

# Windows Forms

Proseminar *Objektorientiertes Programmieren mit .NET und C#*

Victor Duca

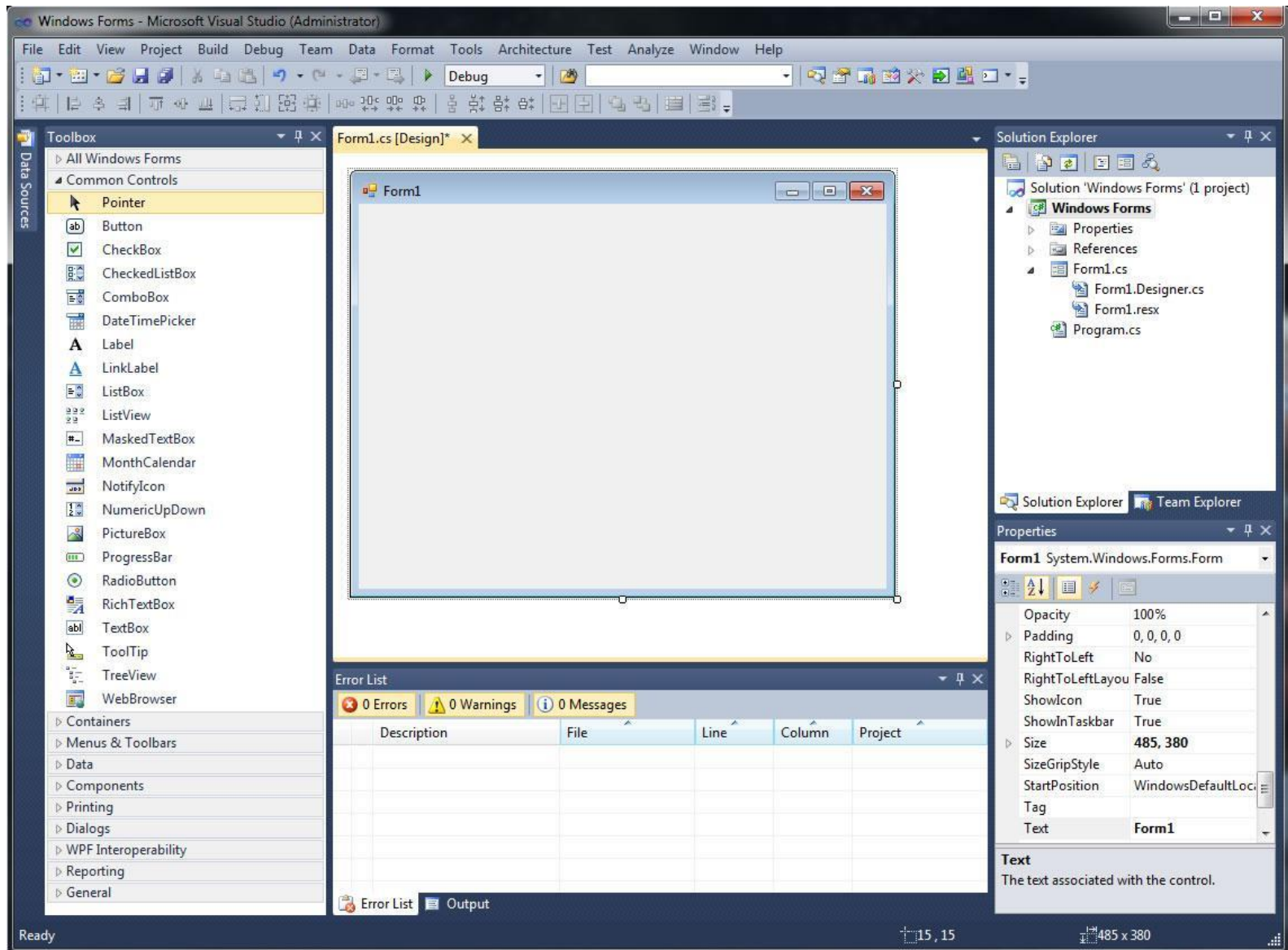
Institut für Informatik  
Software & Systems Engineering

