

# 第 1 章

## 快速入门

### 本章内容

---

- 安装并开始使用 Visual Studio 2015
- 创建并运行第一个应用程序
- 调试并部署应用程序

自从开始开发软件以来，就需要使用工具来帮助我们编写、编译、调试和部署应用程序。Visual Studio 2015 是最佳组合的集成开发环境(Integrated Development Environment, IDE)继续演化的下一个版本。

本章介绍 Visual Studio 2015 的用户体验，并学习使用各种菜单、工具栏和窗口。作为 IDE 的快速入门，本章不详细列举每一个可以更改的设置，也不会介绍如何自定义 IDE 的布局，这些主题会在后续章节讨论。

### 1.1 入门

每次发布 Visual Studio 产品时，安装过程都会逐渐改进，现在安装和运行 Visual Studio 2015 非常简单。本节介绍安装过程，并开始使用 IDE。

#### 1.1.1 安装 Visual Studio 2015

启动 Visual Studio 2015 安装程序时，会显示如图 1-1 所示的对话框，从中可以指定安装的位置。点击底部的链接，阅读许可信息之后(当然要详细阅读)，可以选中复选框，以接受条款，并移到安装过程中的下一个屏幕。

Visual Studio 2015 自身附带许多强制性功能。因为这些功能是内置的，所以安装过程中并没有专门提及它们。然而，可以从 3 个不同的组中选择许多可选功能。第一组如图 1-2 的左边，允许选择额外的编程语言。当然，选择 C#和 VB.NET 不需要额外的步骤。也可以在这里添加 C++、F#和一些使用 Python 的工具。第二组如图 1-2 的中间所示，包括与 Windows 和 Web 开发项目相关的工

具。选择与自己的开发项目有关的功能。第三组如图 1-2 的右边所示，允许安装与跨平台开发任务相关的工具，包括 Xamarin 和 Android 工具。选择确实需要的功能(注意，总是可以在后面添加或移除这些功能)，然后单击 Next。这会显示一个安装预览对话框(如图 1-3 所示)，其中描述了要安装的内容，并允许查看可能需要的任何特殊的许可条款。然后单击 Install 按钮，开始安装过程。

此时会显示进度对话框，如图 1-4 所示。根据已安装到计算机上的组件，安装过程中或结束时可能会提示用户重启计算机。成功安装好所有的组件后，会显示 Setup Completed 对话框，如图 1-5 所示，其中表明安装过程中没有出现任何问题。



