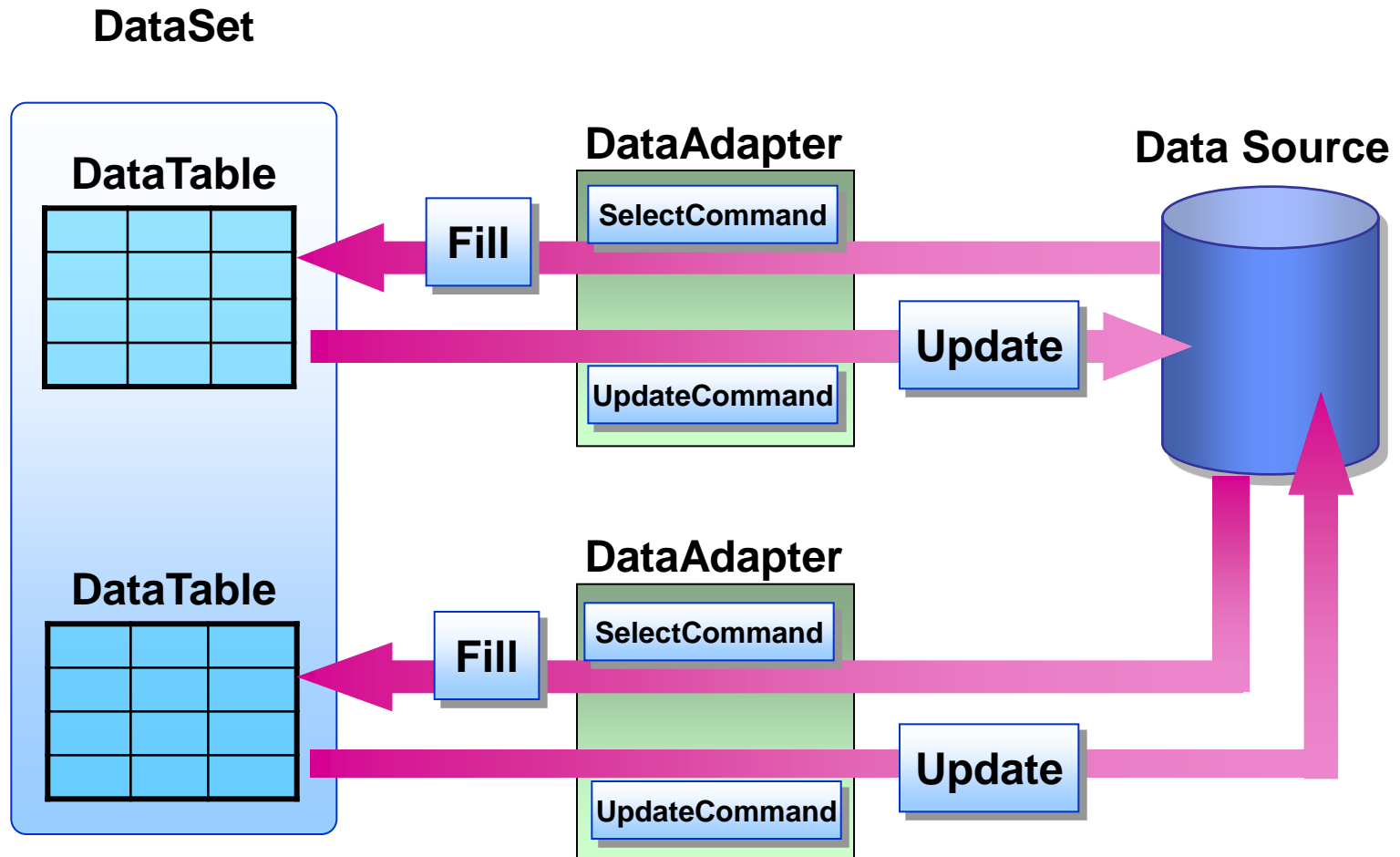
Cơ chế kết nối

NET Data Provider - DataReader

DataReader

```
...  
SqlCommand cmd = new SqlCommand("Select *  
From Sinhvien", conn);  
  
SqlDataReader reader;  
  
conn.Open();  
reader = cmd.ExecuteReader();  
while (reader.Read())  
    listBox1.Items.Add(reader["Hoten"]);  
reader.Close();  
conn.Close();
```

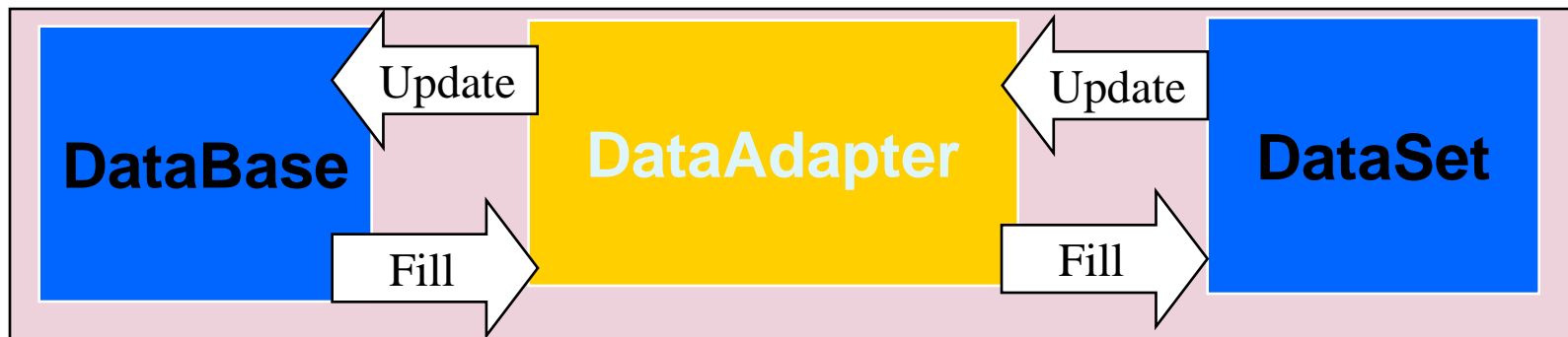

NET Data Provider - DataAdapter



NET Data Provider - DataAdapter

Thuộc tính & Phương thức

- **Fill**(DataSet): sử dụng SelectCommand lấy dữ liệu từ Data Source đổ vào Data Set
- **Update**(DataSet): InsertCommand, UpdateCommand, DeleteCommand cập nhật dữ liệu trong DataSet vào DataSource



