ClickOnce Deployment
(One-click Deployment)

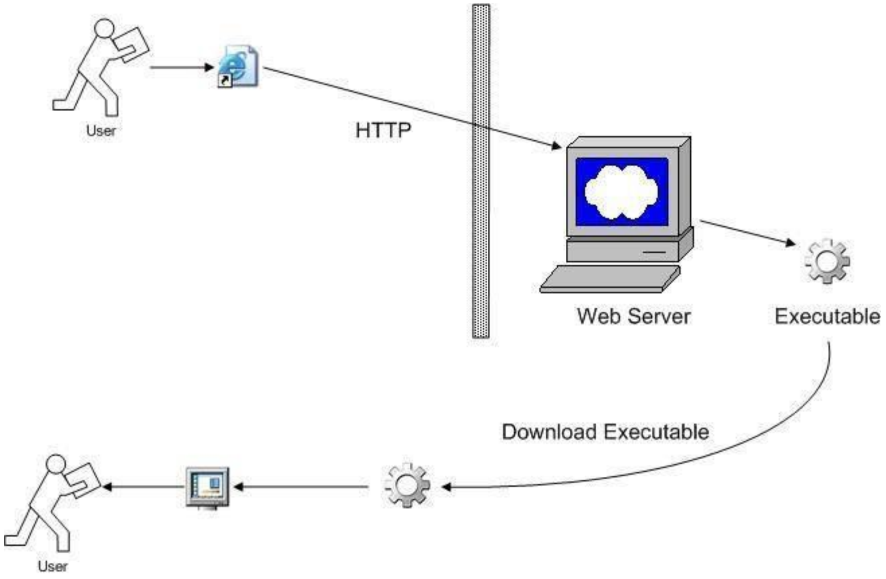
James W. Fawcett, Instructor
Murat K. Gungor, TA
Syracuse University

Goal

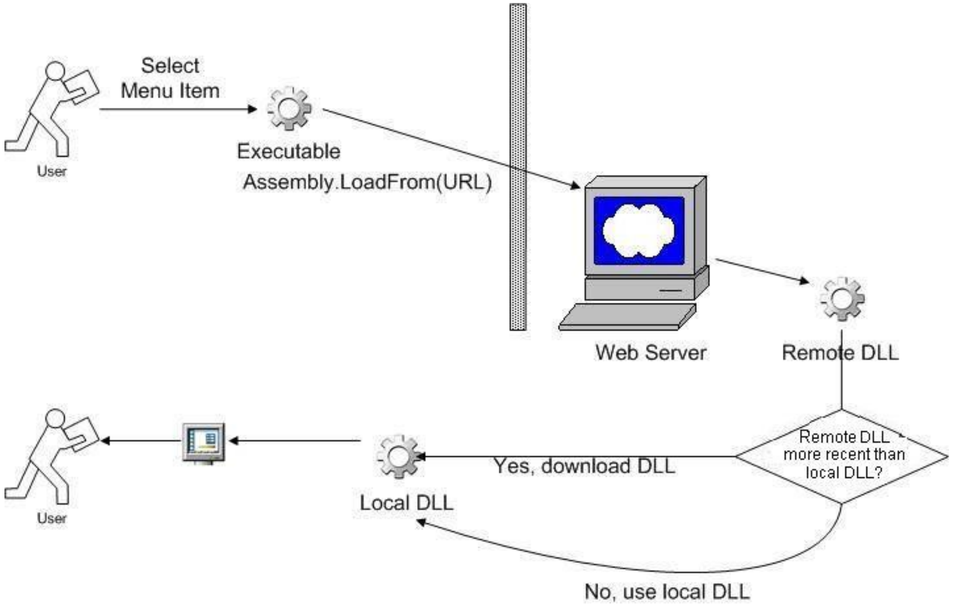
- Have a .Net Windows Forms application act like Web Application from a deployment perspective.
    - Launch a smart client application directly off a Web URL just like a Web page.
- Minimize effort required to deploy and update applications.
- Primary purpose is to present a rich user interface that talks to back-end services on the network.

