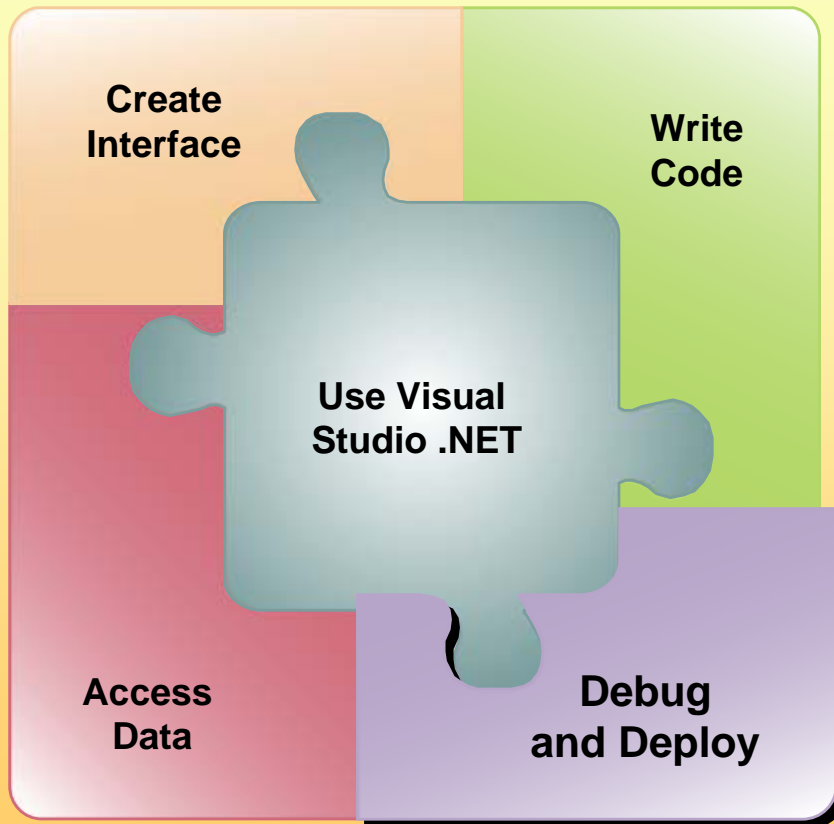


Module 12: Deploying Applications

Overview



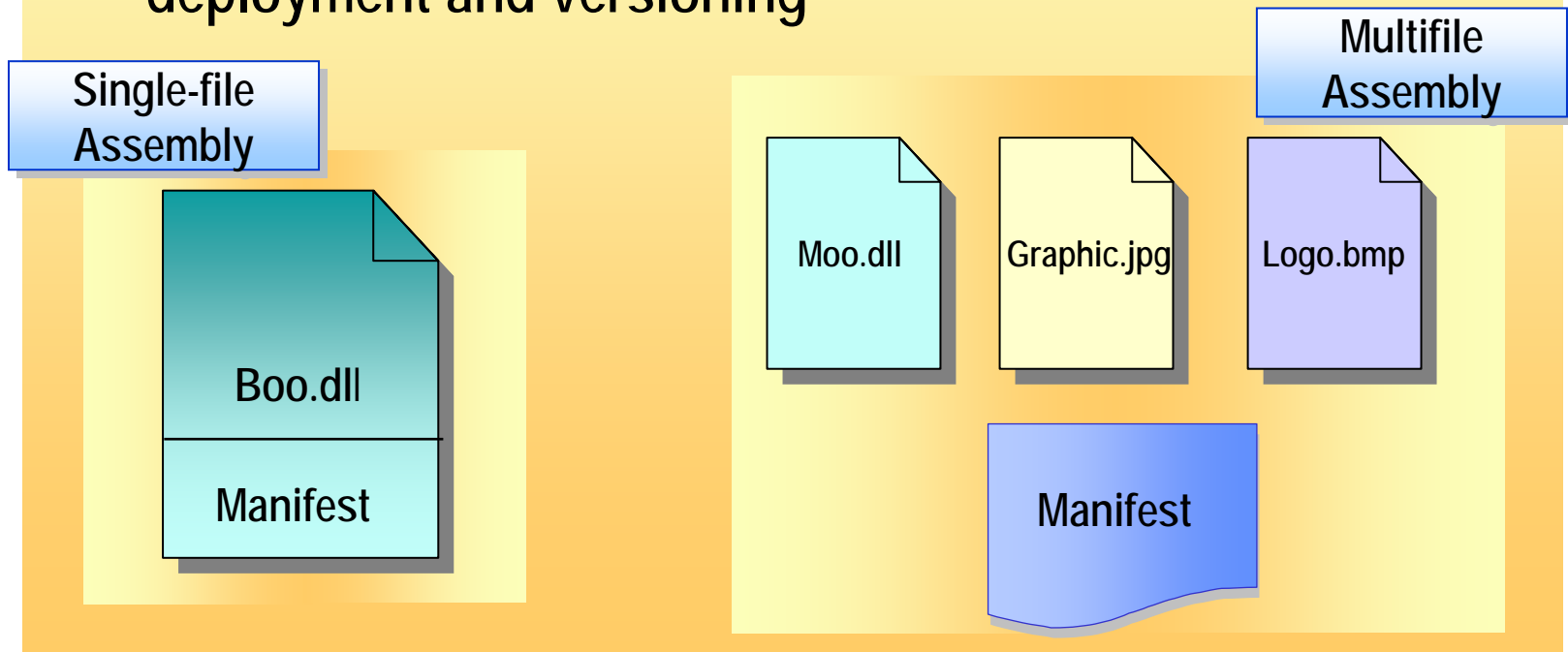
- Introduction to Deployment
- Deploying a Windows-based Application

Lesson: Introduction to Deployment

- What Are Assemblies?
- How Deployment Works in Visual Studio .NET
- How to Choose a Project Type

What Are Assemblies?

- Code must be packaged in a functional unit called an *assembly* before it can be executed by the common language runtime
- Assemblies contain all the information necessary for deployment and versioning



How Deployment Works in Visual Studio .NET

■ Packaging options

- As portable executable files (.dll and .exe files)
- As .cab files
- As a Windows Installer package

■ Deployment

- Based on Windows Installer technology

■ Deployment alternatives

- XCOPY
- The Copy Project command

How to Choose a Project Type

Setup project

- Builds an installer for a Windows-based application (.msi file)

Web Setup project

- Builds an installer file for a Web application (.msi file)

Merge module project

- Packages components that might be shared by multiple applications (.msm file)