NET Data Provider - DataAdapter

DataAdapter

```
string strConn="Server=.\SQLEXPRESS;  
Database=StudentDB; Trusted_connection=true";  
  
SqlDataAdapter adapter = new  
    SqlDataAdapter("Select * From Sinhvien",strConn);  
  
DataSet ds = new DataSet();  
  
adapter.Fill(ds);  
// thao tác trên dataset  
...  
adapter.Update(ds);
```

Nội Dung

- ADO.NET
- Quá trình phát triển
- Đặc điểm ADO.NET
- .NET Data Provider
- **DataSet**
- Data Source Configuration Wizard

DataSet

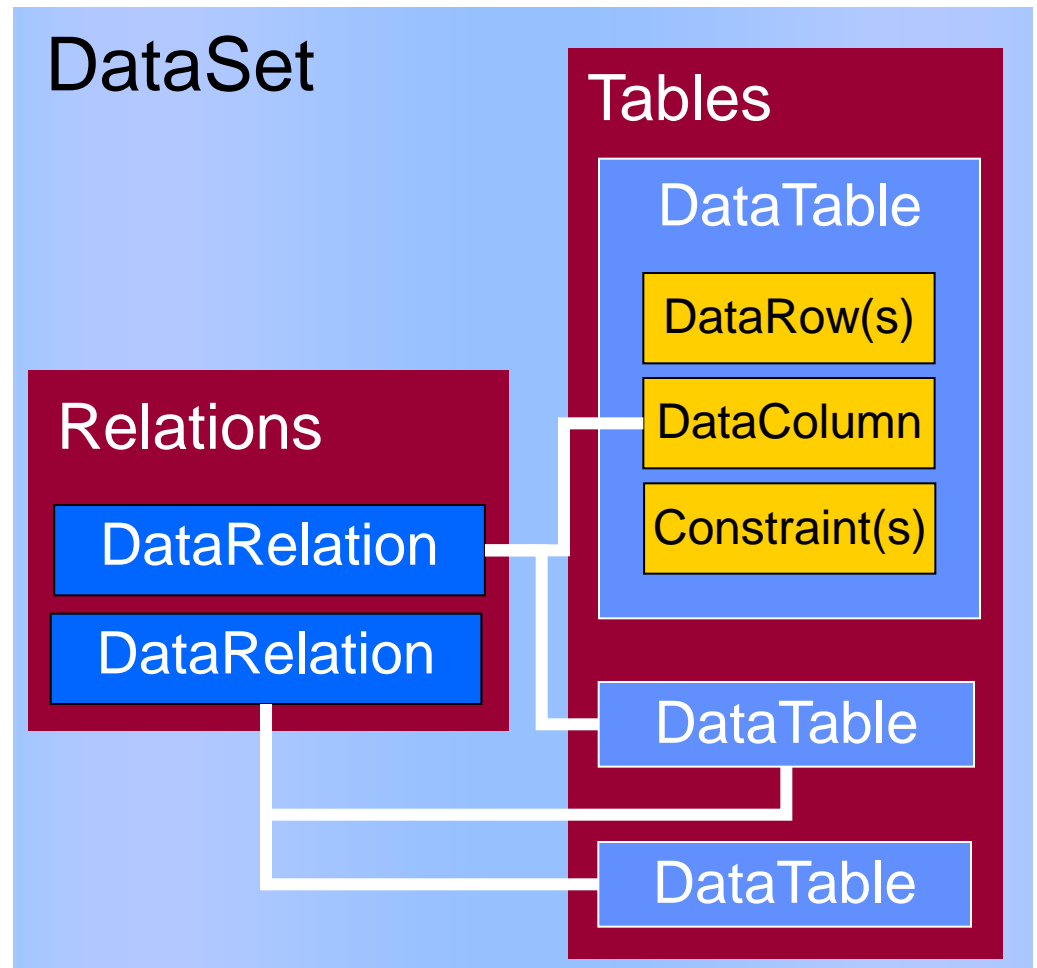
- ▶ **DataSet**: là phần cơ sở dữ liệu được lưu trữ trong bộ nhớ (in-memory database)
- ▶ Cơ chế không kết nối
- ▶ Nhờ đối tượng DataAdapter làm trung gian
- ▶ Hỗ trợ đầy đủ đặc tính XML
- ▶ Thao tác được với tất cả mô hình lưu trữ hiện tại: flat, relational, hierarchical

DataSet vs. DataReader

DataSet	DataReader
Read/write access to data	Read-only
Includes multiple tables from different databases	Based on one SQL statement from one database
Disconnected	Connected
Bind to multiple controls	Bind to one control only
Forward and backward scanning of data	Forward-only
Slower access	Faster access
Supported by Visual Studio .NET tools	Manually coded