Auto-Deploying an Executable

- Users can run the executable by pointing a browser to the URL.
- Users can also run the executable by entering the URL via the Run option in the Start menu.
- Or, you can create a Web page with an HTML HREF tag referencing the URL.



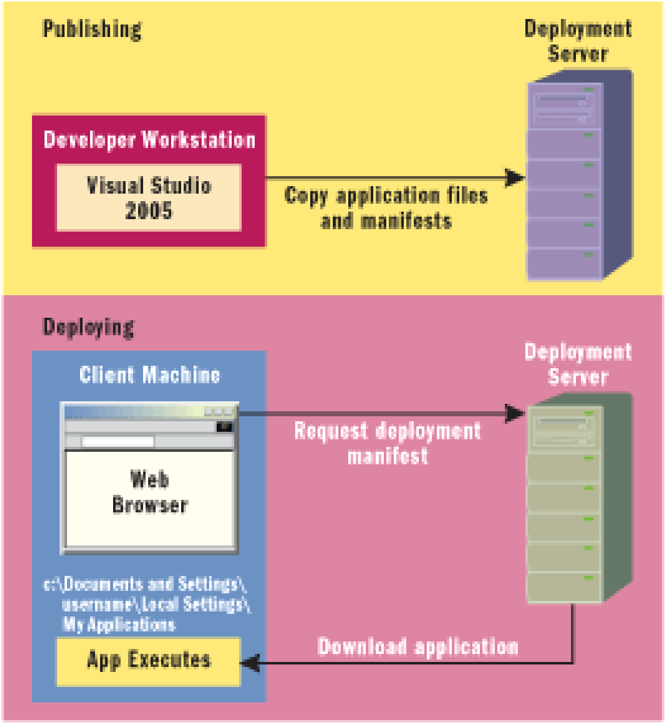
Advantages



- It’s easier to build complex and good looking User Interfaces with Win Forms than with Web forms.
- Update is easy, just upload new version to the server.
- If an application is large, executable can download libraries for the rest of the application on an as-needed basis.

Deployment Option (online/offline)

- ClickOnce allows applications to be deployed offline as well, making the app behave as if it had been installed on the local machine.

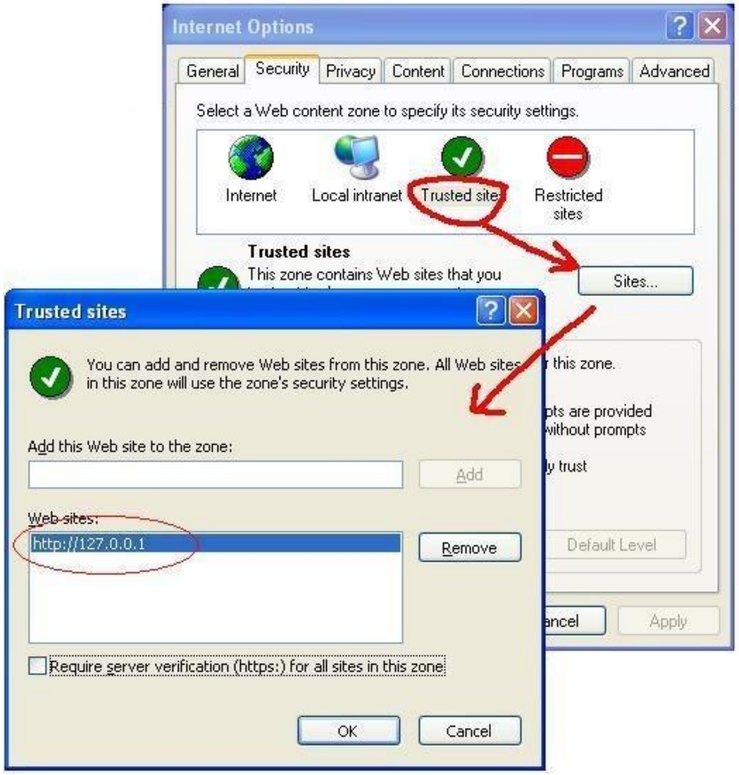


ClickOnce is not for you, if …

- If a program needs to carry out privileged operations that could affect other applications or data on the target machine, such as performing unrestricted file access or accessing the registry, then it may not be suitable for deployment using ClickOnce.
- Additionally, if an application needs privileged access to the local machine at install time, then it is definitely not a candidate for ClickOnce deployment because by design there are no mechanisms in ClickOnce to add registry entries, install shared assemblies in the Global Assembly Cache (GAC), install services, and so on