CAB project

- Creates a .cab file for downloading to a Web browser

Lesson: Deploying a Windows-based Application

1 Create a setup project

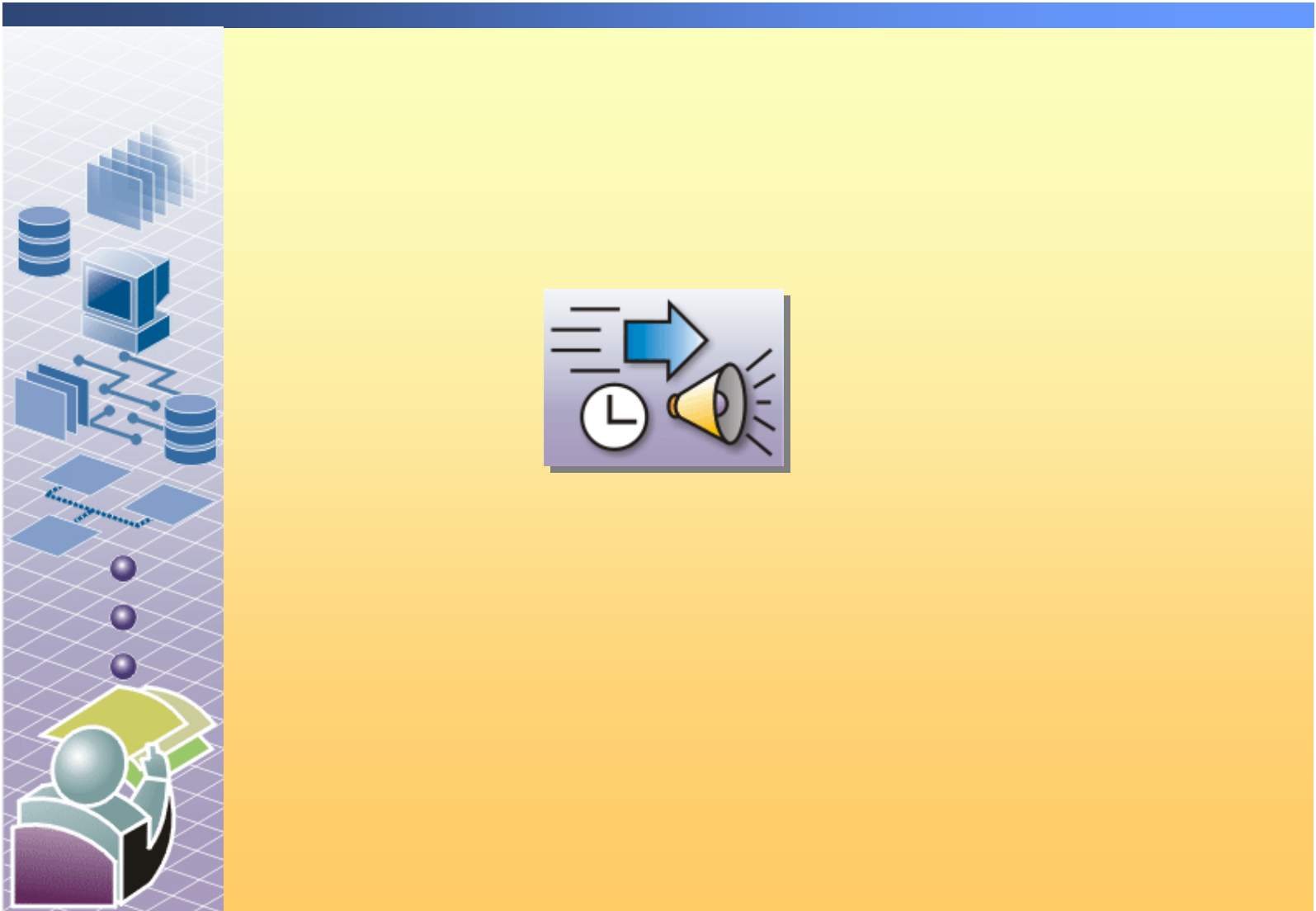
2 Set project properties

3 Customize the setup project

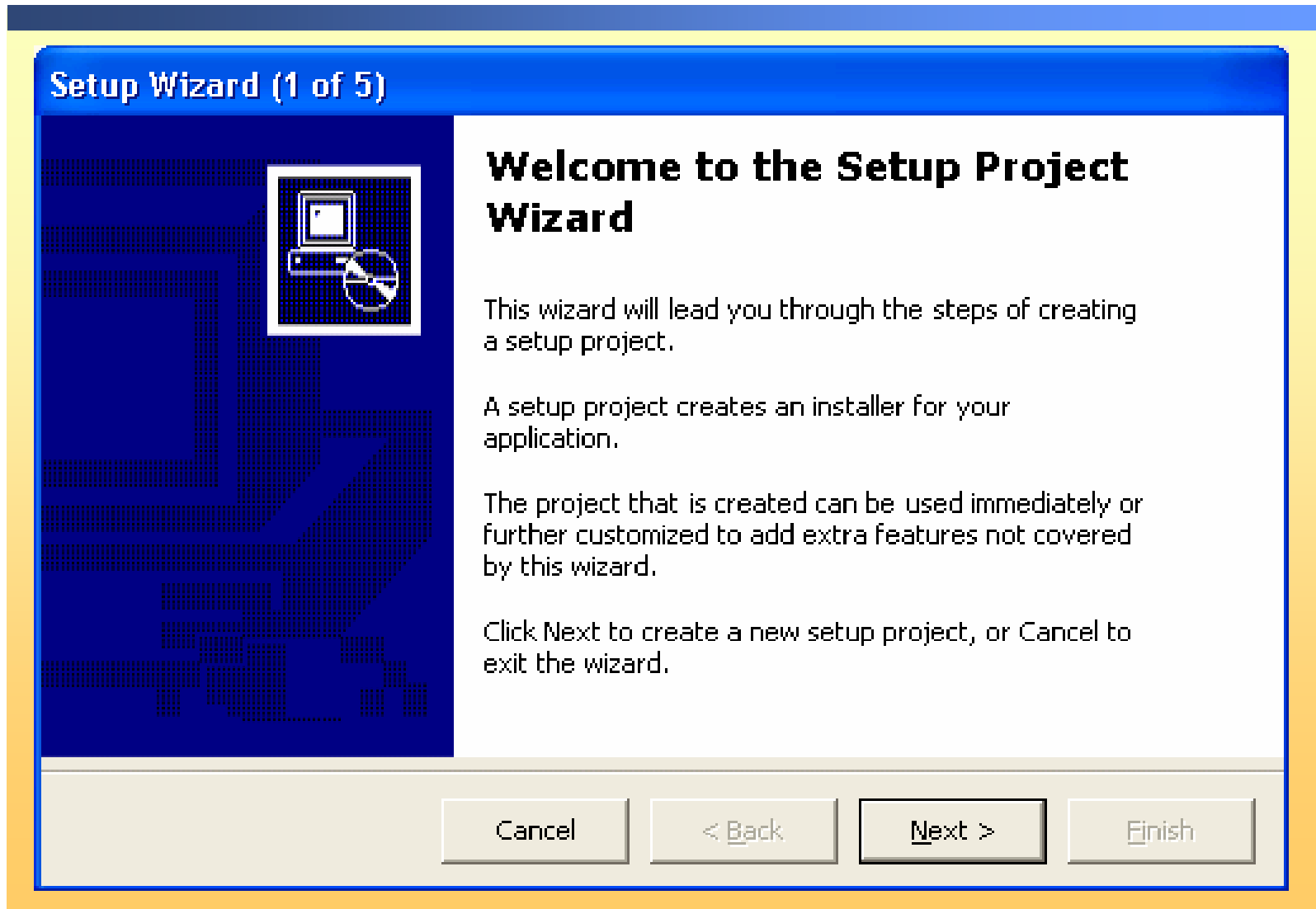
4 Build the application

5 Distribute and install the application

Multimedia: Deploying a Windows-based Application



