

Учебен план

1. Въведение в платформата .NET
2. Език за програмиране C#
3. Изграждане на уеб приложения с технологията WebForms
4. Използване на бази от данни
5. Разработване на уеб услуги
6. Изграждане на уеб приложения с технологията MVC/Razor

.NET технологии

Тема 4. Използване на бази от данни

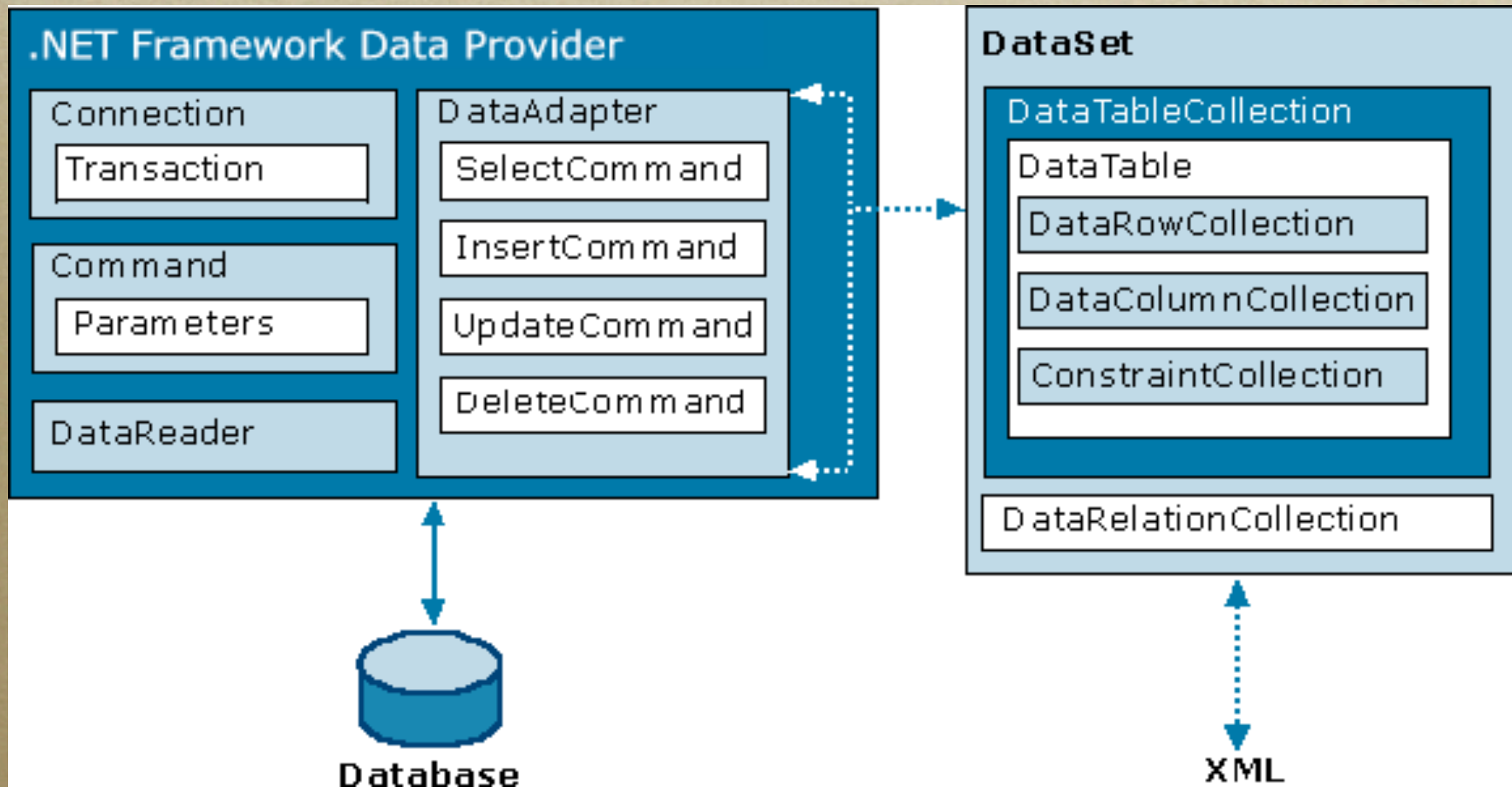
Основни постановки

- ◆ наречен ADO.NET
- ◆ namespaces System.Data.*
- ◆ данните могат да се използват/
манипулират ръчно (директно
писане на код) или да се свързват
с контроли във формите

