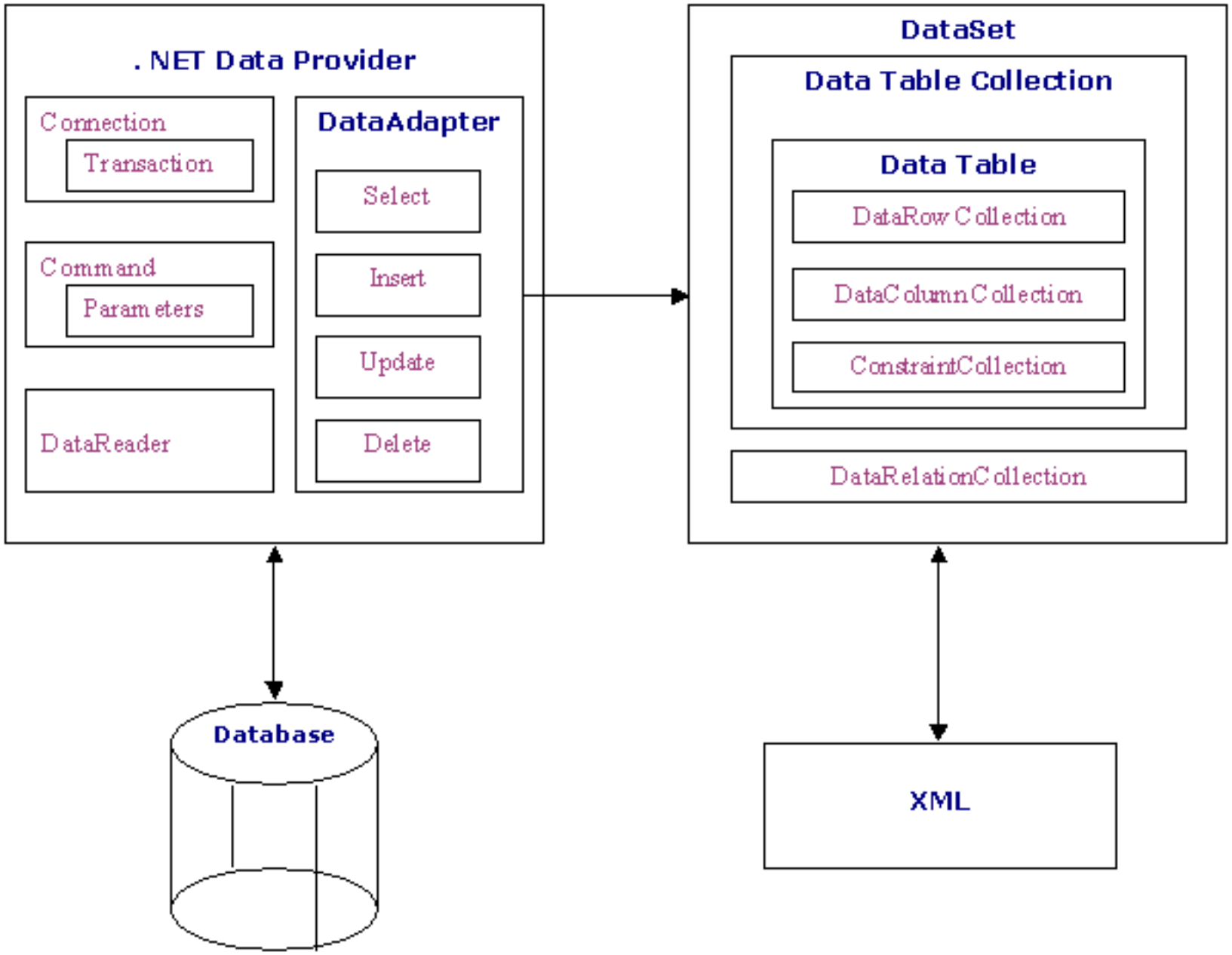
