

FirstButton™ 文档

注意：为避免 Windows Vista 或 Windows7 中可能出现的 Program Compatibility 信息，因为FirstButton 已通过 Visual Studio2010 重新编译，所以兼容性部分已包括在应用程序的列单中。请参阅第8页了解更多信息。

原始程序是通过 Visual Studio 2005 编译而成的。通过使用 VS 2005 将 6 个自定义的图标添加到可执行文件中（添加过程在此文档中有所描述。）

示例文件包包括：

logoinst.sln logoinst.suo logoinst.vbproj logoinst.vbproj.user registry helpers.vb Resource1.Designer.vb Resource1.resx tgt_inst.Designer.vb tgt_inst.resx tgt_inst.vb ~~tgt_license.Designer.vb~~ ~~tgt_license.resx~~ tgt_license.vb AssemblyInfo.vb

许可屏幕含有大量 Liquidity Lighthouse, LLC 公司特殊的信息，但这并未包括在文件包中。强烈建议你在允许继续安装之前，建立一个类似的屏幕，以要求自由软件基金会的许可证的用户协议。

需要 .NET 框架组件库来进行编译：

这些并不是这个组件的分配，也不是 FirstButton™ 源代码的 GNU 许可证，而是需要它们成功实现通过 Visual Studio 来进行程序编译。如果需要 Windows98 的向后兼容性，则应该使用组件的旧版本，如 2.0.0.0版。

Reference Name	DLL	Version
Microsoft.VisualBasic.Compatibility	Microsoft.VisualBasic.Compatibility.dll	8.0.0.0
System	System.dll	2.0.0.0
System.Data	System.Data.dll	2.0.0.0
System.Drawing	System.Drawing.dll	2.0.0.0
System.Windows.Forms	System.Windows.Forms.dll	2.0.0.0
System.Xml	System.Xml.dll	2.0.0.0

变量

这些变量的定义在 FirstButton 程序tgt_inst.vb 下的隐藏子文件 tgt_inst_Load 中，并运用于整个代码中，以定义在 Toolbar Button（工具栏按钮）中使用的重要变量。

logo_new_guid（字符串）如果 logo_install_tools = true，那么这里需要 Windows 特殊的 GUID 用于你的 Tools Menu（工具菜单）扩展。包括由 guidgen.exe 生成的前括号和后括号。

例如：logo_new_guid = "{1E796980-9CC5-11D1-A83F-00C04FC99D61}"

切记要用 GUIDGEN 生成新的 GUIDs，以用于你自己的代码修改！

logo_new_guid_2（字符串）**必需的**。当你在 FirstButton 安装信息的基础上建立一个 Toolbar Button 的时候，此变量应当被设置成为 Windows 的一个特殊的 GUID。这个 GUID 值必须是独一无二的。请不要重复使用与 logo_new_guid 中相同的 GUID。

切记要用 GUIDGEN 生成新的 GUIDs，以用于你自己的代码修改！

logo_brand_name（字符串）出现在 Info Tip（信息提示）上的品牌名称，用于 Toolbar Button。

logo_company_name（字符串）你公司的名称。例如：logo_company_name = "Example, LLC"

logo_company_url（字符串）你公司网站的链接，点击 Toolbar Button 即可以引导浏览器打开此网站。
例如：www.exampleco.com

logo_default_target_pos（整数）对于 IE6：位于工具栏上(相对的) Toolbar Button 应该出现的位置。

设置此变量为 0，你的 Toolbar Button 则会位于用户IE工具栏上的第一个当前出现的按钮的左边。

设置它为 1，Toolbar Button 则会位于工具栏上的第一个当前出现的按钮的右边。

logo_IE7_default_target_pos (整数) 对于 IE7、IE8、IE9、IE10 和 IE11: 位于工具栏上(相对的) Toolbar Button 应该出现的位置。