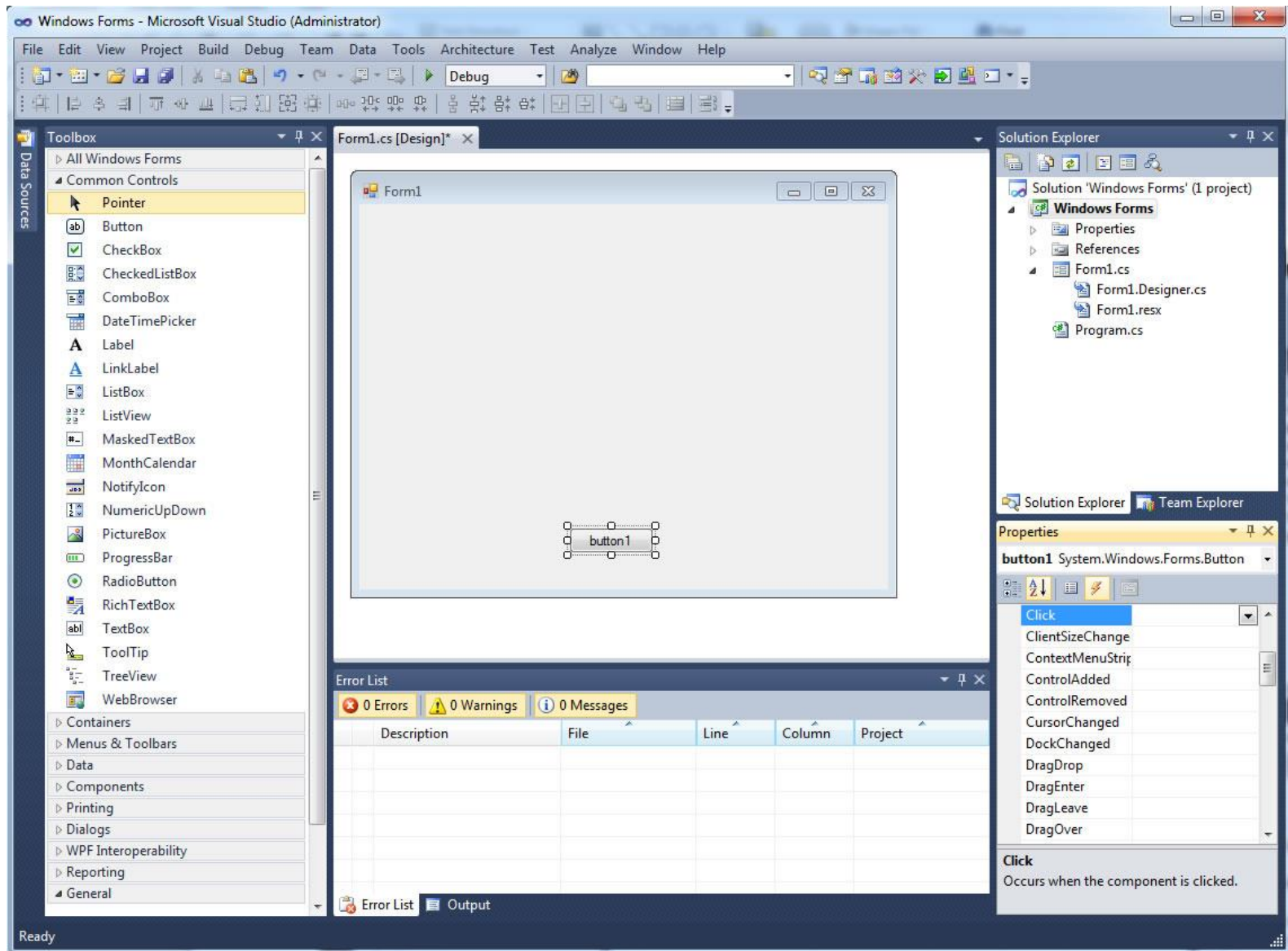
# Agenda

- 1. Einleitung
- 2. Event Handling
- 3. Data Binding

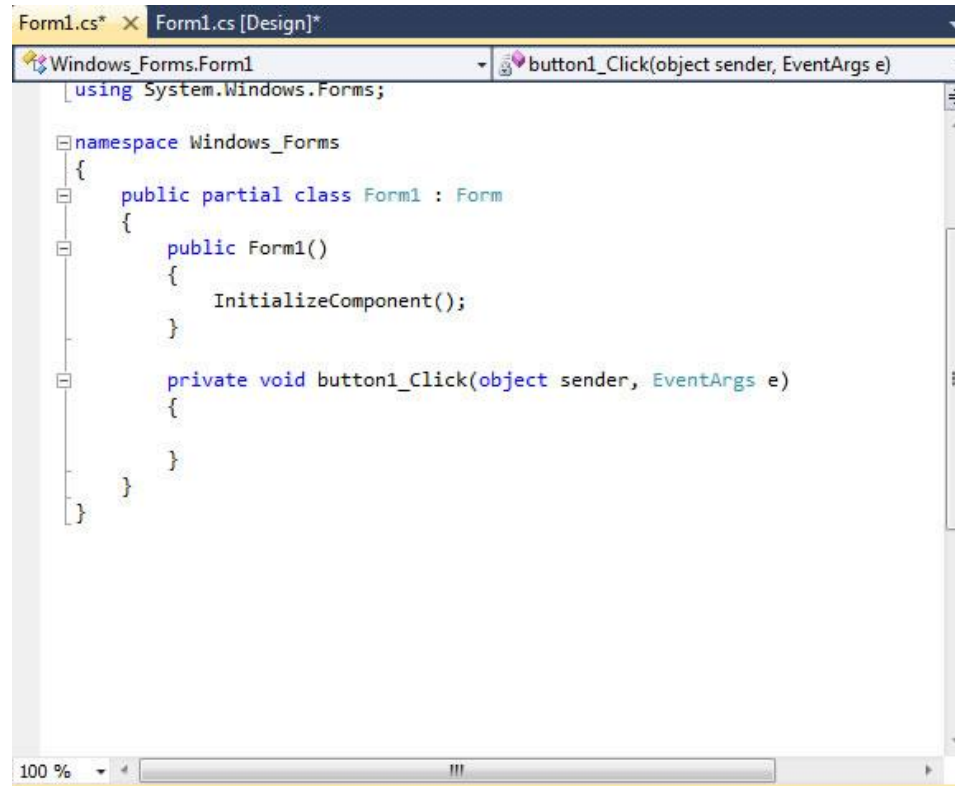
# Windows Forms?

- Bis .NET 2.0 entwickelt
- Leichter GUI entwurf
- Kontrollierten zugriff auf Windows API
- Componenten Bibliothek





```
this.button1.Click += new System.EventHandler(this.button1_Click);
```