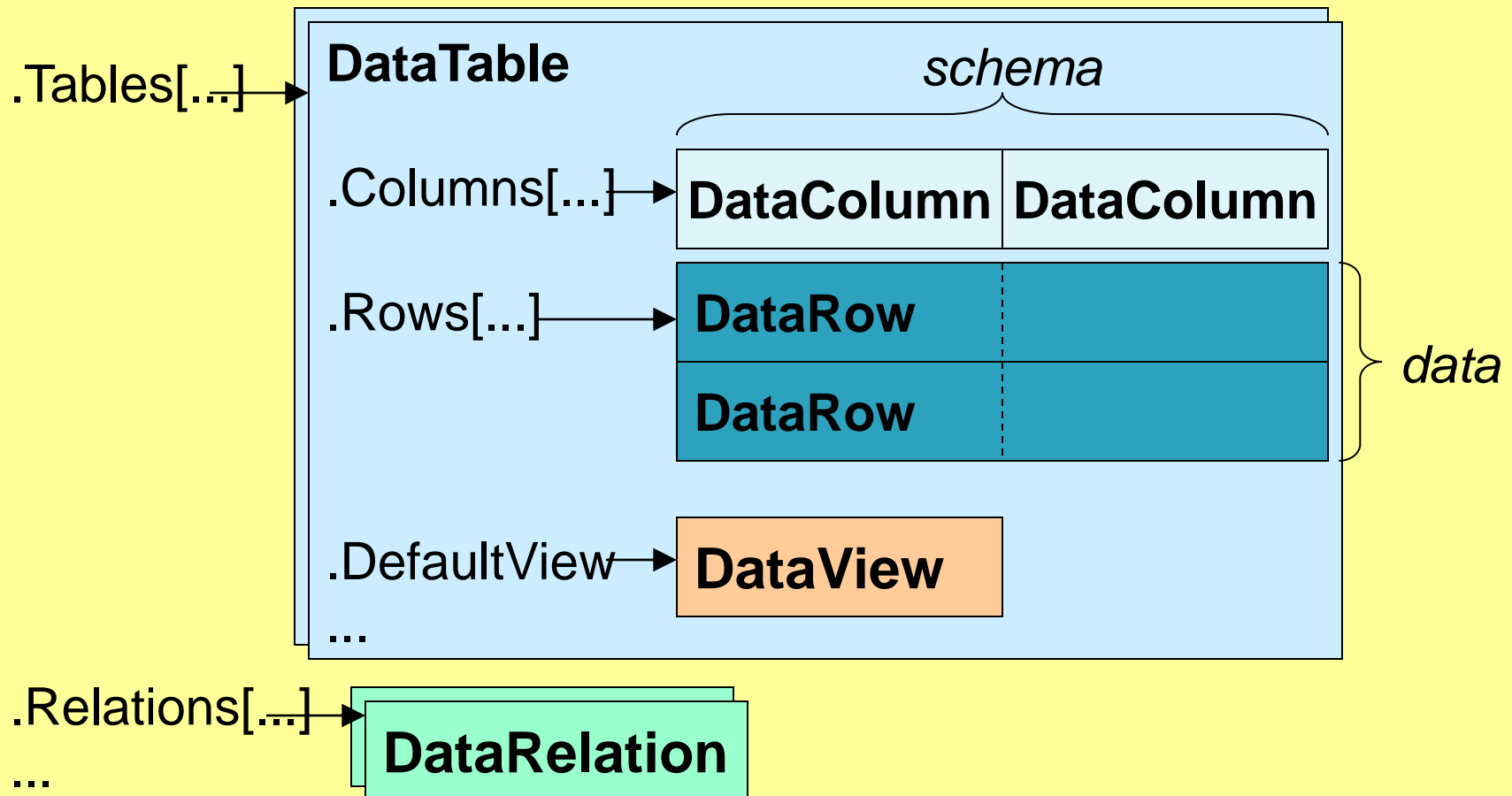
DataSet

- ▶ Các thành phần
 - Tables
 - Relations



DataSet

DataSet



DataSet - DataTable

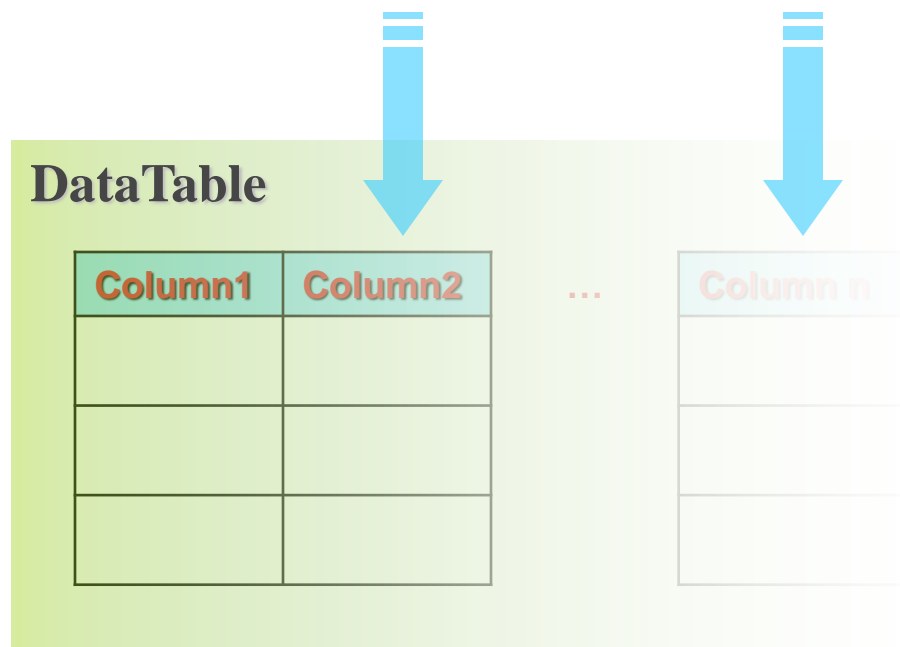
- ▶ DataTable: thể hiện một bảng trong CSDL