设置此变量为 0, 你的 Toolbar Button 则会位于用户IE工具栏上的第一个当前出现的按钮的左边。
设置它为 1, Toolbar Button 则会位于工具栏上的第一个当前出现的按钮的右边。

logo_IE7_use_mapid_logic (布尔值) 对于 IE7、IE8、IE9、IE10 和 IE11: 这时一个非微软按钮出现在默认目标位置之前。

若设置此变量为“true”的话, 那么 FirstButton 将会安装你的按钮1的位置至那个按钮的左边, 而不是默认的目标位置。因此它会出现任何其他按钮的前面。

logo_use_mapid_logic (布尔值) 对于 IE6: 这时一个非微软按钮船舷在默认目标位置之前。若设置此变量为“true”的话, 那么 FirstButton 将会安装你的按钮1的位置至那个按钮的左边, 而不是默认的目标位置。因此它会出现任何其他按钮的前面。

logo_install_favorite (布尔值) 当设置其为“true”的时候, FirstButton 将为你的网站添加一个 Favorites Link (收藏夹链接), 将.exe 安装文件用作一个图形源。使用来自 logo_company_url 中的 URL (链接) 信息。

logo_install_tools (布尔值) 若设置其为“true”, FirstButton 将为你的网站添加一个 Tools Menu (工具菜单), 使用来自 logo_company_url 中的 URL 信息。



图标

在 FirstButton 源代码中, 有一些被添加到了注册表中的行 (如下所示), 用来指定IE浏览器正确的图标大小和图形。图标号的分配和含义的解释如下:

```
(logo_app_path & logo_install_module & ",1")
```

(Windows中包含一些数字来作为字符串的一部分, 这些数字可以指定IE图标的位置)。

以下所有 .ico 文件都包含多个图标, 大小分别为 16x16、20x20、24x24、32x32、48x48。

- 1 对于 IE7、IE8、IE9、IE10、IE11 (桌面模式), 此变量被用作 HotIcon 和 Icon 源。
对于 Windows XP系统下的 IE6, 此变量被用作 Toolbar Button的图标值(256 色)
(包含的图标大小有 16x16、20x20、24x24、32x32、48x48) 
 - 2 对于 Windows XP系统下的 IE6, 此变量被用作 Toolar Button 的 HotIcon 值(256 色, 但为灰色调)
(包含的图标大小有 16x16、20x20、24x24、32x32、48x48) 
- 对于 Windows2000 系统下的 IE5.5或6, 此变量被用作图标 (256 色, 适用于 IE6)
- (包含的图标大小有 16x16、20x20、24x24、32x32、48x48) 
- 4 对于 Windows 2000 系统下的 IE 5.5 或 6, 此变量被用作 HotIcon [鼠标位于Toolbar Button上]
(256 色, 但为灰色调)
(包含的图标大小有 16x16、20x20、24x24、32x32、48x48) 
 - 5 对于 Windows 98、Me、或 NT, 此变量被用作 Toolbar Button 的图标变量 (16 色)
(包含的大小有 16x16、20x20、24x24、32x32) 
 - 6 对于 Windows 98、Me、或 NT, 此变量被用作 HotIcon (鼠标位于Toolbar Button 上)
(16 色, 但为灰色调)
(包含的图标大小有 16x16、20x20、24x24、32x32、48x48) 

自定义你的Toolbar Button的FirstButton™源代码的操作过程

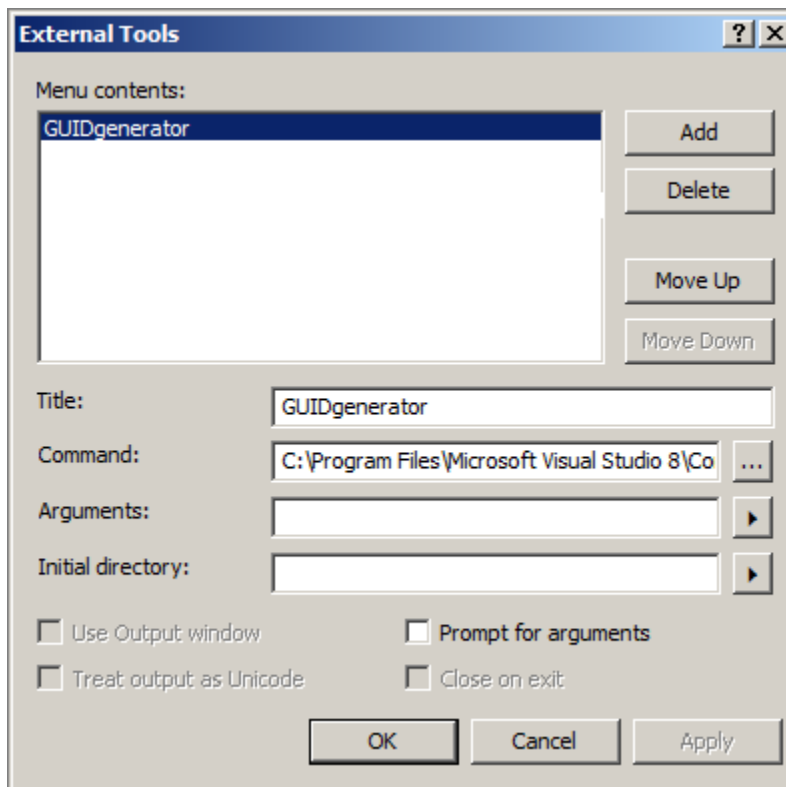
- 因为所有程序组件是在一个Windows文件夹中，所以要使用 Visual Studio 2005（或更高的版本）来打开 `loginst.sln`。
- 使用 GUIDGEN.exe 来生成 2 个特殊的 GUIDs {#####-####-####-####-#####} 使用 Registry Format 选项。一个 GUID 将用于工具栏按钮的 `tgt_inst.frm` 源代码中，而另一个 GUID 则用于工具菜单扩展。