图 1-1

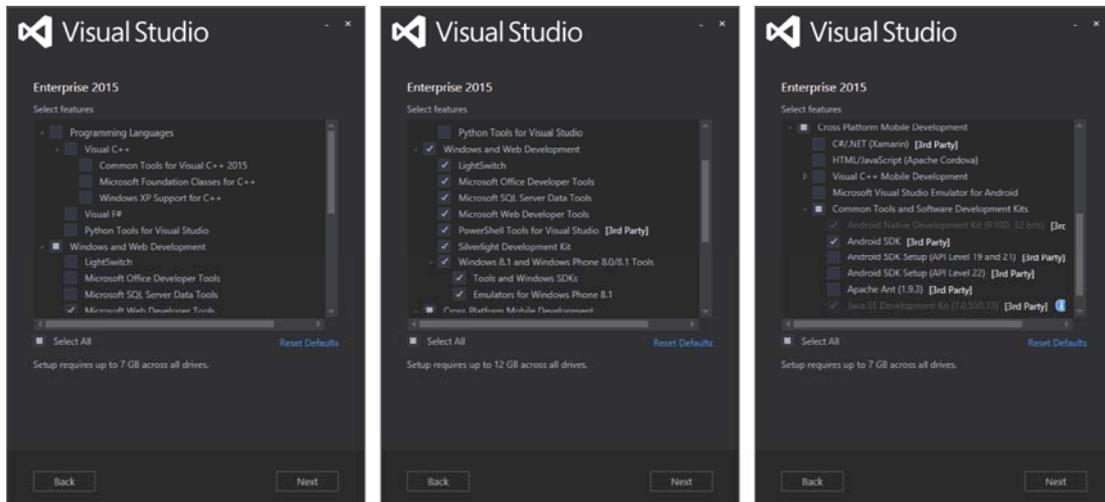


图 1-2

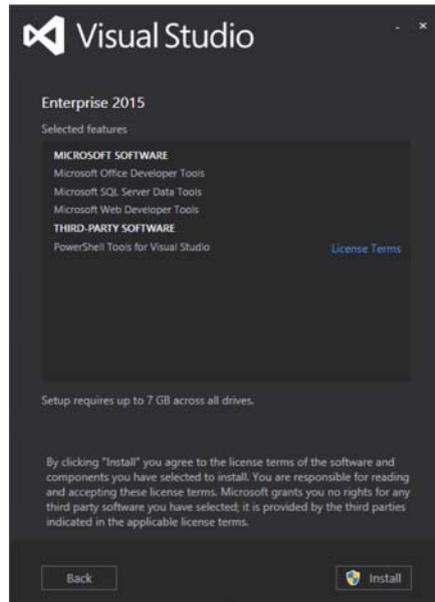


图 1-3

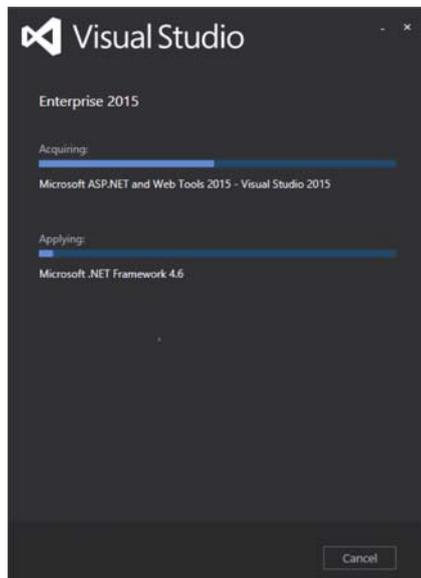


图 1-4



图 1-5

### 1.1.2 运行 Visual Studio 2015

第一次运行 Visual Studio 2015 时，就有机会登录，如图 1-6 所示。

这种行为是 Visual Studio 支持云的努力的一部分——把 Visual Studio 设置和功能连接到互联网上可用的资产上。这不需要登录。如图 1-6 所示，Sign In 按钮的下面是 Not Now, Maybe Later 链接。

点击该链接，跳过一些步骤(仍然需要选择默认的环境设置)，以很快进入 Visual Studio。但登录也有一些优点。

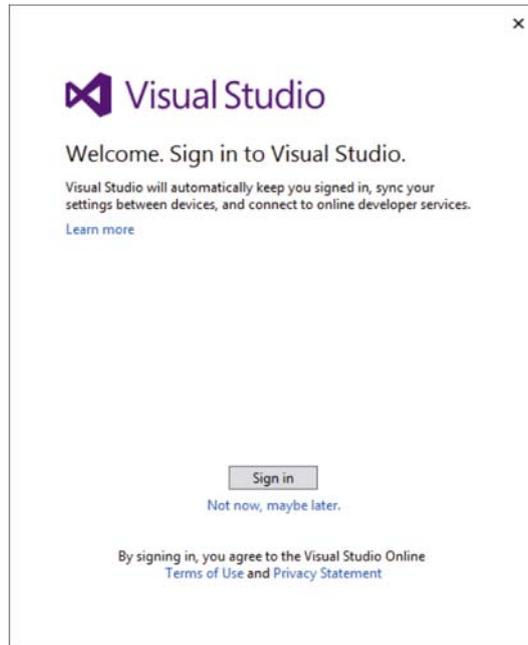


图 1-6

### 1.1.3 Visual Studio 真的支持云吗

简洁的回答是“支持”。更准确的回答是“支持，如果需要的话”。在创建这个功能时，研究工作的一部分涉及 Microsoft 要理解开发人员如何识别各种在线功能。一般来说，大多数开发人员都有两个或多个在开发时使用的 Microsoft 账户。他们有一个主要的身份，一般映射到工作时使用的凭据。他们还有其他身份，用于访问外部功能，比如 Team Foundation Server，或者把应用程序发布到不同的 Microsoft 存储。