Thuộc tính & Phương thức

- **TableName:** tên của bảng dữ liệu
- **Columns:** danh sách các cột
- **Rows:** danh sách các mẫu tin
- **PrimaryKey:** danh sách các cột là khóa chính
- **NewRow():** tạo một mẫu tin mới

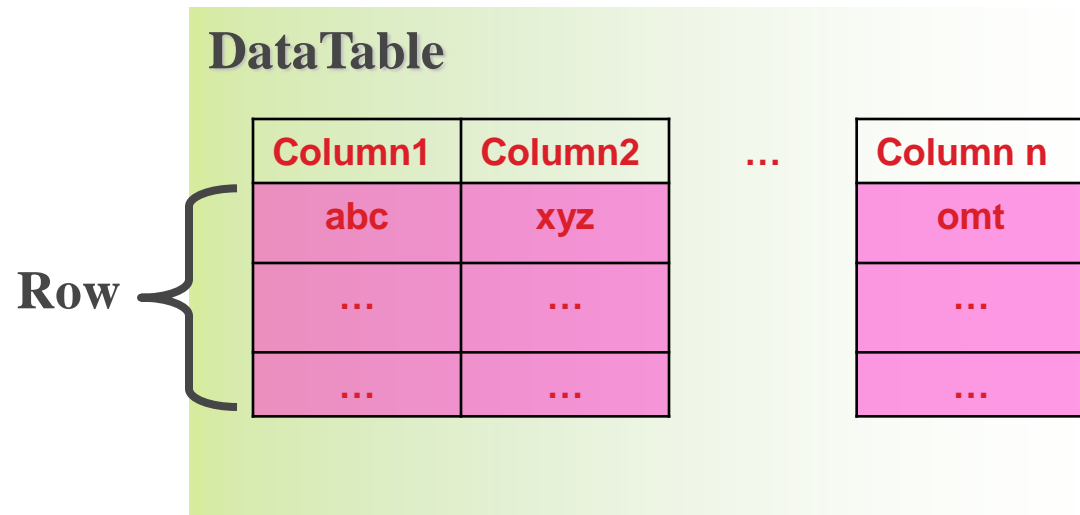
DataSet – DataColumn

- ▶ DataColumn: đại diện cho một cột trong bảng
 - ColumnName: tên cột
 - DataType: kiểu dữ liệu



DataSet - DataRow

- ▶ DataRow: đại diện cho mẫu tin trong bảng
 - RowState: trạng thái Added, Modified, Deleted,...
 - [i]: truy xuất đến cột i
 - Delete(): đánh dấu xóa mẫu tin



DataSet

DataSet – Update Row

```
string strConn = "data source=.\SQLEXPRESS; Initial  
Catalog=StudentDB; integrated security=yes;";  
string strCmd = "Select * From Sinhvien";  
SqlDataAdapter da = new SqlDataAdapter(strCmd, strConn);  
  
SqlCommandBuilder builder = new SqlCommandBuilder(da);  
DataSet ds = new DataSet();  
  
da.Fill(ds);  
  
foreach (DataRow dr in ds.Tables[0].Rows)  
    dr["Ngaysinh"] = DateTime.Now;  
  
da.Update(ds);
```

DataSet

DataSet – Delete Row

...

```
SqlDataAdapter da = new SqlDataAdapter(strCmd, strConn);  
SqlCommandBuilder builder = new SqlCommandBuilder(da);
```

```
DataSet ds = new DataSet();  
da.Fill(ds);
```

```
DataTable table = ds.Tables[0];
```

```
DataRow[] rows = table.Select("Ngaysinh<'1/1/1980'");
```

```
foreach (DataRow r in rows)  
    r.Delete();  
da.Update(ds);
```