当进行 VS 2005 安装时，GUIDGEN.exe 的默认路径为：`C:\Program Files\Microsoft Visual Studio 8\Common7\Tools\GUIDGEN.exe`

GUIDGEN的使用说明可以参见此链接：[http://msdn.microsoft.com/en-us/library/ms241442\(v=vs.80\).aspx](http://msdn.microsoft.com/en-us/library/ms241442(v=vs.80).aspx)

据我的经验，GUIDGEN 必须通过手动添加到工具菜单选项中：

选择 Tools Menu，然后选择 External Tools 打开 External Tools 对话框。单击 “Add”，并提供一个有意义的 Menu 选项标题，比如 “GUIDgenerator”，并指定 Command 路径为 `C:\Program Files\Microsoft Visual Studio 8\Common7\Tools\GUIDGEN.exe`，或到你电脑上的 GUIDGEN.exe 的位置。



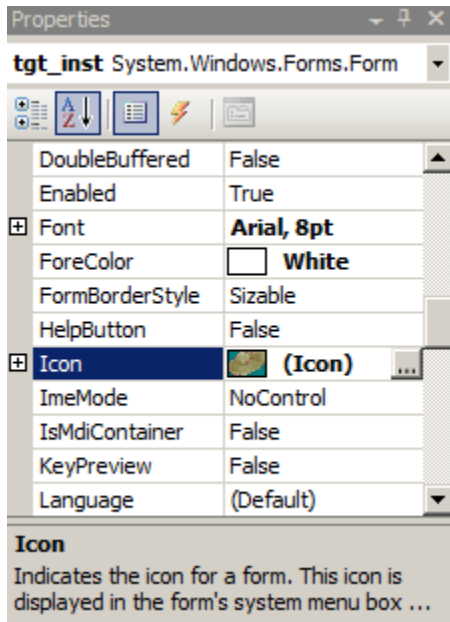
通过双击 Windows Explorer 中的 Utility，GUIDGEN.exe 也可运行在 Visual Studio 的外部。

- 将这些变量定位在 `tgt_inst_Load` 区域中，并使用那些刚才由 GUIDGEN 生成的 GUIDs 来替换已存在的 GUIDs：

```
logo_new_guid = "{DED51647-3FFD-48da-9B48-814A053F6388}" <- copy your 1st GUIDGEN value here
```

```
logo_new_guid_2 = "{D294CEA5-4017-4cb1-8B95-054C4A2FB2AE}" <- copy your 2nd GUIDGEN value here
```

- 使用你的个人品牌来替换 `tgt_inst.frm` 和 `tgt_license.frm` 的图标。这应该是一个 `.ico` 文件，此文件包含的图标大小有 16x16、20x20、24x24、32x32、48x48。Icons 区域的一号图标可用作 `tgt_inst.vb` 的图标定义。