为了模仿开发人员如何使用多个在线身份，Microsoft 在 Visual Studio 中给这些身份引入了一个层次关系。登录时，指定的账户是用于 Visual Studio IDE 的主要身份。从理论上来说，它应该代表开发人员。用同一个凭据登录到 Visual Studio 的任何地方，首选设置都不变。这包括主题和键盘绑定等自定义设置。对一个设备的改变会自动反映到已登录的其他设备。

为了处理二级凭据，Visual Studio 2015 包含了一个安全凭据库。这允许记录并使用到外部服务的连接，而不必每次都提供身份验证。当然，可以从特定的连接中手动注销，并删除凭据。

为了支持在机器之间漫游的设置，需要登录，单击如图 1-6 所示的 Sign In 按钮，就会启动这一过程。几分钟后，屏幕会提示输入 Microsoft 账户的凭据。

登录后，Visual Studio 就会显示欢迎屏幕。

作为云支持的一部分，用户名会显示在 IDE 的右上角(假设已登录)。如果单击下拉箭头(如图 1-7 所示)，就会看到 Account Settings 链接。点击该链接，会打开一个对话框(如图 1-8 所示)，在这里可以管理账户的细节，包括将 Visual Studio 与不同的账户关联起来。

除了提供一个机制来编辑配置文件的基本联系信息外，该对话框还包含一个与该账户关联的 Team Foundation Service (TFS)账户列表。目前，配置页面似乎有点空洞无物。然而，当引入额外功能时，这里的信息量会随着时间的推移而增加。

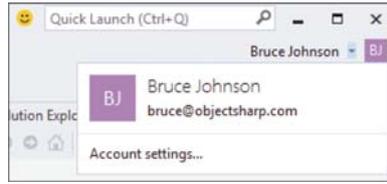


图 1-7

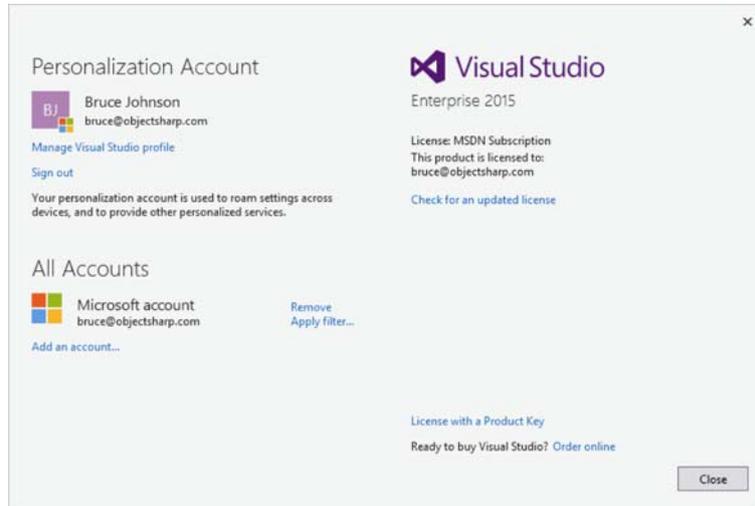


图 1-8

## 1.2 Visual Studio IDE

根据所选的环境设置，单击 Start Visual Studio 按钮时，会显示一个对话框，指示 Visual Studio 正在配置开发环境。当该过程完成时，将打开 Visual Studio 2015，此时就可以开始工作了，如图 1-9 所示。