Nội Dung

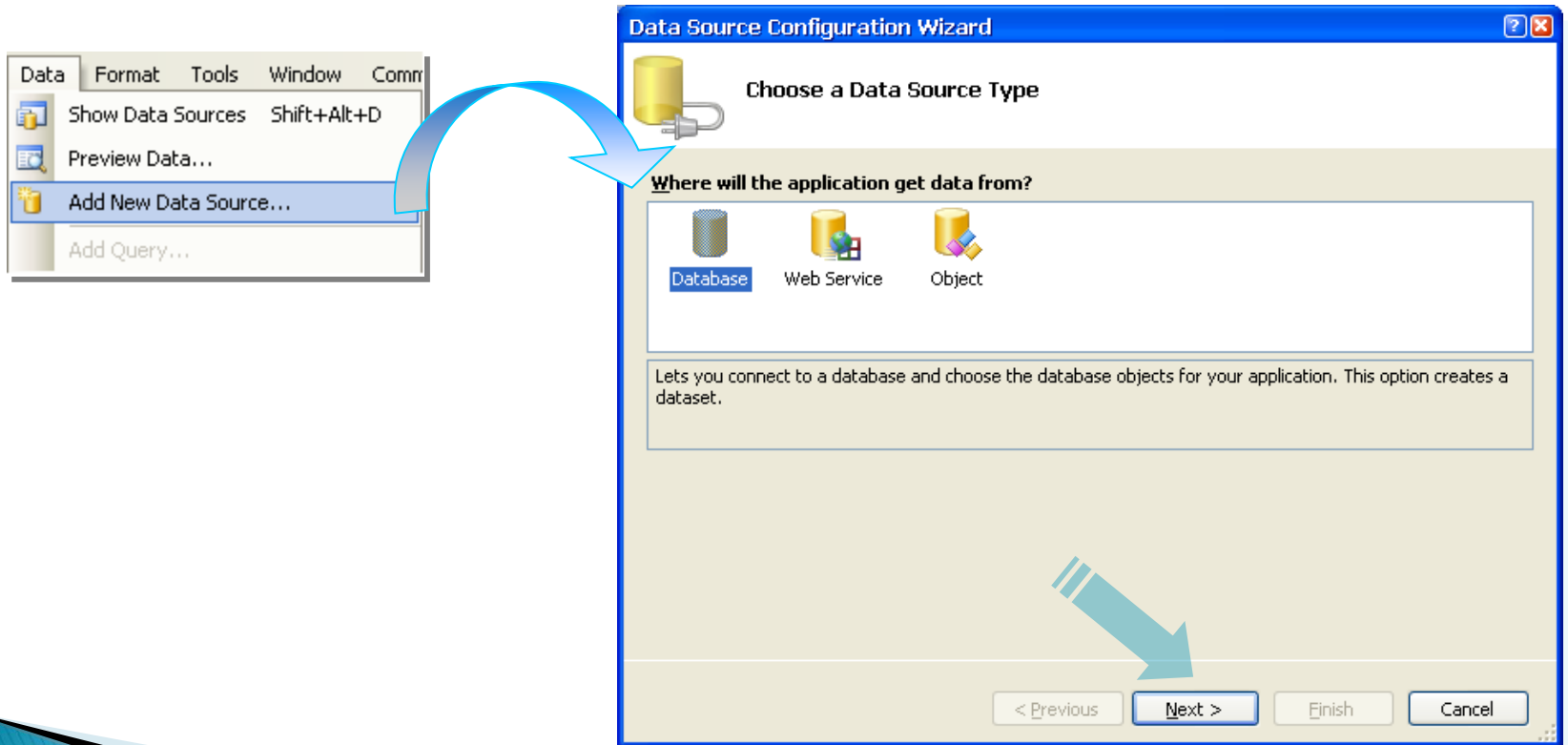
- ADO.NET
- Quá trình phát triển
- Đặc điểm ADO.NET
- .NET Data Provider
- DataSet
- **Data Source Configuration Wizard**

Data Source Configuration Wizard

- ▶ Trong VS.NET có chức năng Data Source Configuration Wizard
 - Nhanh chóng thiết lập Data Source cho project
 - Xây dựng form hiển thị và thao tác dữ liệu
 - Thông qua thao tác kéo thả từ Data Source

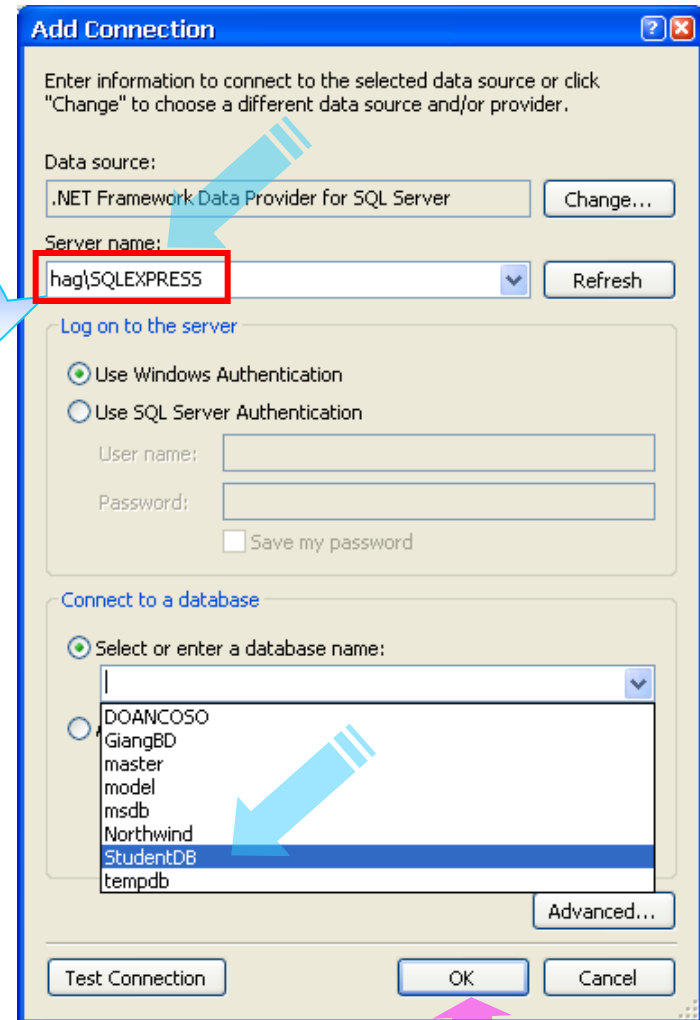
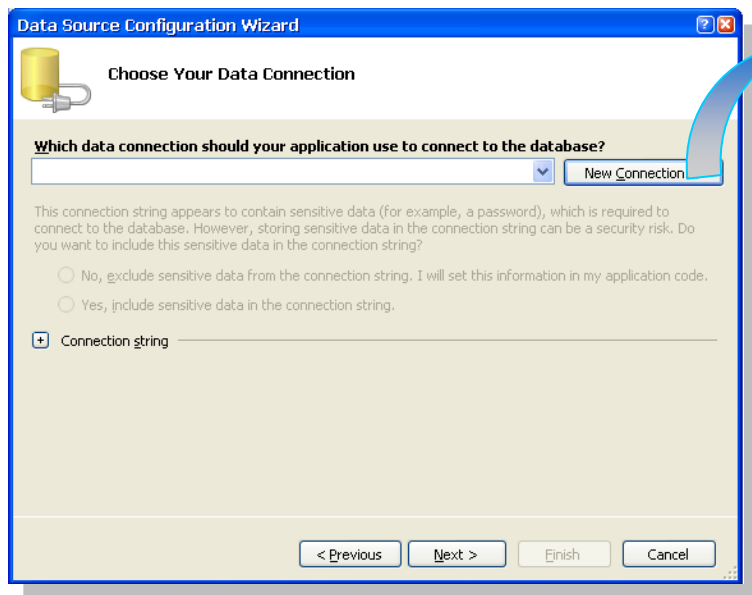
Data Source Configuration Wizard

- ▶ Tạo một ứng dụng Windows Application
- ▶ Trong menu Data | Add New Data Source...