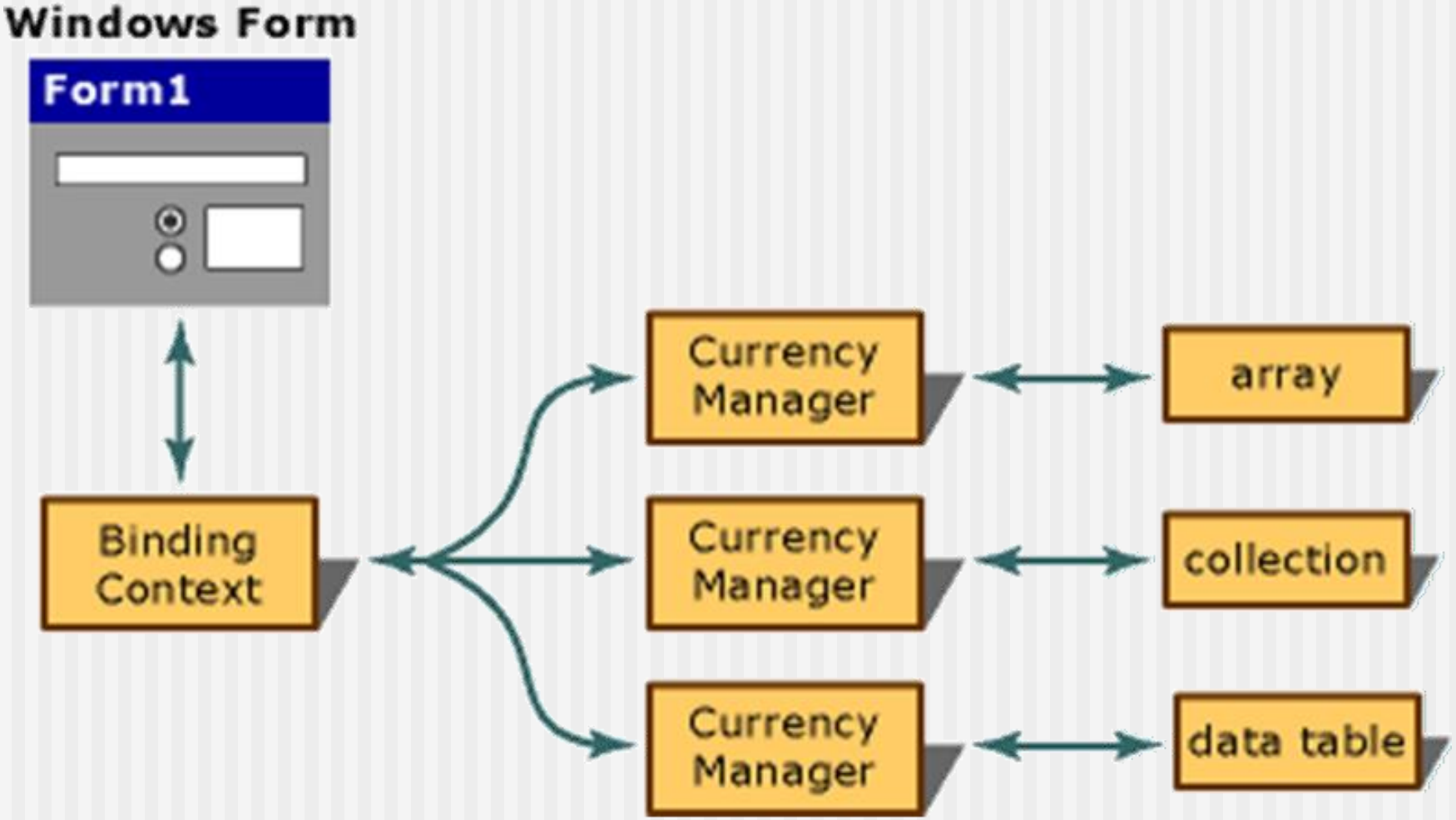
Seminar 2