The screenshot shows a Visual Studio window with two tabs: 'Form1.cs\*' and 'Form1.cs [Design]\*'. The active tab is 'Form1.cs [Design]\*', which displays the code for the 'button1\_Click' event handler. The code is as follows:

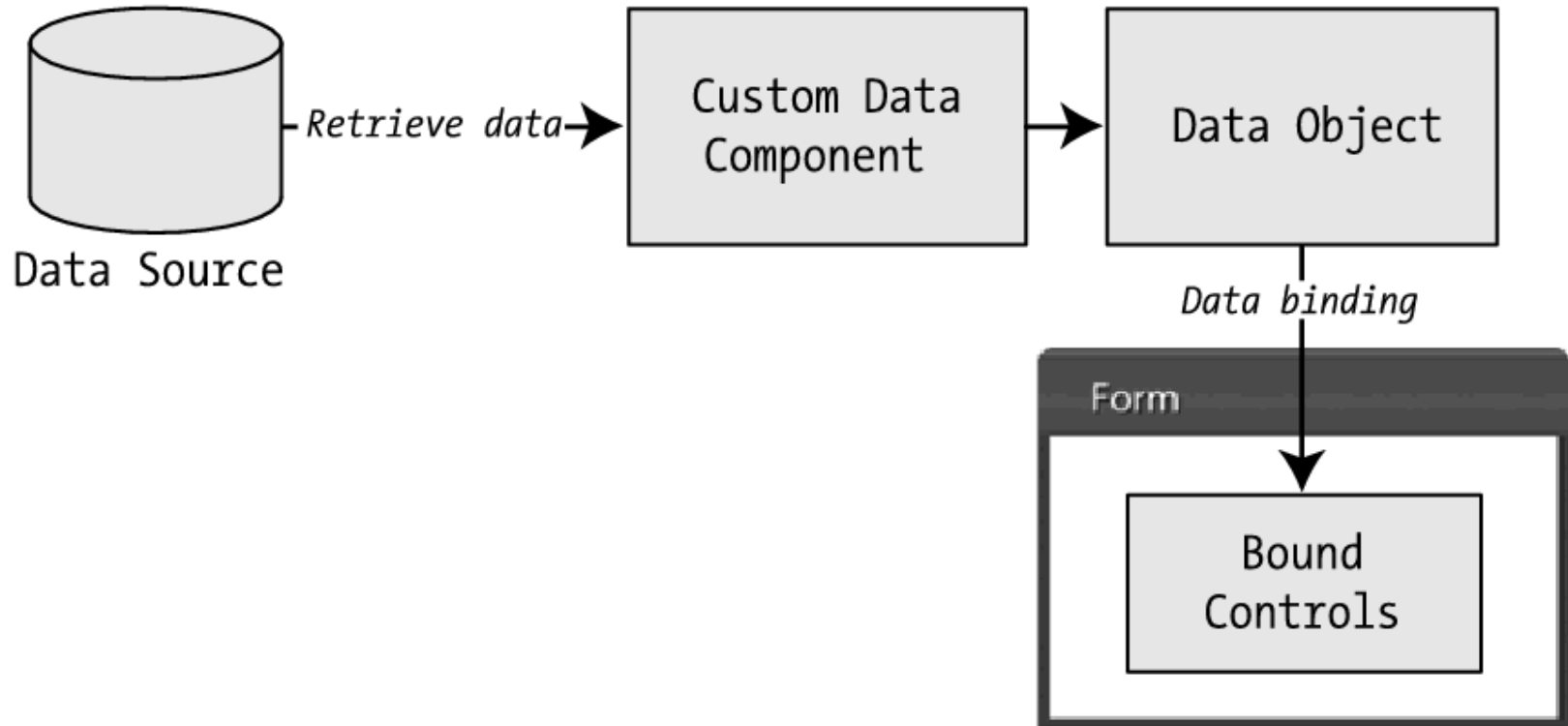
```
using System.Windows.Forms;

namespace Windows_Forms
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
        }
    }
}
```