How to Create a Windows Installer Setup Project



How to Set Project Properties

■ Properties window

- Set general properties
- Examples: **Author, Manufacturer, ProductName**

■ Deployment Properties dialog box

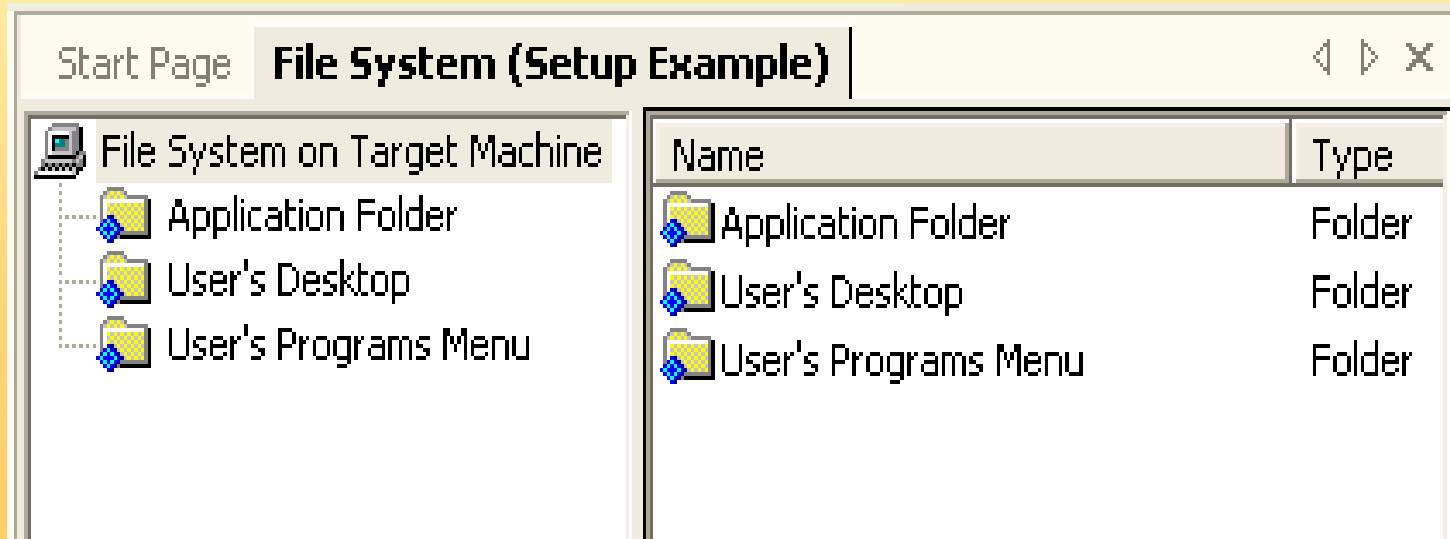
- Set configuration-dependent properties
- Examples: **Bootstrapper, Compression, Output file name, Package files**