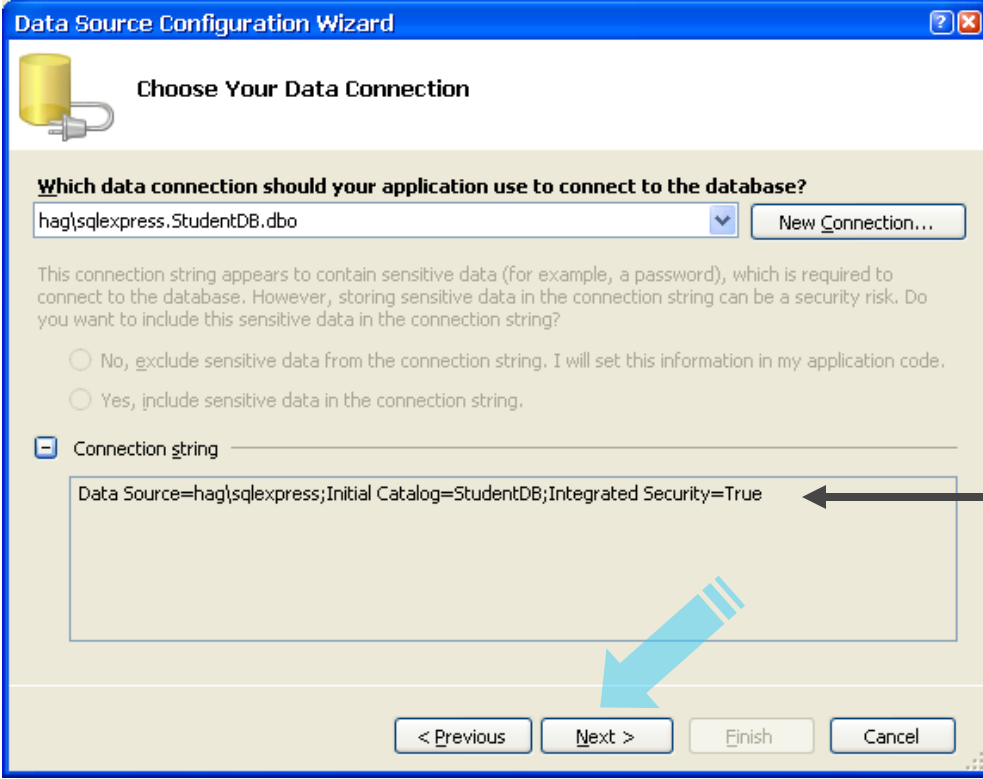
Data Source Configuration Wizard

► Tạo New Connection



Data Source Configuration Wizard

► Hoàn tất khai báo Data Source



Data Source Configuration Wizard

Choose Your Data Connection

Which data connection should your application use to connect to the database?

hag\sqlexpress.StudentDB.dbo New Connection...

This connection string appears to contain sensitive data (for example, a password), which is required to connect to the database. However, storing sensitive data in the connection string can be a security risk. Do you want to include this sensitive data in the connection string?

☐ No, exclude sensitive data from the connection string. I will set this information in my application code.

☒ Yes, include sensitive data in the connection string.

☒ Connection string

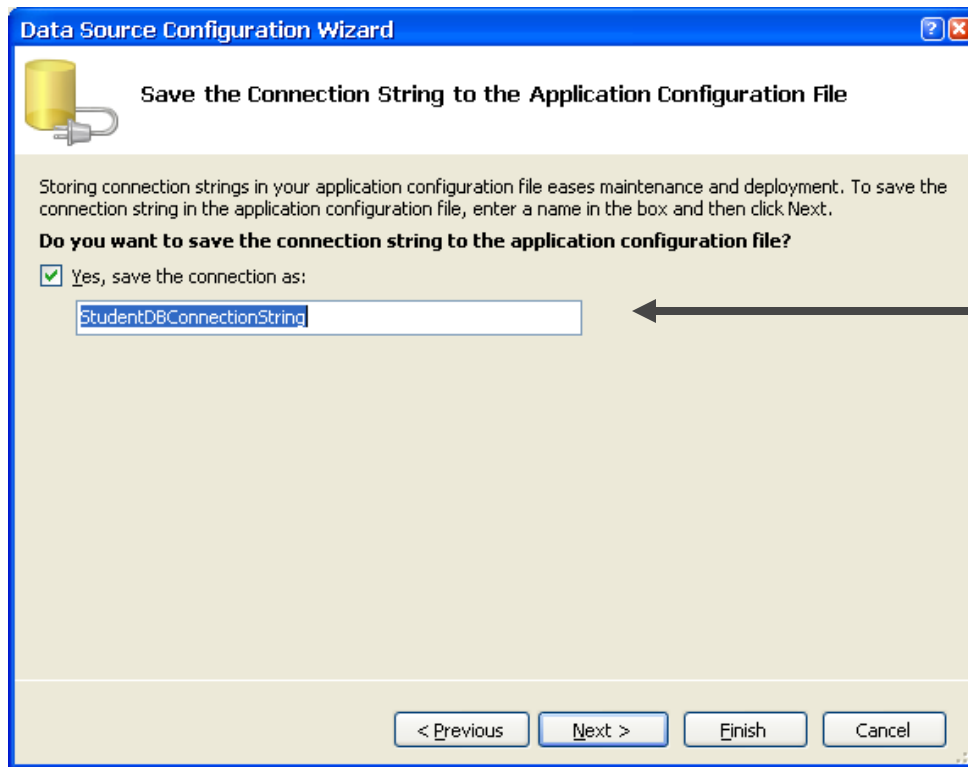
Data Source=hag\sqlexpress;Initial Catalog=StudentDB;Integrated Security=True