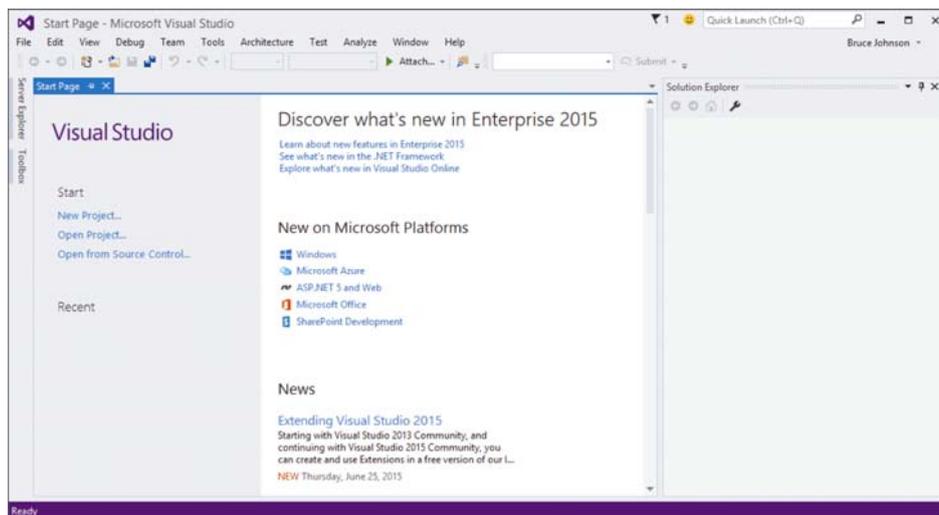


图 1-9

无论选择什么环境设置，在屏幕的中心都会显示 **Start Page**。该页面包含的链接可以用于查找各种 Microsoft 技术的新内容，讨论 Visual Studio 或其他产品中的不同特性的视频集合(在图 1-9 中不可见，但向下滚动，可以看见它们)，以及与 Visual Studio 相关的最新公告列表。

在开始生成第一个应用程序之前，应先回过头来看看组成 Visual Studio 2015 IDE 的组件。菜单和工具栏位于 IDE 的顶部，一系列子窗口或窗格显示在主窗口区域的左边、右边和底部。在其中心是主编辑区域：只要打开代码文件、XML 文档、窗体或其他文件，它们都会显示在这个区域中以供编辑。每打开一个文件都会创建一个新的选项卡，以便在这些打开的文件之间切换。

在编辑区域的两边是一组工具窗口：这些区域提供了额外的上下文信息和功能。对于一般的开发人员设置，默认的布局包括：右边有 Solution Explorer 和 Class View，左边有 Server Explorer 和 Toolbox。左边的工具窗口处于折叠(或取消固定)状态。如果单击某个工具窗口的标题，该窗口就会展开，当它不再是焦点或把光标移到屏幕的另一个区域时，该窗口会再次折叠起来。工具窗口展开时，在其右上角会显示 3 个图标，如图 1-10 的右上角所示。