- When installing to versions of Windows previous to Windows XP for the first time, you must include a bootstrapper in the installer

Deployment Editors

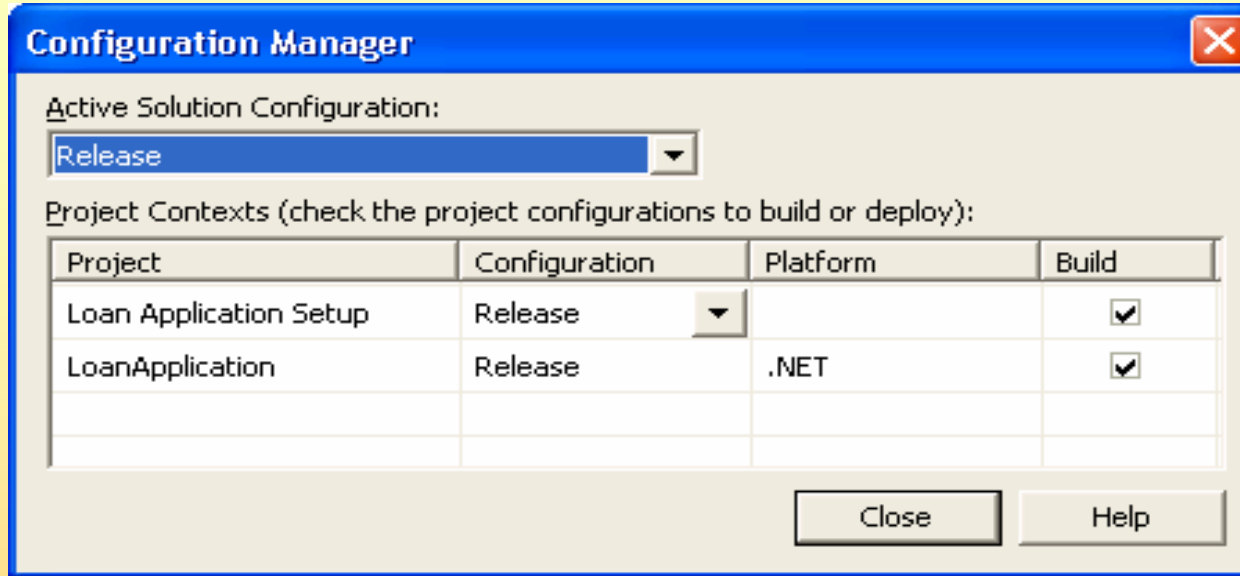
- Use the deployment editors to configure the installer that you create
 - File System Editor
 - Registry Editor
 - File Types Editor
 - User Interface Editor
 - Custom Actions Editor
 - Launch Conditions Editor