< Previous Next > Finish Cancel

Chuỗi kết nối

Data Source Configuration Wizard

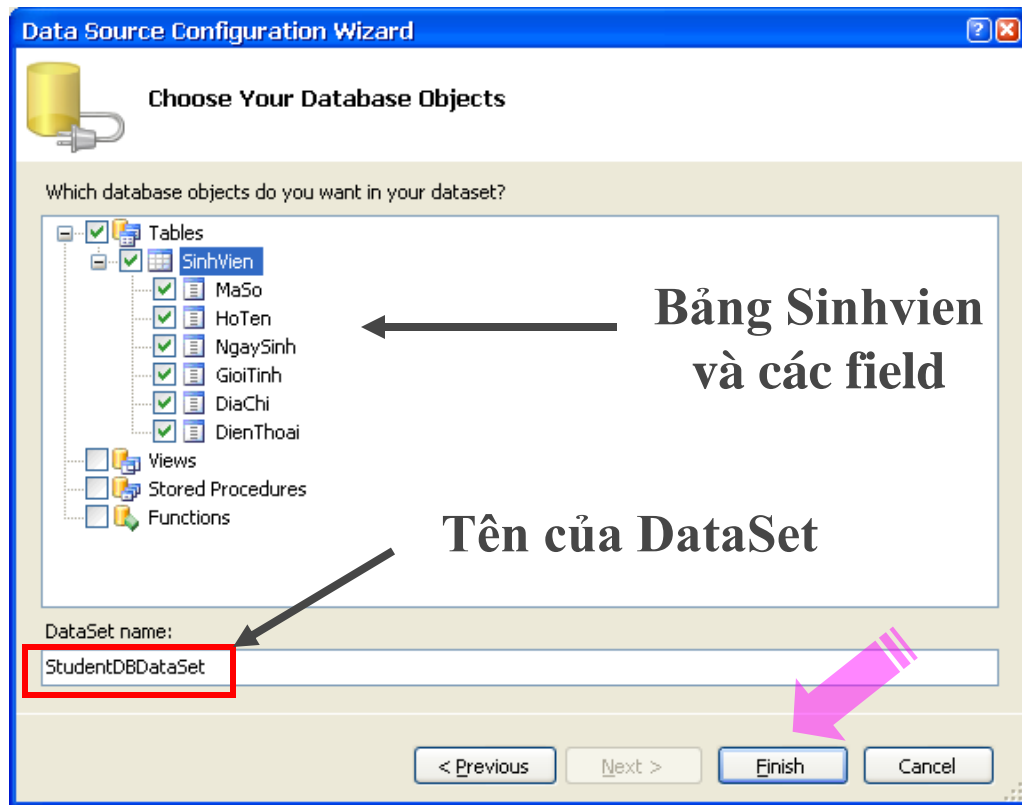
- ▶ Lưu chuỗi kết nối trong file cấu hình



