下载并安装 Ultimate 版本 ed2k://|file|en\_visual\_studio\_2010\_ultimate\_x86\_dvd\_509116.iso|2446950400|0694CFA0756D8 C11499DE5B42E205410|/

在干净的 Win2003 上安装会提示需要先装 WIC 组件

不装 VB、VC、F#等组件, 只装 C#也要 4G 的空间!因为要安装 VC Runtime、.NET Framework 4 等组件。

安装后新建项目,提示"Error writing the project file. 没有注册类别",这是因为操作系统中 不存在 MSXML6 造成的,下载 MSXML6 安装后,再建立新项目的时候就不会出现上面的 错误了。

MSDN 包含在光盘里面, ProductDocumentation 文件夹, 原来叫 msdnlibrary, 现在叫 helplibrary, 基于 web 的, helplibrary 无法独立安装, 必须要先安装 vs2010, 安装好后打开, 所有程序->vs2010->vs2010tools->manage help setting, 选择 install content from disk, 然后选择光盘里面的 ProductDocumentation 文件夹下的 HelpContentSetup.msha

安装后发现有专门的 TestManager 2010 ,但是需要连接 TFS(Team Foundation Server)。

新建测试项目,可以添加各种类型的测试,包括单元测试、负载测试、数据库单元测试、界面测试、Web性能测试等。



Coded UI Test 的脚本可以通过录制产生。





录制时提示安装后需要重启系统才能使用录制功能。



## 重启操作系统即可

录制之后可产生脚本:

Coded UI Test Builder - Generate Code	? ×
Method Name: (for example: MyMethod)	
RecordedMethod1	•
Add and <u>G</u> enez	rate

public void RecordedMethod1()

{

#region Variable Declarations

WinTitleBar uI 计算器 TitleBar = this.UI 计算器 Window.UI 计算器 TitleBar; WinButton uIItem1Button = this.UI 计算器 Window.UIItem1Window.UIItem1Button; WinButton uIItemButton = this.UI 计算器 Window.UIItemWindow.UIItemButton; WinButton uIItemButton1 = this.UI 计算器 Window.UIItemWindow1.UIItemButton; #endregion

// Click '计算器' title bar Mouse.Click(uI 计算器 TitleBar, new Point(91, 12));

// Click '1' button
Mouse.Click(uIItem1Button, new Point(21, 13));

// Click '+' button
Mouse.Click(uIItemButton, new Point(26, 16));

// Click '1' button
Mouse.Click(uIItem1Button, new Point(17, 6));

// Click '=' button
Mouse.Click(uIItemButton1, new Point(15, 10));

}

可通过 Coded UI Test Builder 的 crosshair 来添加检查点, windows logo 键+I 可选择要检查的 控件。

<ul> <li>○ UII+算器Window</li> <li>□ UII+算器TitleBar</li> <li>○ UIItemIWindow</li> <li>○ UIItemIWindow</li> <li>○ UIItemIButton</li> <li>○ UIItemWindow</li> <li>○ UIItemWindow</li> <li>○ UIItemWindow</li> <li>○ UIItemButton</li> <li>○ Octorol Specific</li> <li>○ HelpText</li> <li>○ Add Assertion</li> <li>○ Control Name</li> <li>○ Control Id</li> <li>○ DisplayText</li> <li>○ Shortout</li> <li>○ Generic</li> <li>○ ClassName</li> <li>○ Button</li> </ul>	Coded UI Test Builder - Add Assertions: UIItem1Butt	ton ?	х
□ UI计算器Window UI计算器TitleBar □ UIItemIWindow UIItemButton □ UIItemButton □ UIIte		» → Add Assertion	
	<ul> <li>□ UI计算器Window</li> <li>□ UIItem1Window</li> <li>□ UIItem1Button</li> <li>□ UIItemWindow</li> <li>□ UIItemButton</li> <li>□ UIItemButton</li> <li>□ UIItemButton</li> </ul>	Froperty Value          Search       ControlType       Button         TechnologyName       MSAA         Name       1         Control Specific       HelpText         AccessKey       ControlName         ControlId       0         DisplayText       Shortcut         Generic       ClassName       Button	

```
public void AssertMethod1()
```

```
#region Variable Declarations
WinEdit uIItemEdit = this.UI 计算器 Window.UIItemWindow2.UIItemEdit;
#endregion
// Verify that 'Unknown Name' text box's property 'Text' equals '2. '
```

Assert.AreEqual(this.AssertMethod1ExpectedValues.UIItemEditText, uIItemEdit.Text);

}

{

其他参考资料:

```
http://blogs.msdn.com/mathew_aniyan/pages/content-index-for-coded-ui-test.aspx
http://blogs.msdn.com/vstsqualitytools/archive/2009/06/12/automated-user-interface-testing-with-
coded-ui-test.aspx
http://www.sqaforums.com/postlist.php?Cat=0&Board=UBB58
http://www.testautomationfx.com/?gclid=CK_-zLWx2KECFRMsbwod8zi7Jw
```

TIB 自动化测试工作室 http://www.cnblogs.com/testware/