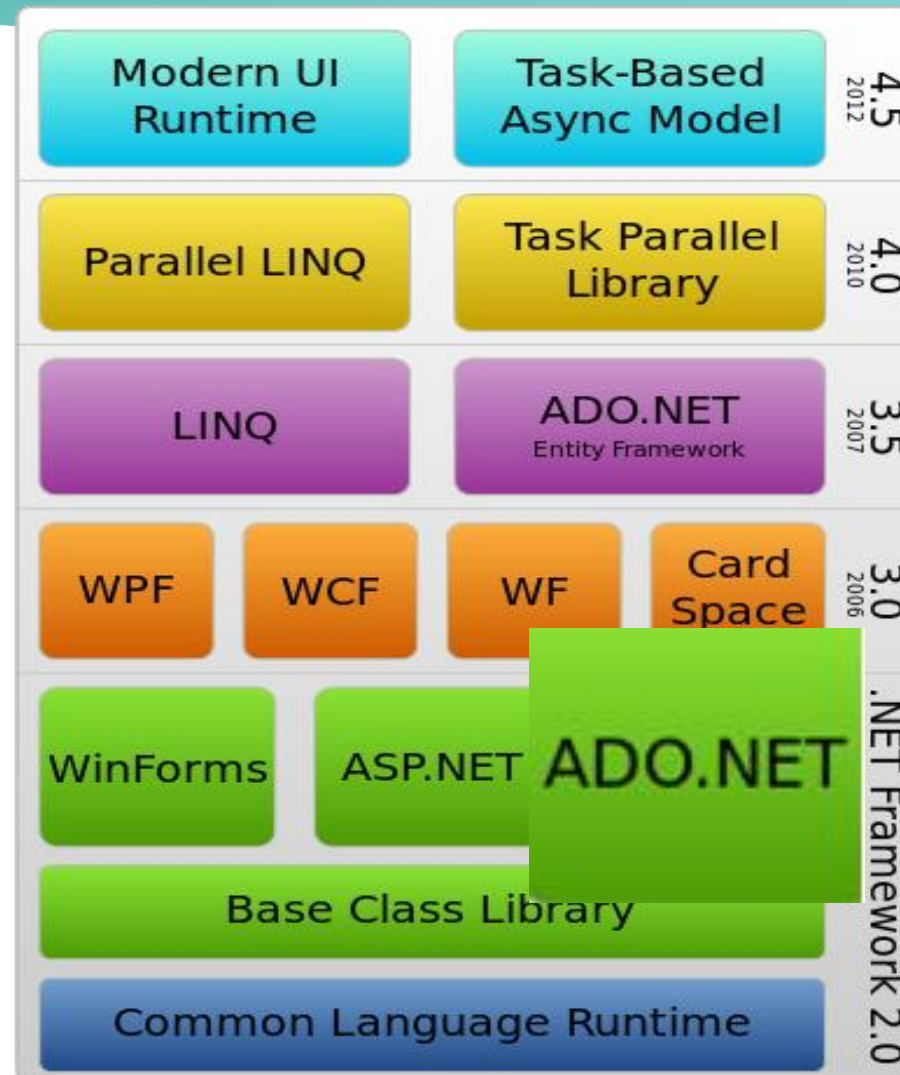


# Primena OOP u izradi složenih aplikacija (programski jezik C#)

Uvod u ADO.NET  
(prvi deo)

- ❑ Deo .NET razvojnog okruženja
- ❑ Tehnologija pristupa podacima
  - Učitavanje podataka u .NET objekte kako bi im se na taj način omogućio pristup iz programskog koda
  - Obično izvor podataka predstavljaju baze podataka, ali može da bude i tekstualni fajl, Excel tabela i sl.
- ❑ Pogodnosti
  - Aplikacije mogu da pristupaju različitim izvorima podataka koristeći isti način pristupa
    - Koriste samo različit skup klasa za različite izvore podataka
  - Postoje dva modela pristupa podacima:
    - Otvorena veza ka bazi (konektovani režim)
    - Zatvorena veza ka bazi (diskonektovan režim)