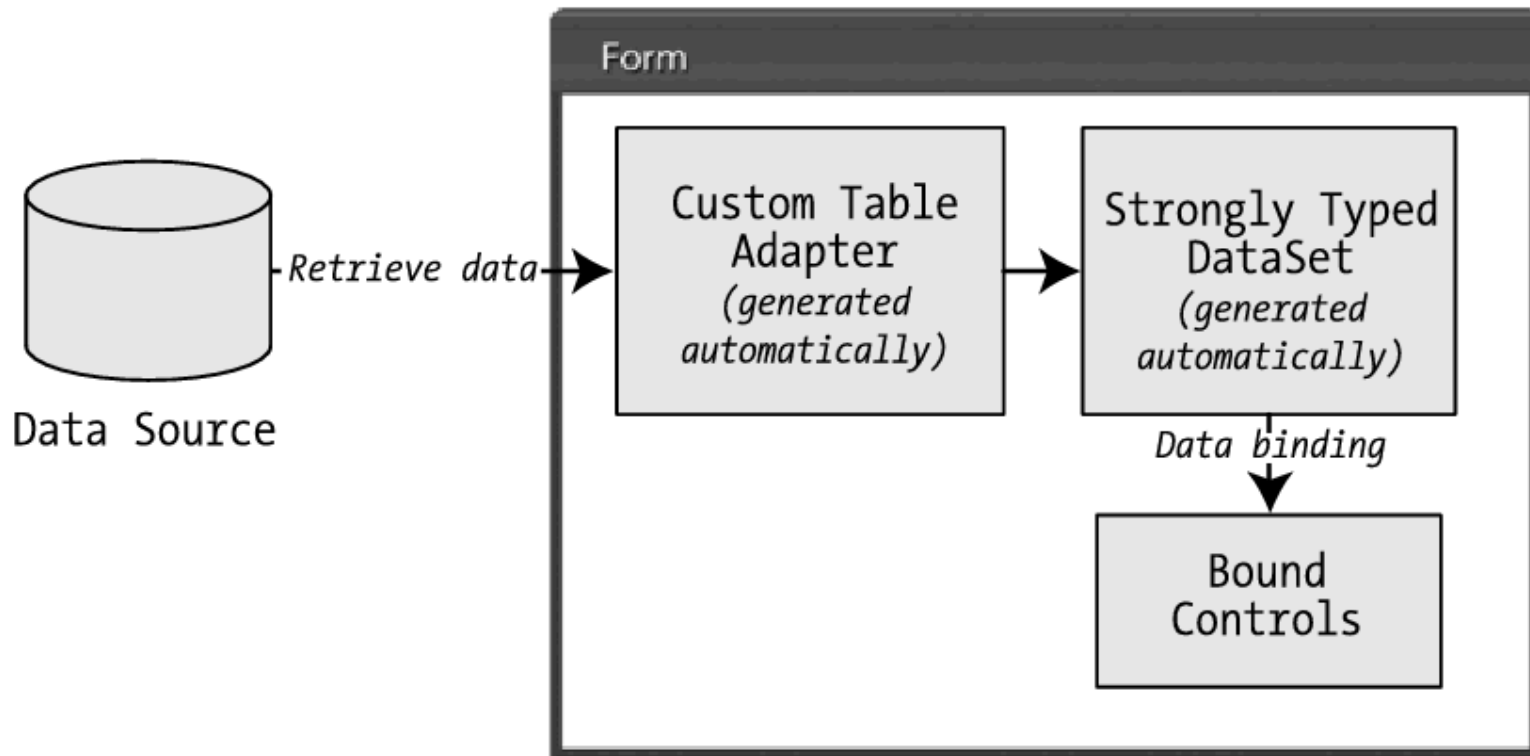
# Data Binding

- Basic Data Binding
- No-code Data Binding

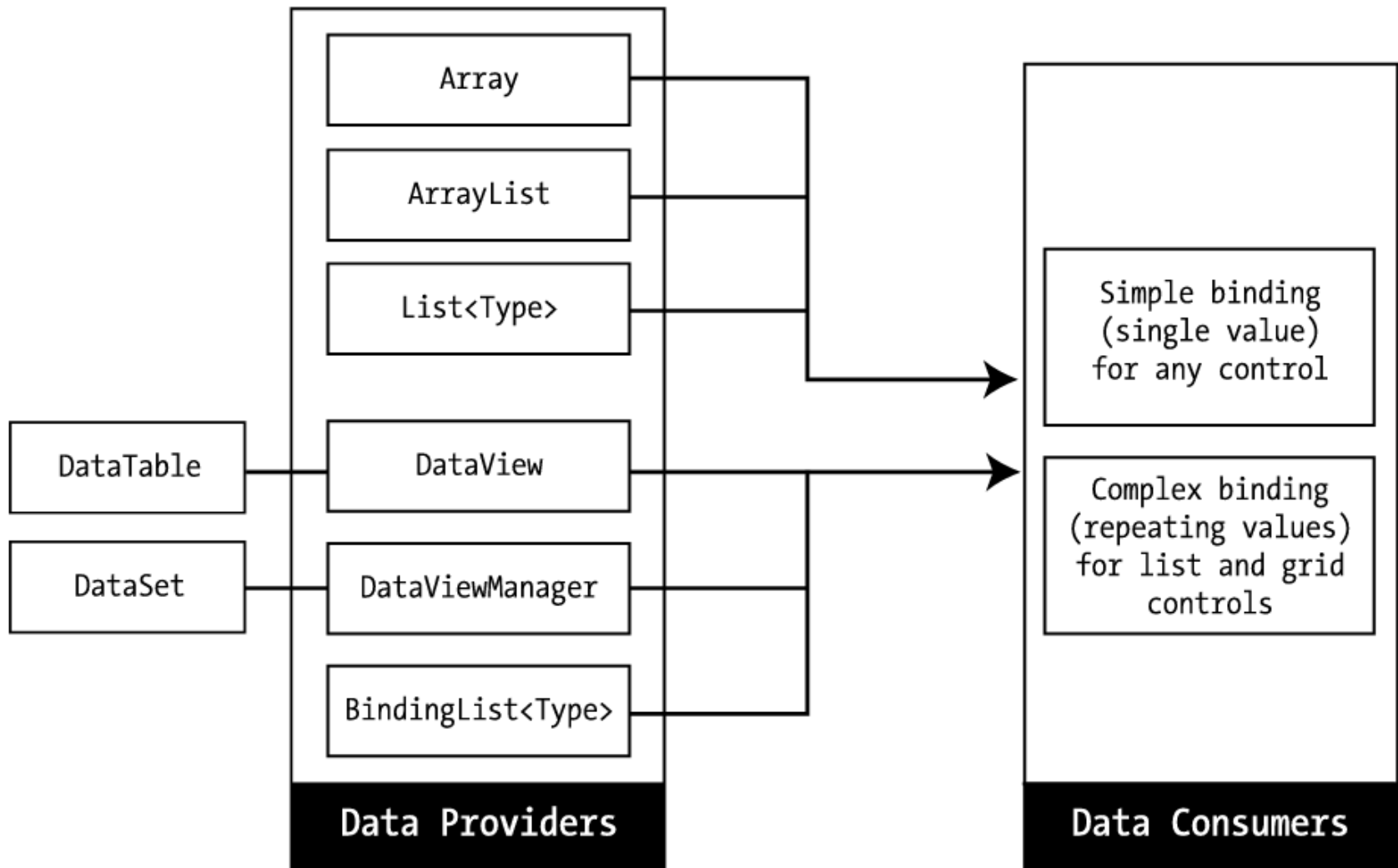


## Basic Data Binding





## No-code Data Binding



# Data Consumers

- Simple Data Binding
  - TextBox.Text
  - Button.Text
  - PictureBox.Image
  
- Complex Data Binding
  - ListBox
  - CheckedListBox
  - ComboBox
  
- Spezialfall DataGridView

# Data Providers

- Jede Klasse die IList implementiert
- Zusätzliche Funktionalität(Optional): IBindingList, IEditableObject, IDataErrorInfo

```
public DataTable GetProducts()
{
    // Get the connection string from the .config file.
    string connectionString = Properties.Settings.Default.Store;
    // Create the ADO.NET objects.
    SqlConnection con = new SqlConnection(connectionString);
    SqlCommand cmd = new SqlCommand("GetProducts", con);
    cmd.CommandType = CommandType.StoredProcedure;
    SqlDataAdapter adapter = new SqlDataAdapter(cmd);
    // Fill a DataTable.
    |
    DataSet ds = new DataSet();
    adapter.Fill(ds, "Products");
    return ds.Tables["Products"];
}
```

## Simple Data Binding

```
private void Form1_Load(object sender, EventArgs e)
{
    Table = GetProducts();

    label1.DataBindings.Add("Text", Table, "ModelName");
    button1.DataBindings.Add("Text", Table, "ModelName");
    textBox1.DataBindings.Add("Text", Table, "ModelName");
}
```

## Complex Data Binding

```
private void Form1_Load(object sender, EventArgs e)
{
    Table = GetProducts();

    comboBox1.DataSource = Table;
    comboBox1.DisplayMember = "ModelName";
    label2.DataBindings.Add("Text", Table, "ModelNumber");
    label3.DataBindings.Add("Text", Table, "CategoryID");
    textBox2.DataBindings.Add("Text", Table, "ModelNumber");
    textBox3.DataBindings.Add("Text", Table, "CategoryID");
    textBox4.DataBindings.Add("Text", Table, "ModelName");
}
```