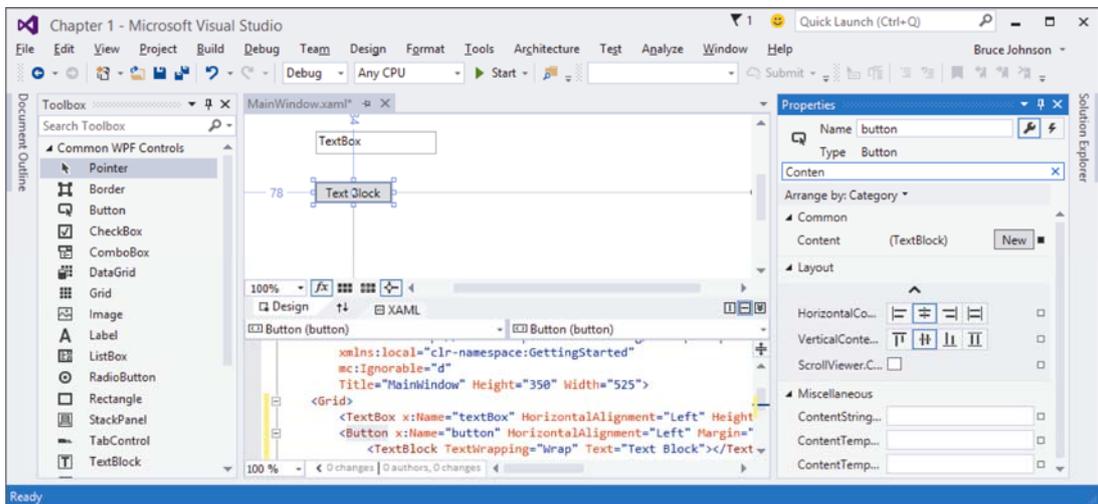


图 1-10

如果希望工具窗口保持展开(或固定)状态，可以单击中间的图标，它看起来像一个图钉。当这个图钉旋转 90° 时，表示该窗口现在被固定了。单击第 3 个图标“×”，就会关闭窗口。如果以后再要再次打开这个窗口或另一个工具窗口，可以从 View 菜单中选择。



一些工具窗口不能通过 View 菜单来访问，例如与调试相关的窗口，如线程和观察窗口。在大多数情况下，这些窗口可以通过另一个菜单项来访问。对于调试窗口而言，就是 Debug 菜单。

单击第一个图标(向下箭头)时，会显示一个上下文菜单。这个列表中的每一项都表示工具窗口的一种不同的排列方式。Float 选项可以把工具窗口放在屏幕的任意位置，独立于主 IDE 窗口。如果有多个屏幕，Float 选项比较有效，因为可以把各个工具窗口移到附加屏幕上，让编辑区域使用最大的屏幕空间。选择 Dock as Tabbed Document 选项会把工具窗口变成编辑区域的一个附加选项卡。第

4 章将介绍如何通过停靠和固定工具窗口来高效地管理工作区域。

## 开发、生成、调试和部署第一个应用程序

概览了 Visual Studio 2015 IDE 之后，本节介绍如何逐步创建一个简单的应用程序来演示如何使用其中一些组件。当然，这是每个开发人员都必须掌握的 Hello World 示例，根据用户的习惯，可以用 Visual Basic .NET 或 C#来完成。

(1) 首先选择 File | New | Project 命令，打开 New Project 对话框，如图 1-11 所示。对话框的左边有一个树状结构，用于根据语言和技术分组模板。在右上角还有一个搜索框。这个对话框的右窗格显示了所选项目模板的其他信息。最后，通过对话框顶部的下拉列表，可以选择应用程序所面向的 .NET Framework 版本。