It's All About Security

- With .NET Security Policies, you can protect users from accidentally running code from unknown sources.



Select a Web content zone to specify its security settings.

Trusted sites

This zone contains Web sites that you

this zone.

pts are provided
without prompts

ly trust

You can add and remove Web sites from this zone. All Web sites in this zone will use the zone's security settings.

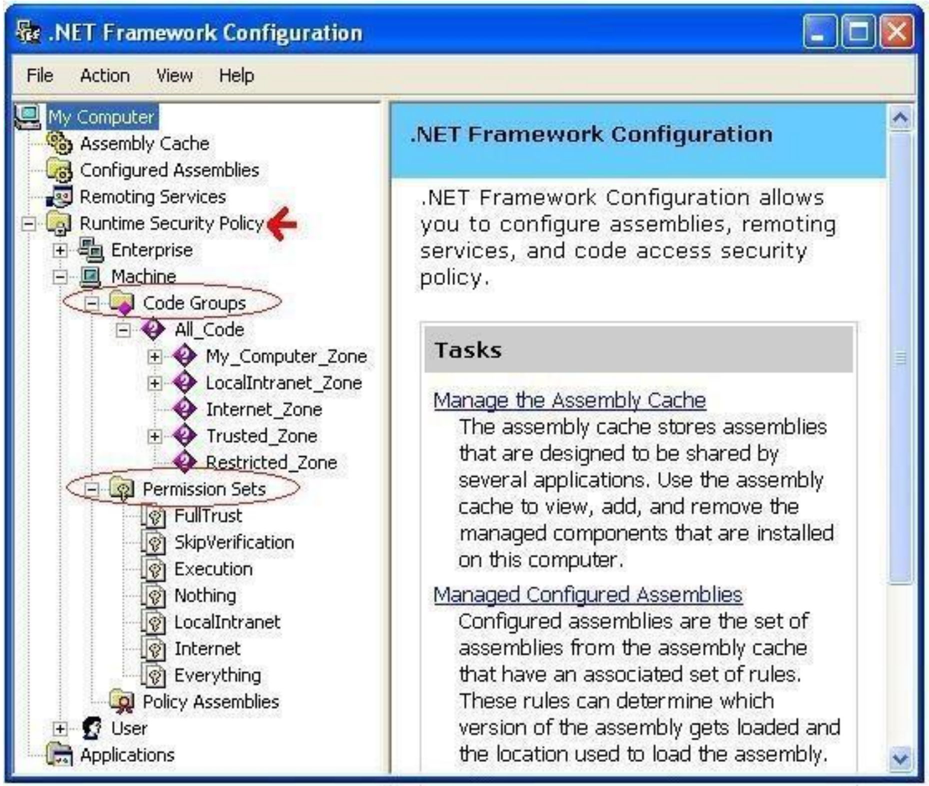
Add this Web site to the zone:

Web sites:

http://127.0.0.1

☐ Require server verification (https:) for all sites in this zone

.NET Security and the Framework Configuration Tool



.NET Framework Configuration

.NET Framework Configuration allows you to configure assemblies, remoting services, and code access security policy.