Tên của chuỗi kết nối

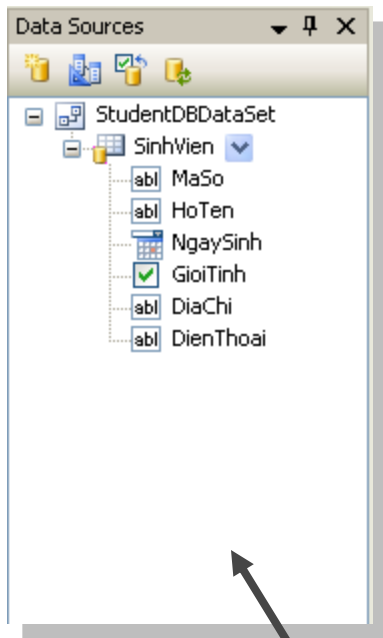
Data Source Configuration Wizard

- ▶ Chọn bảng dữ liệu

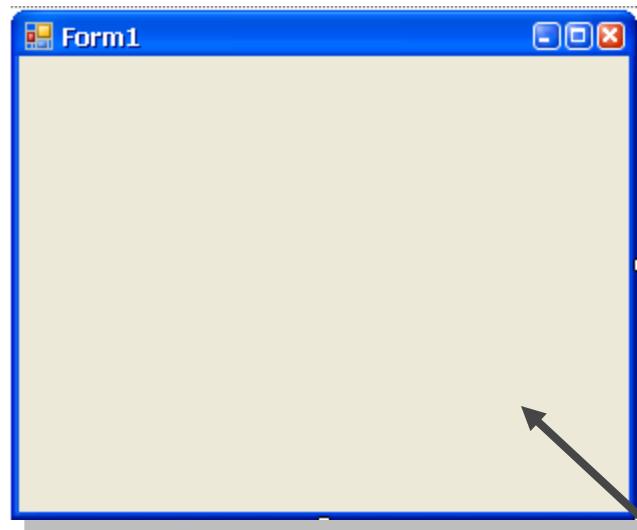


Data Source Configuration Wizard

- ▶ Wizard sẽ tạo ứng dụng với Data Source



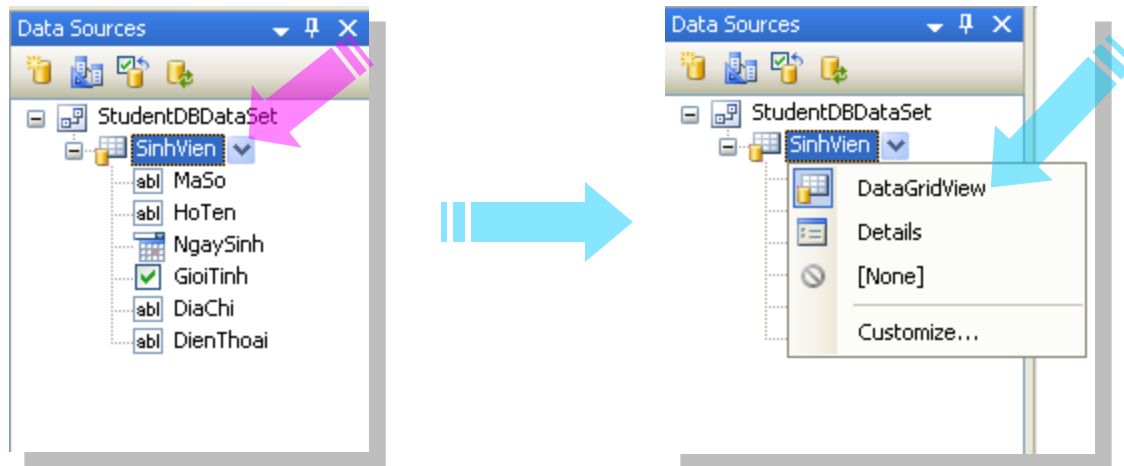
Data Sources Windows



Form in Design View

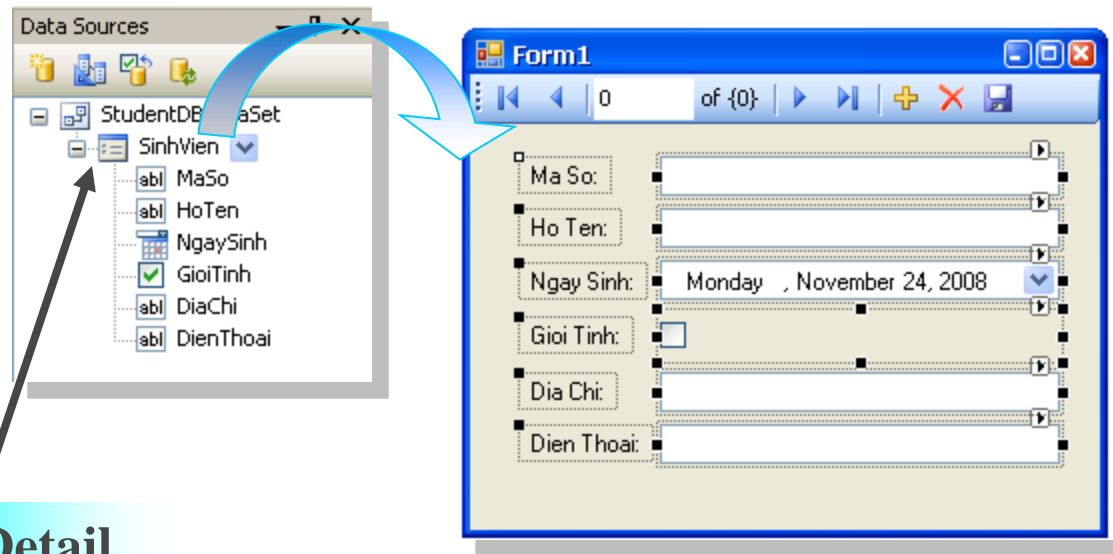
Data Source Configuration Wizard

- ▶ Kéo thả binding control vào Form
 - Trong cửa sổ Data Source
 - Chọn bảng cần sử dụng
 - Thiết lập view là DataGridView hay Details
 - Kích chọn vào dấu mũi tên xuống sau tên bảng



Data Source Configuration Wizard

- ▶ Kéo Table thả vào Form
 - Tự động tạo các binding control cho table



Detail



Data Source Configuration Wizard

- ▶ Bổ sung DataGridView cho Form
 - Thay đổi Table sang DataGrid
 - Kéo Table trong Data Source thả vào Form

DataGridView

	MaSo	HoTen	NgaySinh	GioiTinh	DiaChi	DienThoai
*				<input type="checkbox"/>		

studentDBDataSet sinhVienTableAdapter
sinhVienBindingSource sinhVienBindingNavigator

Data Source Configuration Wizard

► Kết quả ứng dụng

Form1

1 of 4

Ma So: Gioid Tinh: ☒

Ho Ten: Dia Chi:

Ngay Sinh: Dien Thoai:

	MaSo	HoTen	NgaySinh	GioidTinh	DiaChi	DienThoai
►	1	Nguyen Ha Giang	12/4/1978	<input checked="" type="checkbox"/>	Hoang Hoa Tham	5120391
	2	Nguyen Ha Nam	1/1/2010	<input checked="" type="checkbox"/>	Le Van Sy	
	3	Nguyen Ha MyTien	2/10/1999	<input type="checkbox"/>	Hoa Hao	9318146
	4	Nguyen Ha Thanh Tung	8/26/2000	<input checked="" type="checkbox"/>	Le Van Sy	
*				<input type="checkbox"/>		

Chưa viết code!