- 为你的安装模块选择一个特殊的程序名称，比如像 `sample.exe`。以 “`sample.exe`” 更新 `logo_install_module` 变量，“`sample[1].exe`” 更新 `logo_install_module_2`，“`sample[2].exe`” 更新 `logo_install_module_3`，这里所说的 `sample` 是你的程序名称。`.exe` 扩展名在这种情况下是必须要加上的，因为所有被指定的图标都是自身包含在安装模块之中。

```

logo_company_name = "[your company name goes here]"
logo_brand_name = "[your company or brand name goes here]"
logo_company_url = "[your company URL or target URL for the toolbar button]"
logo_new_guid = "[GUID for your Tools menu extension]"
logo_new_guid_2 = "[GUID for your Toolbar Button]"
logo_install_module = "sombremos.exe"
logo_install_module2 = "sombremos[1].exe"
logo_install_module3 = "sombremos[2].exe"
logo_IE7_install_guid = "N"           (首字母大写设定, 应保留为 'N')
logo_IE7_install_guid_2 = "N"        (首字母大写设定, 应保留为 'N')
logo_install_guid = "N"              (首字母大写设定, 应保留为 'N')
logo_install_guid_2 = "N"            (首字母大写设定, 应保留为 'N')
logo_toolbar_cmdmap_exists = "N"     (首字母大写设定, 应保留为 'N')
logo_button_installed = False        (首字母大写设定, 应保留为 'False')
logo_default_toolbar_built = False   (首字母大写设定, 应保留为 'False')
logo_toolbar_prev_install = False     (首字母大写设定, 应保留为 'False')
logo_default_target_pos = 2           IE6将插入你的 Toolbar Button 的工具栏位置
logo_IE7_default_target_pos = 2       IE7-IE10 将插入你的 Toolbar Button 的工具栏位置
logo_IE7_use_mapid_logic = True        参见变量部分中的描述
logo_IE7_allow_multi_installs = False  参见变量部分中的描述
logo_use_mapid_logic = True            参见变量部分中的描述
logo_allow_multi_installs = False      保留为 'False' - IE 不能很好的处理多个安装程序
logo_install_tools = True              如需安装 Tools Menu Extensions, 设为 'True'
logo_install_favorite = True           如需安装 Favorites Menu Item, 设为 'True'
logo_install_add_remove = True         建议保留设置 'True'。否则, 你的 Toolbar Button 的卸载 Control Pane
                                        1选项卡, 用户将无法使用。

```

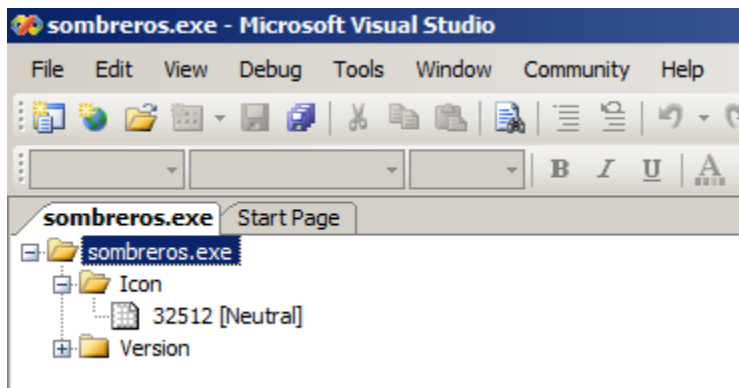
导入图标变量

1) 启动 Visual Studio

2) 从 File Menu 中选择 “Open File” 。

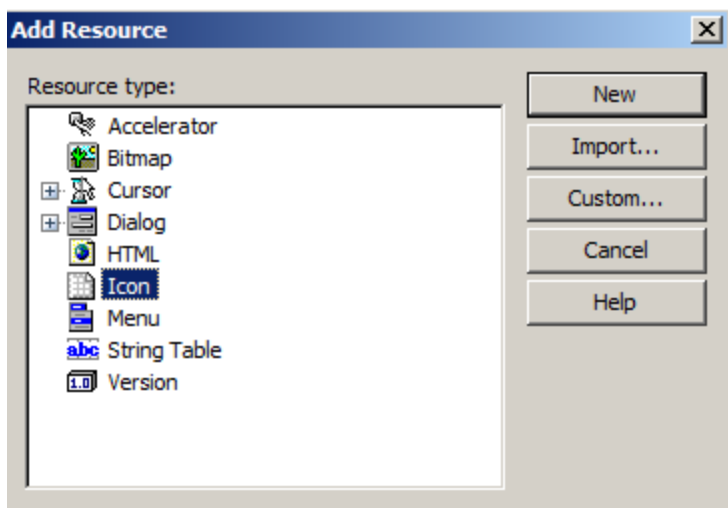
在 File Open 对话框中，选择你所编译的可执行文件，如 “sample.exe” 。

屏幕显示的示例中，我已概括为 “somebreros.exe”