Tasks

Manage the Assembly Cache

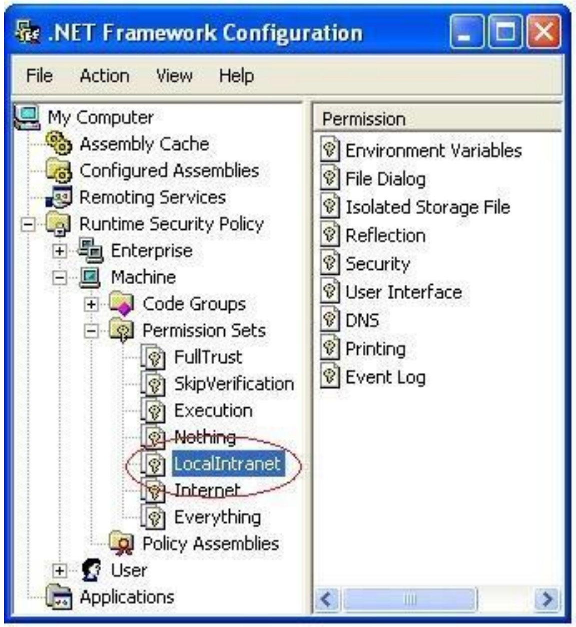
The assembly cache stores assemblies that are designed to be shared by several applications. Use the assembly cache to view, add, and remove the managed components that are installed on this computer.

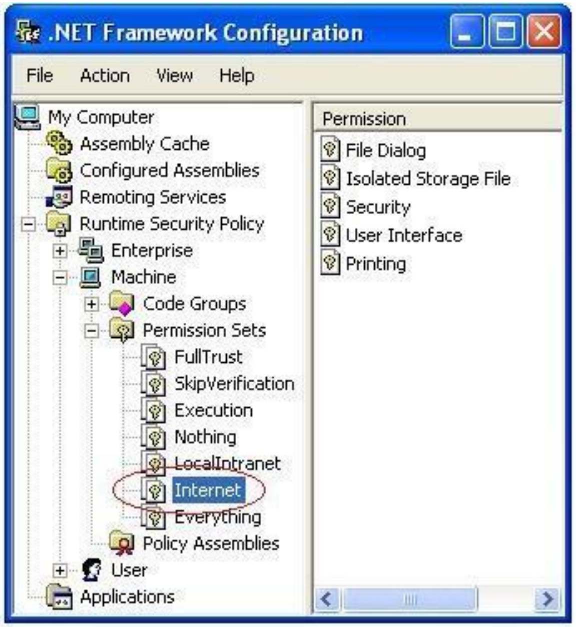
Managed Configured Assemblies

Configured assemblies are the set of assemblies from the assembly cache that have an associated set of rules. These rules can determine which version of the assembly gets loaded and the location used to load the assembly.

- To launch it, go to the Windows Control Panel, select Administrative Tools, then choose Microsoft .NET Framework Configuration.
- This tool is part of the .NET Framework (not VS.NET), which means that every user with the .NET Framework installed has the Configuration Tool installed as well.
- Has two parts:
  - Code Groups, and
  - Permission Sets

.NET Runtime Security Policy





- You can define custom Code Groups and Permission Sets by right-clicking in the appropriate node.

Security and Permissions

- Its developer’s responsibility to ensure that the application will run with the appropriate permissions.
- To prevent users from being prompted, a System Administrator can specify a ClickOnce deployment policy that defines a specific application publisher as a trusted source. On computers where this policy is deployed, permissions will automatically be granted and the user will not be prompted.

Elevating Security Permissions

- To allow the application to obtain the permissions it needs without administrator intervention on the client machine, ClickOnce can prompt the user at installation time to ask them to grant the application the elevated permissions.
- Note that this capability circumvents some of the security protections provided by the CLR by allowing a user to elevate security permissions for that application.
- You can prevent the user from being allowed to do this through security policy in an enterprise environment.
- Allowing the user to elevate permissions is an important capability for applications deployed over the Internet, where no administrators may be available.