ADO.NET



ADO .NET Data Architecture

Data Binding in Windows Forms



Data providers in .NET

- To act as a data source, a list must implement the **IList** interface;
- ADO.NET provides data structures suitable for binding to:
  - DataColumn
  - DataTable – composed by Columns, Rows, Constraints
  - DataView – customized view of a single data table
  - DataSet – composed by Tables, Relationships, Constraints
  - DataViewManager – customized view of a DataSet

Data consumers in .NET

- Currency Manager
  - keeps data-bound controls synchronized with each other
  - For each data source associated with a Windows Form, there is one **CurrencyManager** object.
  - Currency = the currentness of position within a data structure
  - Position - determine the currency of all controls bound to the same **CurrencyManager**
- BindingContext
  - Manages the collection of ***CurrencyManager*** objects for any container control.

Populating Datasets with Data

- A dataset contains no actual data by default

- Data tables filled by executing ***TableAdapter*** queries, or executing data adapter commands (***SqlDataAdapter***)

```
aTableAdapter.Fill(aDataSet.TableName);
```

- Saving data

```
aTableAdapter.Update(aDataSet.TableName)
```

Update method examines the value of the ***RowState*** property to determine which records need to be saved and what specific database command (*InsertCommand*, *UpdateCommand*, and *DeleteCommand*) should be invoked.

Navigating Records

- each table exposes a collection of rows.

- like any collection, rows are accessed by means of the collection's index or using collection-specific statements in host programming language.
- typed dataset

```
TextBox1.Text = aDataSet.TableName[3].aField
```

- untyped dataset

```
string IDValue = (string)
dataset1.Tables[“aTable"].Rows[0][“aField"];
```

Related Tables and *DataRelation* Objects

- The information in the tables of a ***DataSet*** might be related
- Create ***DataRelation*** objects that describe the relations between the tables in the dataset
- You can use a ***DataRelation*** object to locate related records by calling the ***GetChildRows*** method of a ***DataRow*** in the parent table; this method returns an array of related child records.
- Or you can call the ***GetParentRow*** method of a ***DataRow*** in the child table; this method returns a single ***DataRow*** from the parent table.

Return child records of a parent record

```
string custID = "ALFKI";
NorthwindDataSet.OrdersRow[] orders;
orders = ( NorthwindDataSet.OrdersRow[] )
     northwindDataSet.Customers.
     FindByCustomerID(custID).GetChildRows
          ("FK_Orders_Customers");
MessageBox.Show(orders.Length.ToString());
```

Return parent records of a child record

```
int orderID = 10707;
NorthwindDataSet.CustomersRow customer;
customer = (NorthwindDataSet.CustomersRow)
      northwindDataSet.Orders.
      FindByOrderID(orderID).
      GetParentRow("FK_Orders_Customers");
MessageBox.Show(customer.CompanyName);
```

Create new Application in Visual Studio

Create new ***Windows Form*** project

- From the ***File menu,*** create a new project.
- Select ***Windows Forms Application*** and give it a name.
- Click ***OK***. The project is created and added to ***Solution Explorer***.

Create ***Data Source***

- Start ***Data Source Configuration Wizard*** (*Data Source* window)
- Choose Your Data Connection
- Select needed tables

Data-bound controls

- **DataSet** - the typed dataset that contains tables.
- **Binding Source** - binds the controls on the form to the data table in *DataSet*.
- **Binding Navigator** - is used for traversing the records in the table.
- **Table Adapter** - communicates between the database and *DataSet*.
- **TableAdapterManager** - is used to control the order of Inserts, Updates and Deletes for all TableAdapter components.

Constraints

- Two types of constraints:
  - A unique constraint: checks that the new values in a column are unique in the table.
  - A foreign-key constraint: defines rules for how related child records should be updated when a record in a master table is updated or deleted. (by creating a DataRelation object in a dataset a foreign-key constraint is automatically added)
- Constraints are implemented as objects of type UniqueConstraint or ForeignKeyConstraint. They are then added to the Constraints collection of a DataTable.
- The dataset itself supports a Boolean EnforceConstraints property that specifies whether constraints will be enforced or not (by default it is true).

Entity Framework

- set of technologies in ADO.NET
- support development of data-oriented applications
- enables developers to work with data in the form of domain-specific objects and properties, without having to concern with the underlying database tables and columns where this data is stored.
- gives life to models by enabling developers to query entities and relationships in the domain model while relying on the Entity Framework to translate those operations to data source–specific commands.

Entity Framework