Класове / Компоненти

- ◆ Data Provider – осигурява достъп до определен тип данни
- ◆ Connection – връзка с източник на данни
- ◆ Data Adapter – представя част от данните на източника (напр. таблица или резултат от SELECT)
- ◆ Data Set – набор от данни в паметта, получен от Data Adapter или XML файл

Класове/Компоненти



Основни видове data provider(s)

- ◆ SQL Server
- ◆ Oracle (развитието се спира)
- ◆ ODBC
- ◆ OLE DB (има вероятност също да бъде премахнат)

SQL Server

- ◆ оптимизиран достъп до SQL Server, да се избира именно този вариант за тази СУБД
- ◆ поддържа се версия 7.0 и по-нови
- ◆ namespace System.Data.SqlClient

Oracle

- ◆ оптимизиран достъп до Oracle, да се избира именно този вариант за тази СУБД
- ◆ поддържа се версия 8.1.7 и по-нови
- ◆ namespace
System.Data.OracleClient
- ◆ да се търси нова поддръжка от Oracle или трети страни

ODBC

- ◆ достъп до произволна база от данни, за която има ODBC драйвер
- ◆ достъпът не е толкова добре оптимизиран; да не се използва за SQL Server или Oracle
- ◆ namespace System.Data.Odbc

OLE DB

- ◆ достъп до произволна база от данни, за която има OLE DB Provider
- ◆ достъпът не е добре оптимизиран, стара технология, за съвместимост
- ◆ за SQL Server версия 6.5 или по-стари, възможност за достъп и до Access
- ◆ namespace System.Data.OleDb

"Ръчен" програмен достъп

- ◆ примери за SQL Server, namespace `System.Data.SqlClient`
- ◆ клас `SqlConnection` – създава връзка със сървъра
- ◆ клас `SqlCommand` – формира SQL заявка
- ◆ клас `SqlDataReader` – получава резултат от `SELECT` заявка

Клас SqlConnection

- ◆ конструктор SqlConnection(string ConnectionString)
- ◆ ConnectionString е и свойство, което може да се укаже/промени допълнително

ConnectionString – някои опции

- ◆ `Server="server";` (свойство `DataSource`)
- ◆ `Database="database";` (свойство `Database`)
- ◆ `Integrated Security="true";/"false";`
– Windows имена/пароли или се дават директно в стринга
- ◆ `User Id = "user";`
- ◆ `Password = "password";`

SqlConnection – някои методи

- ◆ Open – за отваряне, трябва да е зададен `ConnectionString`
- ◆ Close – затваряне
- ◆ `BeginTransaction` – започва/създава транзакция

Отваряне на SqlConnection

```
SqlConnection mySqlCon = new  
    SqlConnection("server=localhost; database=stud;  
    integrated security=true");  
  
mySqlCon.Open();  
  
...  
  
mySqlCon.Close();
```


Транзакции

- ◆ операциите, извършени по време на транзакцията или се прилагат или се отменят като цяло
- ◆ клас `SqlTransaction`, връща се от `SqlConnection.BeginTransaction`
- ◆ `Commit` – прилага операциите (транзакцията)
- ◆ `Rollback` – отменя операциите (транзакциите)

Транзакция – пример

```
SqlConnection mySqlConnection = new  
    SqlConnection("server=localhost; database=stud;  
    integrated security=true");
```

```
mySqlConnection.Open();
```

```
SqlTransaction myTrans;  
myTrans = mySqlConnection.BeginTransaction();
```

```
...
```

```
myTrans.Commit(); или myTrans.Rollback();
```

```
mySqlConnection.Close();
```


Клас SqlDataAdapter

- ◆ свързващото звено между SqlConnection и DataSet (който може да се използва ръчно или да се избира за източник на данни за контроли)
- ◆ може да се извършват двупосочни операции – получаване на данни от него и изпращане на данни към него

SqlDataAdapter – някои свойства/методи

- ◆ конструктор `SqlDataAdapter(string SelectCommand, SqlConnection connection)`
- ◆ `Fill(DataSet ds)`
- ◆ `Update(DataSet ds)`

Клас DataSet

- ◆ представяне в паметта на данни от база от данни или файл
- ◆ може да се използва програмно или като източник на данни за контроли по формите
- ◆ има колекция Tables от тип DataTable

Клас DataTable

- ◆ представя таблица от база от данни или файл
- ◆ има колекция Columns
- ◆ има колекция Rows от тип DataRow
- ◆ метод NewRow