ClickOnce Deployment and Authenticode

- A digital certificate is a file that contains a cryptographic public/private key pair, along with metadata describing the publisher to whom the certificate was issued and the agency that issued the certificate.
- You can obtain a certificate for code signing in one of three ways:
  - Purchase one from a certificate vendor.
  - Receive one from a group in your organization responsible for creating digital certificates.
  - Generate your own certificate with MakeCert.exe, which is included with the .NET Framework Software Development Kit (SDK).
- **Certification Authorities**
  - Certification authorities (CAs) are trustworthy persons or organizations that issue certificates to applicants whose identity has in some way been verified by the CA.

Application Deployment Prerequisites

- The minimum required version of the common language runtime.
- The minimum required version of the Windows operating system required by the application.
- The minimum version of any and all assemblies that must be preinstalled in the global assembly cache (GAC).
- All above are specified as assembly dependencies in the application manifest.

Bootstrapping for ClickOnce Applications

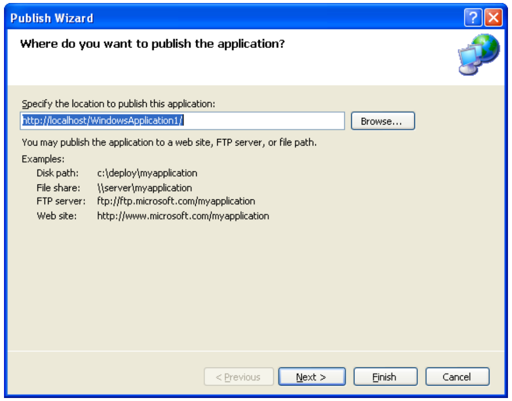
- Visual Studio includes the following deployable packages:
  - Microsoft .NET Framework version 2.0
  - Microsoft Data Access Components (MDAC) 2.8
  - Microsoft Visual J# .NET Framework Redistributable Package 3.1
  - SQL Server 2005 Express Edition
  - Windows Installer 2.0

Accessing Local and Remote Data in ClickOnce Applications

- With ClickOnce, you can load and store data locally by using any one of the following methods:
  - ClickOnce Data Directory
  - Isolated Storage
  - Other Local Files

How to Deploy a Win Form App.

- Using Visual Studio
  - Select Project | Publish Project, and
  - Publish Wizard asks from you to specify the location for the publishing server.



Where do you want to publish the application?

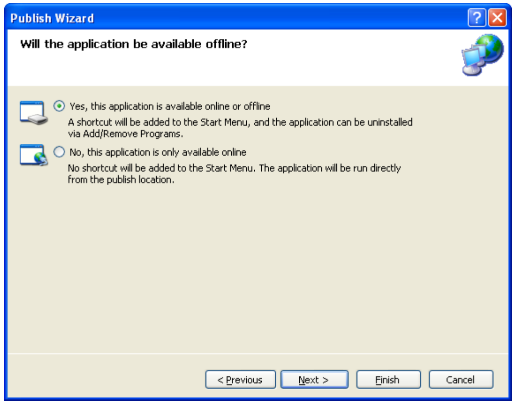
Specify the location to publish this application:

http://localhost/WindowsApplication1/

You may publish the application to a web site, FTP server, or file path.

Examples:

Disk path:	c:\deploy\myapplication
File share:	\\server\myapplication
FTP server:	ftp://ftp.microsoft.com/myapplication
Web site:	http://www.microsoft.com/myapplication



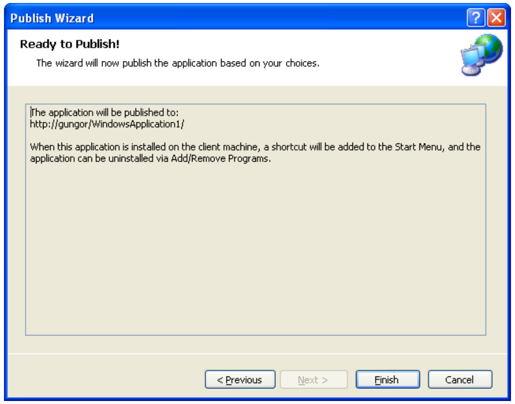
Will the application be available offline?

● Yes, this application is available online or offline

A shortcut will be added to the Start Menu, and the application can be uninstalled via Add/Remove Programs.