How to Use the File System Editor

- Provides a representation of the file system on a destination computer
- Uses the concept of abstract folders to ensure that files are installed where you want them

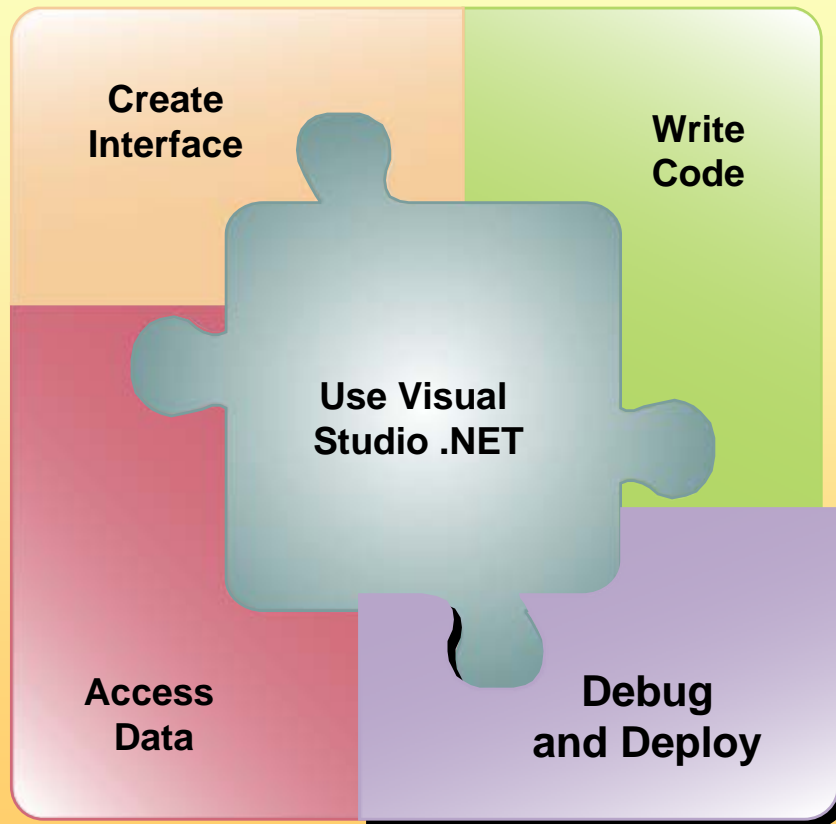


How to Build the Setup Project



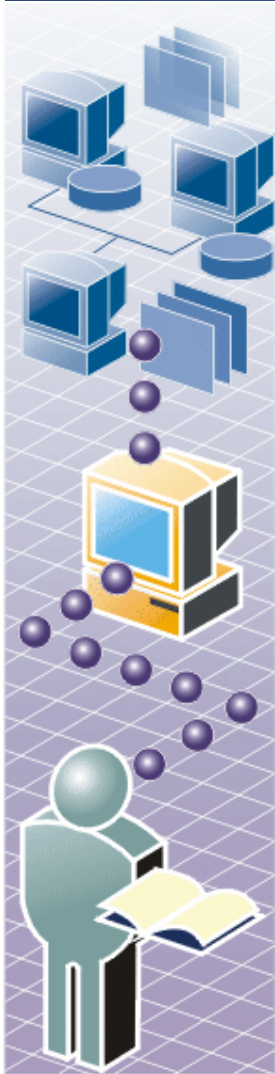
- The .NET Framework must be installed on any computer that will run an application created with Visual Studio .NET
- To install the .NET Framework, use the redistributable installer Dotnetfx.exe

Review



- Introduction to Deployment
- Deploying a Windows-based Application

Lab 12.1: Deploying an Application



- Exercise 1: Using the Setup Wizard
- Exercise 2: Installing the Loan Application

Course Evaluation