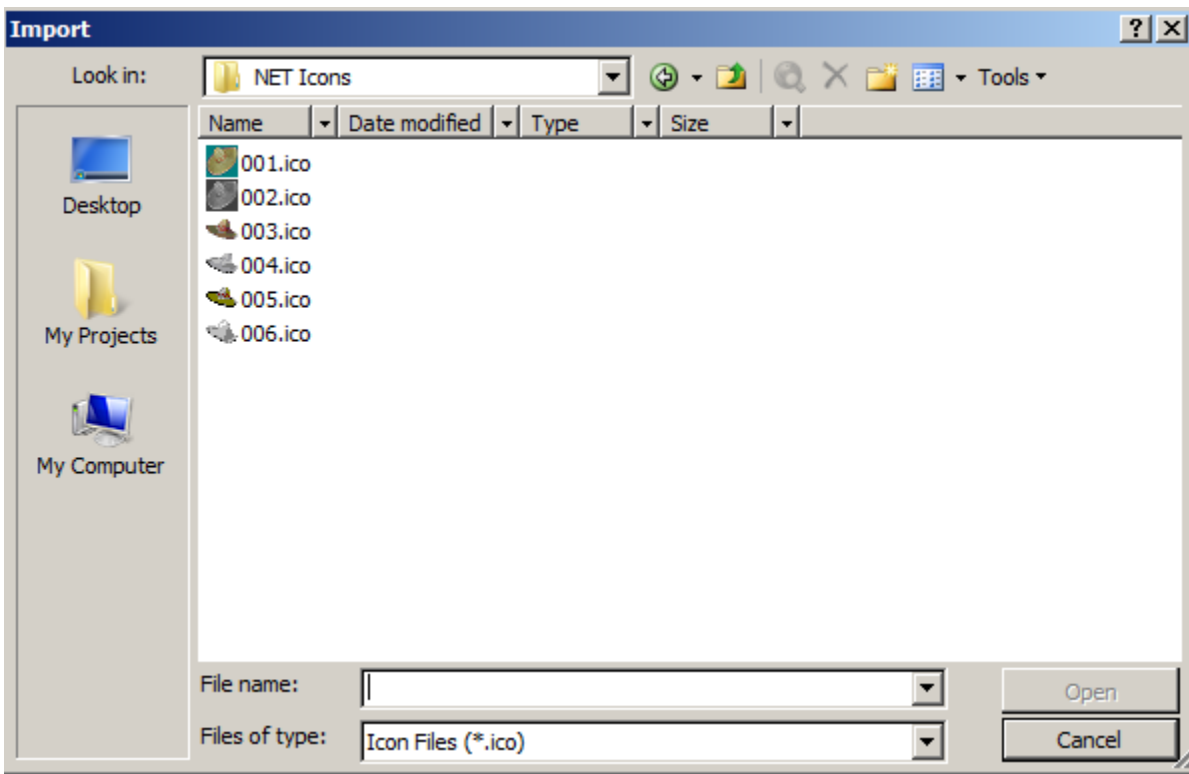
3) sombreros.exe 被突出显示，从 Edit Menu 中选择 “Add Resource” 。