○ No, this application is only available online

No shortcut will be added to the Start Menu. The application will be run directly from the publish location.



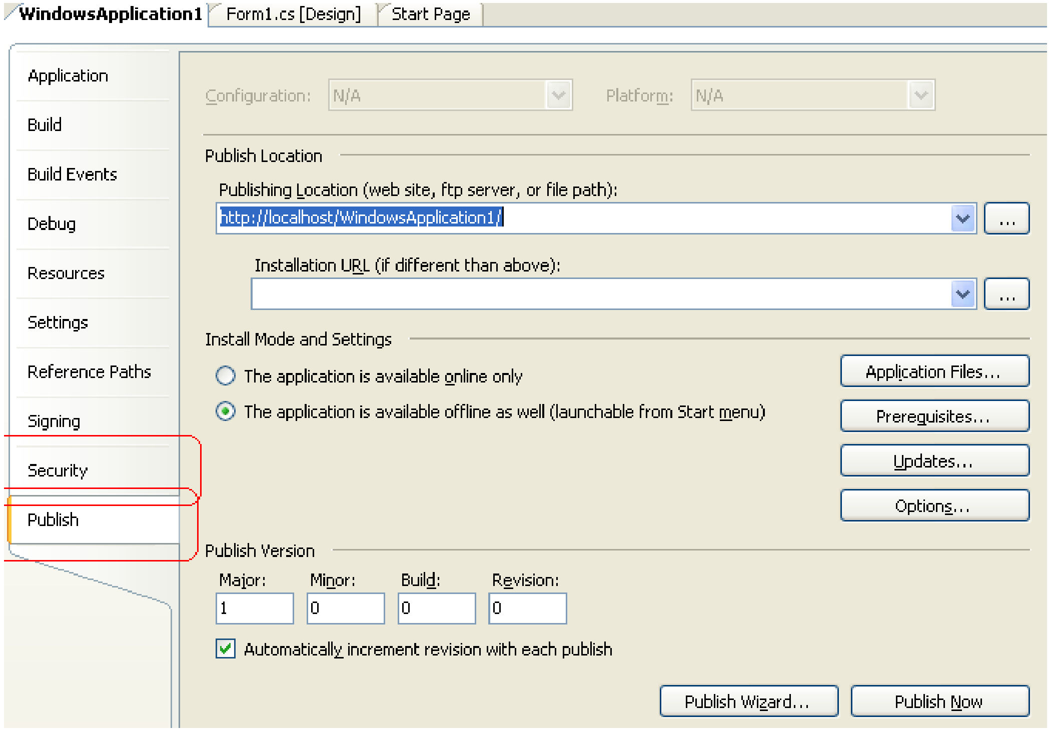
Ready to Publish!

The wizard will now publish the application based on your choices.

The application will be published to:
http://gungor/WindowsApplication1/

When this application is installed on the client machine, a shortcut will be added to the Start Menu, and the application can be uninstalled via Add/Remove Programs.

Necessary Settings for Deployment



Configuration: N/A Platform: N/A

Publish Location

Publishing Location (web site, ftp server, or file path): http://localhost/WindowsApplication1/

Installation URL (if different than above):

Install Mode and Settings

○ The application is available online only

● The application is available offline as well (launchable from Start menu)

Publish Version

Major:	Minor:	Build:	Revision:
1	0	0	0

☑ Automatically increment revision with each publish

- Project | Properties
- Security and Publish Tabs are important tools for successful deployment.

Resources

- Chris Sells, Michael Weinhardt, Windows Forms 2.0 Programming, Addison – Wesley, 2006.
- Deploy and Update Your Smart Client Projects Using a Central Server

  http://msdn.microsoft.com/msdnmag/issues/04/05/ClickOnce/
- http://msdn.microsoft.com/vbasic/learning/clickonce/
- http://www.code-magazine.com/Article.aspx?quickid=0307061
- Rick Strahl's WebLog
  http://west-wind.com/weblog/

End of Presentation