# DataGridView

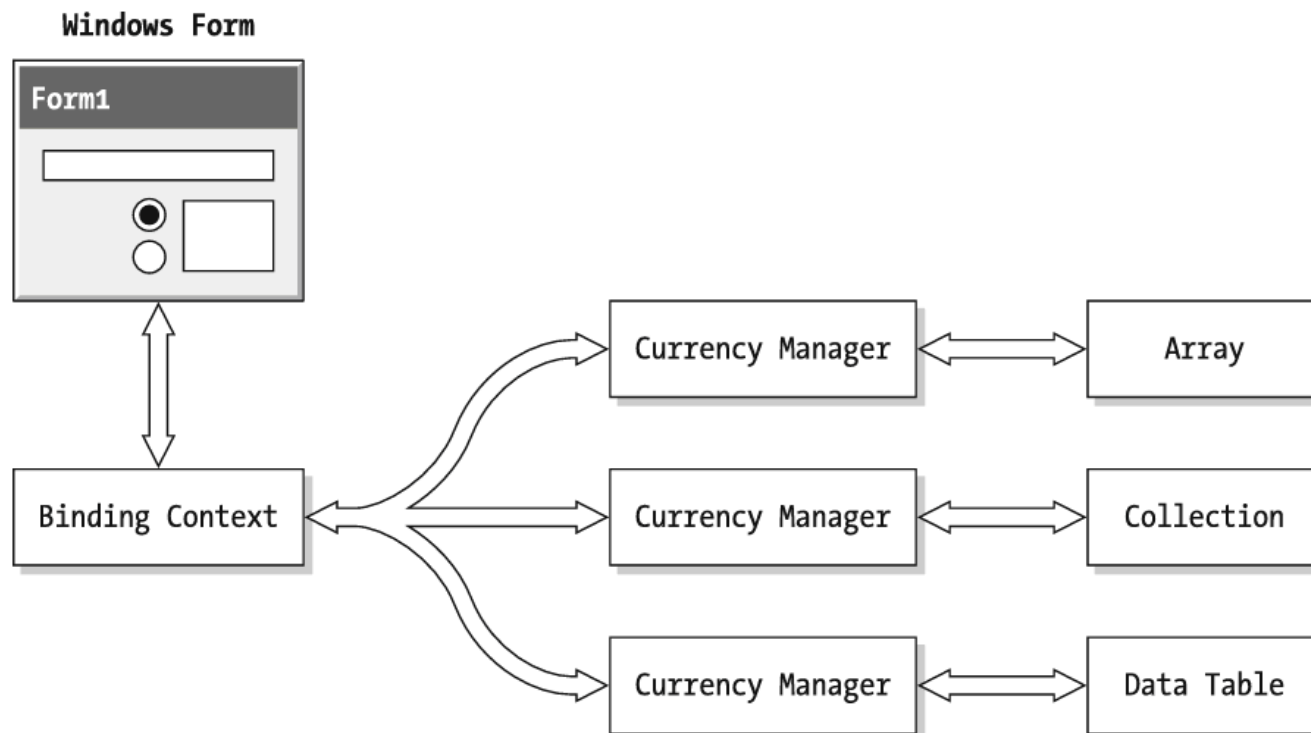
```
private void Form1_Load(object sender, EventArgs e)
{