从 Resource type 变量中，选择 Icon，并单击 Import。

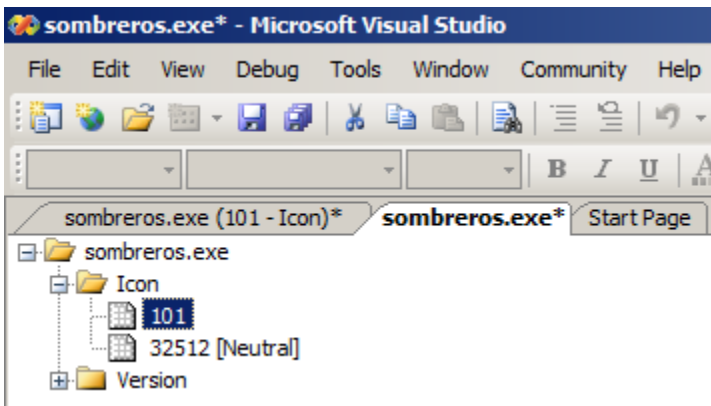


将出现一个文件打开对话框。将对话框引到你的自定义图标的文件夹位置。

输入 *.ico 作为文件名，并按 enter 键，只能显示图标文件。

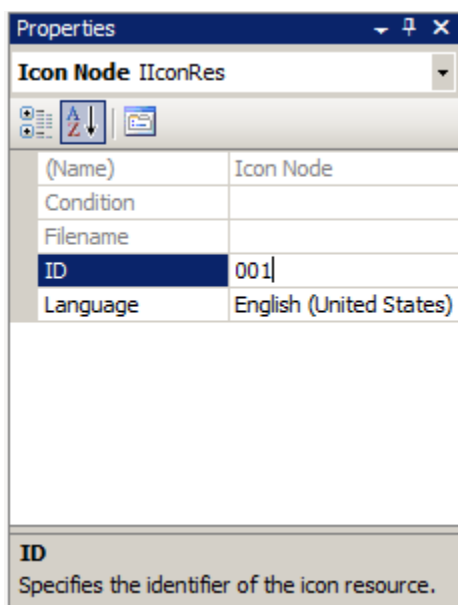
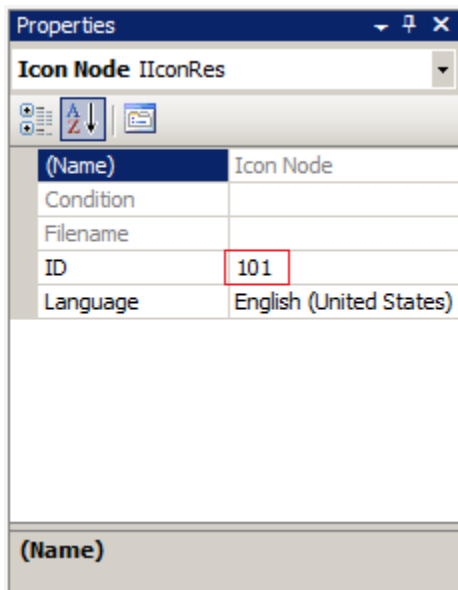


选择第一个图标，我们示例中的“001.ico”，并单击“Open”导入文件到可执行文件中。导入图标之后，它会显示在 Visual Studio 中。接下来，单击“sombros.exe”选项卡：

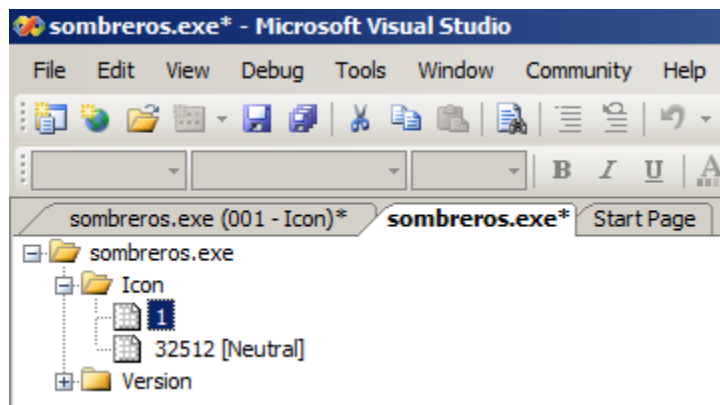


文件图标号“101”应与我们刚才导入的图标文件相关联。

“101”突出显示,单击右键显示菜单，并选择“properties”。在属性窗口中，ID 为当前的设置“101”。我们需要将其更改为“001”。



在可执行文件属性树形视图中，它会有“1”的标识符，但它内部 ID 号仍然是“001”，如属性窗口中所列。



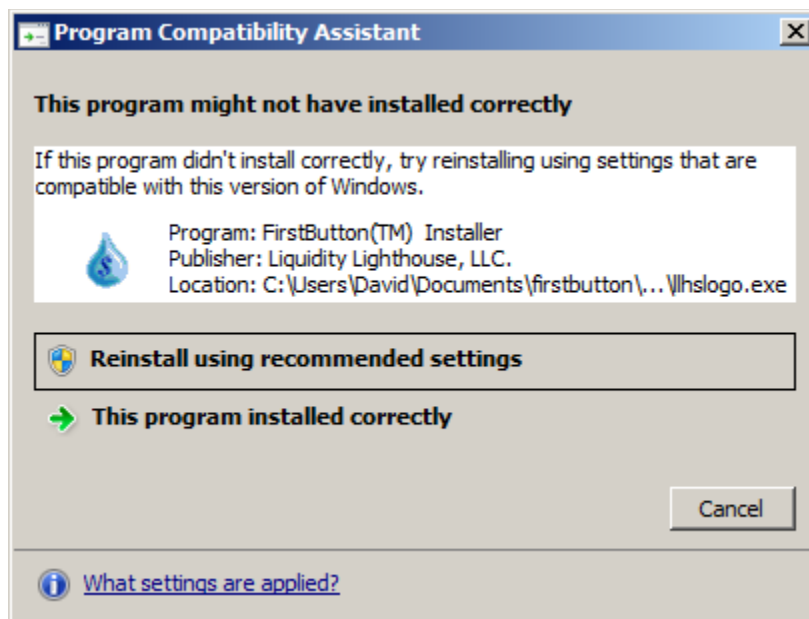
使用 CTRL+S 或右键单击“sombreros.exe”选项卡，并使用下拉菜单来保存“sombreros.exe”。