# .NET Framework stek – ADO.NET



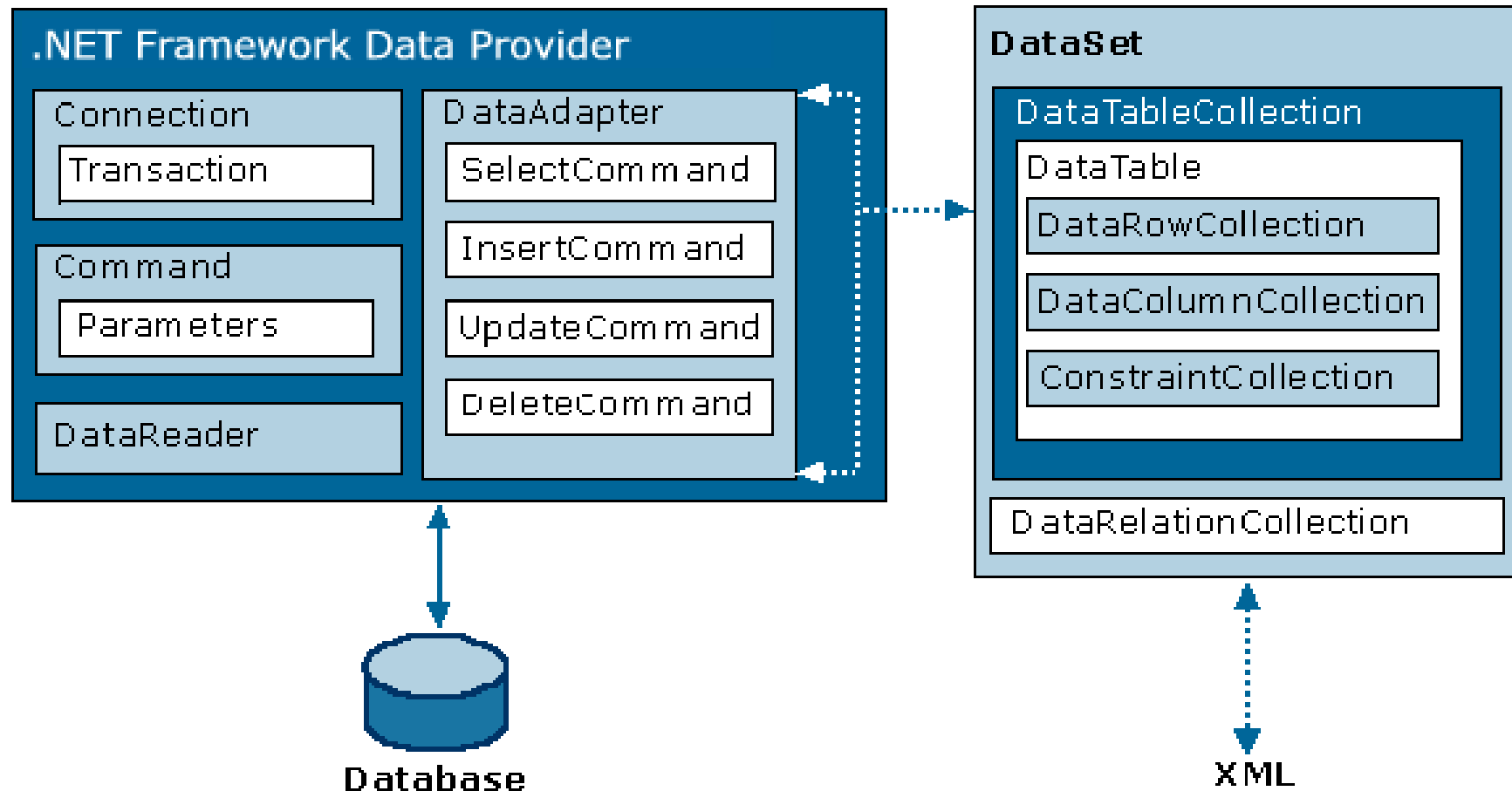
The .NET Framework Stack

# Pregled arhitekture

## □ Dve osnovne komponente

- Skup podataka (eng. *DataSet*)
  - Nije povezan sa izvorom podataka i ne zahteva poznavanje porekla podataka
- Dobavljač podataka (eng. *.NET Framework Data Provider*)
  - Omogućuje povezivanje sa izvorom podataka i izvršenje naredbi nad tim podacima

# Pregled arhitekture



# .NET Framework Data Provider

- ❑ Omogućuje povezivanje sa izvorom podataka i izvršenje naredbi nad tim podacima
- ❑ Sastoji se od četiri glavne komponente
  - *Connection*
  - *Command*
  - *DataReader*
  - *DataAdapter*

# Connection

- ❑ Koristi se za povezivanje sa izvorom podataka
- ❑ U zavisnosti od vrste izvora podataka, *Connection* objekat može da bude *SqlConnection*, *OleDbConnection*, *OdbcConnection* ili *OracleConnection*
  - Koristi se MS SQL Server SUBP ⇔ koriste se *Sql<Ime\_komponente>* klase
  - Klase se nalaze u *System.Data.SqlClient* imenskom prostoru
- ❑ Atribut *ConnectionString*
  - *DataSource* – Lokacija na kojoj se nalazi baza podataka
  - *Integrated Security*
    - Vrednost *true* ili *SSPI* – koristi se trenutni Windows nalog za autentikaciju
    - Vrednost *false* – potrebni su *username* i *password* u *ConnectionString*-u
  - *Initial Catalog* – Ime baze podataka

# Connection – Primer

```
using (SqlConnection konekcija = new SqlConnection())  
{  
    konekcija.ConnectionString = "Data Source = localhost;  
Integrated Security = SSPI; Initial Catalog=CMT";  
    konekcija.Open();  
}
```



# Command

- ❑ Posle uspostavljanja konekcije, potrebno je bazi podataka reći koju je akciju (naredbu) potrebno izvršiti nad njom
- ❑ Pomoću objekta klase *Command* zadaju se SQL naredbe bazi podataka
- ❑ Neke od metoda *Command* objekta:
  - *ExecuteNonQuery* – Izvršava naredbu i vraća broj redova zahvaćenih naredbom
  - *ExecuteScalar* – Izvršava naredbu i kao rezultat vraća podatak u prvoj koloni prvog reda skupa podataka

# Command - Primer

```
string naredba = "INSERT INTO skola VALUES (4, 'Gimnazija Jovan  
Jovanović Zmaj', 'Zlatne Grede 4')";
```

```
SqlCommand naredba= new SqlCommand(naredba, konekcija);
```

```
naredba.ExecuteNonQuery();
```

# Hvala na pažnji!



CENTAR ZA  
mlade talente