    Table = GetProducts();

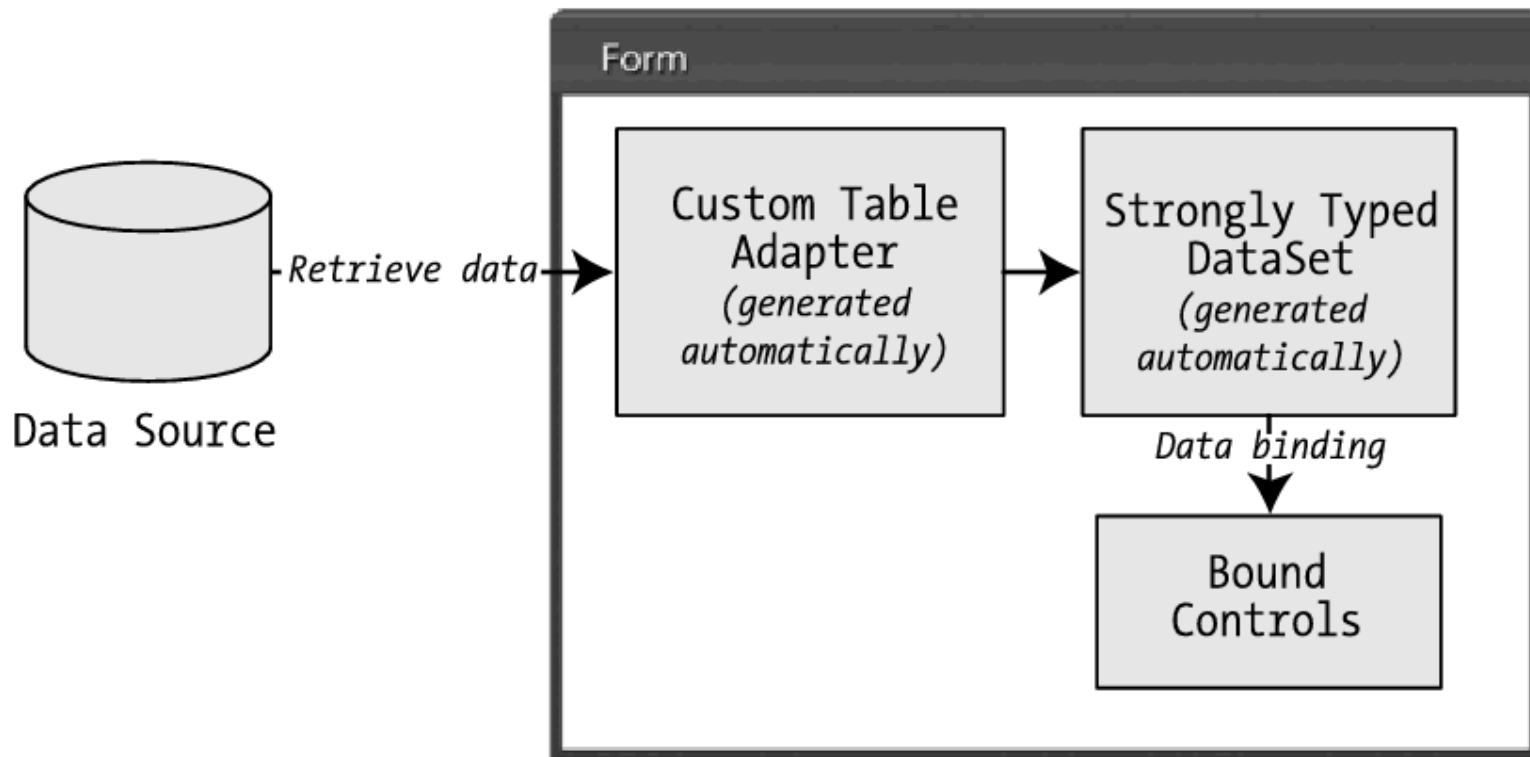
    dataGridView1.DataSource = Table;
}
```