重复图标导入并重新命名进程，来对另外5个图标与 Variables 区域中的描述进行匹配，

- 1 对于 IE7、IE8、IE9、IE10 和 IE11，这个值被用作 HotIcon 和 Icon 源
 对于 Windows XP 系统下的 IE6，这个值被用作 Toolbar Button 的 Icon 值。 (ID = 001)
- 2 对于 Windows XP 系统下的 IE6，它被用作 Toolbar Button 的 HotIcon 值 (ID = 002)
- 3 对于 Windows 2000 系统下的 IE 5.5 或 6，它被用作 Icon (ID = 003)
- 4 对于 Windows 2000 系统下的 IE 5.5 或 6，它被用作 HotIcon (鼠标位于 Toolbar Button 之中) (ID = 004)
- 5 对于 Windows 98、Me 和 NT，它被用作 Toolbar Button 的 Icon 值 (ID = 005)
- 6 对于 Windows 98、Me、或 NT，它被用作 HotIcon (鼠标位于 Toolbar Button 之中) (ID = 006)

一旦用 Visual Studio 保存了 .exe，该程序即可被测试。

如果解决方案或项目的编译没有兼容性部分的话，Windows Vista 或 Windows 7 中可能会出现以下警告信息：



兼容性部分位于 My Project 文件夹中的 app.manifest 文件的中间，但是这个文件夹中的 VS 2005 版本并未包含它：

```
<compatibility xmlns="urn:schemas-microsoft-com:compatibility.v1">
  <application>
    <!-- A list of all Windows versions that this application is designed to work with. Windows will automatic
ally select the most compatible environment. -->
    <!-- The ID below indicates application support for Windows 8.1 -->
    <supportedOS Id="{1f676c76-80e1-4239-95bb-83d0f6d0da78}" />
    <!-- The ID below indicates application support for Windows Vista -->
    <supportedOS Id="{e2011457-1546-43c5-a5fe-008deee3d3f0}" />
    <!-- The ID below indicates application support for Windows 7 -->
    <supportedOS Id="{35138b9a-5d96-4fbd-8e2d-a2440225f93a}" />
```



```
<!-- The ID below indicates application support for Windows 8 -->
<supportedOS Id="{4a2f28e3-53b9-4441-ba9c-d69d4a4a6e38}" />
</application>
</compatibility>
```

请发送问题或与此文档相关信息至 documents@liquiditylighthouse.com.

无担保责任。Liquidity Lighthouse, LLC 公司明确否认就任何形式的软件产品负有保责任。软件产品和任何相关文档均是“按原样”发布，不承担任何形式的担保，无论是明确或隐含担保，包括但不限于隐含的适销性担保、商品具体适销性担保、或非侵权产品的担保。

责任范围

根据可适用的法律所允许的最大范围，在任何情况下，Liquidity Lighthouse, LLC 公司或其公司的供应商均不就任何形式的特殊、连带、间接、或结果性的损失承担赔偿责任，无论是由于使用或是非法使用**软件产品**、还是由于**支持服务**条款或是提供**支持服务**失败所导致的任何形式的损失（包括但不限于营业利润损失、业务中断、商业信息丢失、或任何其他形式金钱损失）。

使用此**安装软件之时**，你即同意不宣称任何知识产权，包括但不限于侵犯 Lighthouse, LLC 公司及其供应商现在或将来的商标权的申明。

Liquidity Lighthouse, LLC
PO Box 211091
Bedford, TX 76095



FirstButton 和 Liquidity Lighthouse 均为 Liquidity Lighthouse, LLC 公司的商标。