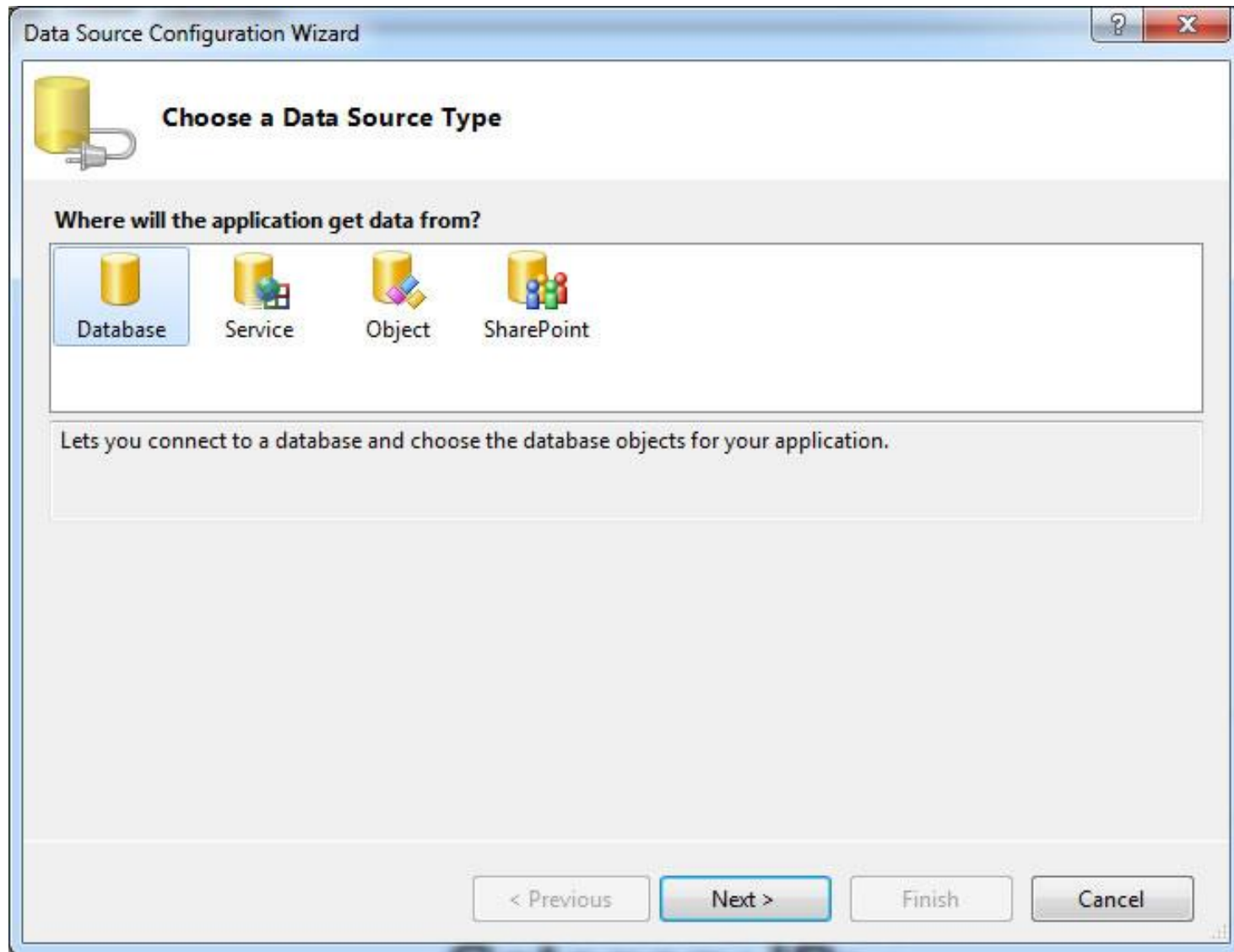
## BindingManagerBase(abstrakt)

- PropertyManager
- CurrencyManager





## No-code Data Binding



```
DataSet ds = Program.StoreDB.GetProducts();  
foreach (DataRow row in ds.Tables["Products"])  
{  
    MessageBox.Show(row["ModelName"].ToString());  
}
```

*// VS*

```
StoreDataSet ds = Program.StoreDB.GetProducts();  
foreach (ProductsDataRow row in ds.ProductsDataTable)  
{  
    MessageBox.Show(row.ModelName);  
}
```

# Fragen?

**Zur gutter letzt**

***Vielen Dank für Ihre Aufmerksamkeit!***