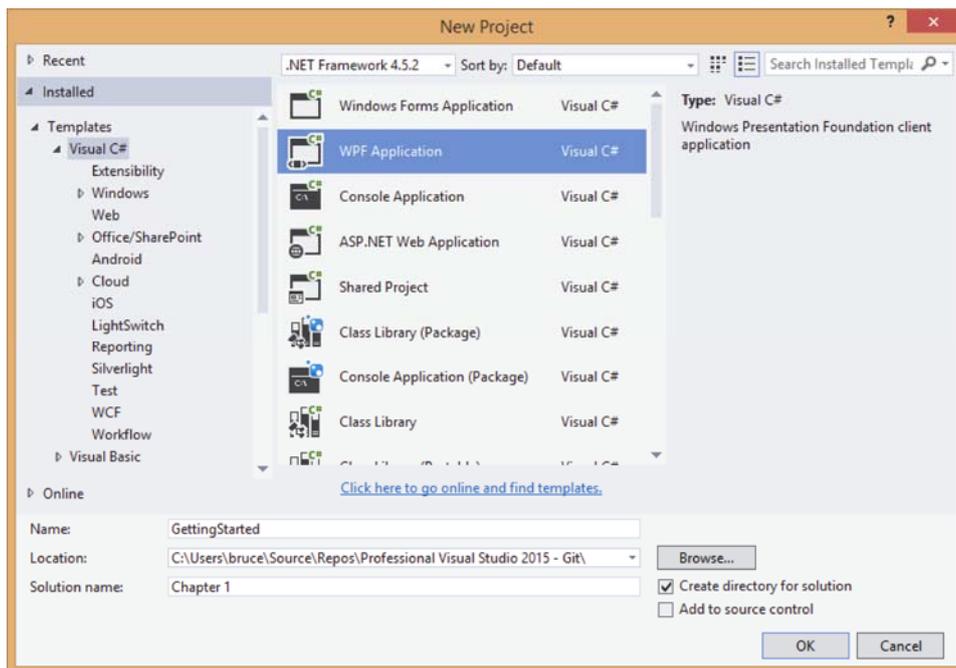


图 1-11

从 Templates 区域选择 WPF Application(这一项在根节点 Visual Basic 和 Visual C#下，或在子节点 Windows 下)，把 Name 设置为 GettingStarted，之后单击 OK 按钮。这将创建一个新的 WPF 应用程序项目，它包括一个开始窗口并包含在解决方案 Chapter 1 中，如图 1-12 的 Solution Explorer 窗口所示。这个开始窗口自动在可视化设计器中打开，给出了运行应用程序时窗口的图形化外观。Properties 工具窗口会折叠，并位于右边的工具窗口区域。

(2) 单击折叠的 Toolbox 窗口，其显示在屏幕的左边。这会展开 Toolbox 窗口。然后单击图钉图标，固定该工具窗口。要在 GettingStarted 项目的窗口中添加控件，可以从 Toolbox 中选择相应的项并拖放到窗体上。还可以双击该项，Visual Studio 会自动把它们添加到窗体上。

(3) 在窗体上添加一个按钮和一个文本框，布局应如图 1-13 所示。选择文本框，再选择 Properties 工具窗口(按下 F4 键会自动打开 Properties 工具窗口)。把该控件的名称设置为 txtSayHello(显示在 Properties 工具窗口的顶部)。对按钮控件重复这个操作，把它命名为 btnSayHello，将其 Content 属性

设置为“Say Hello!”。

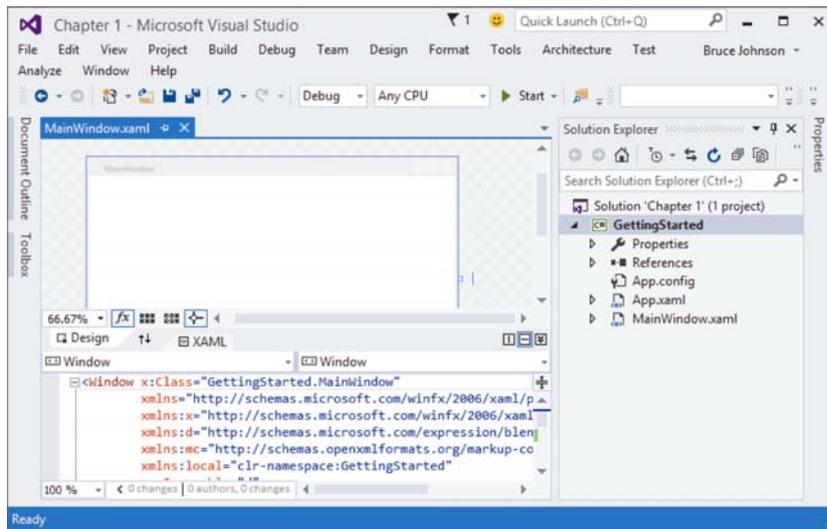


图 1-12

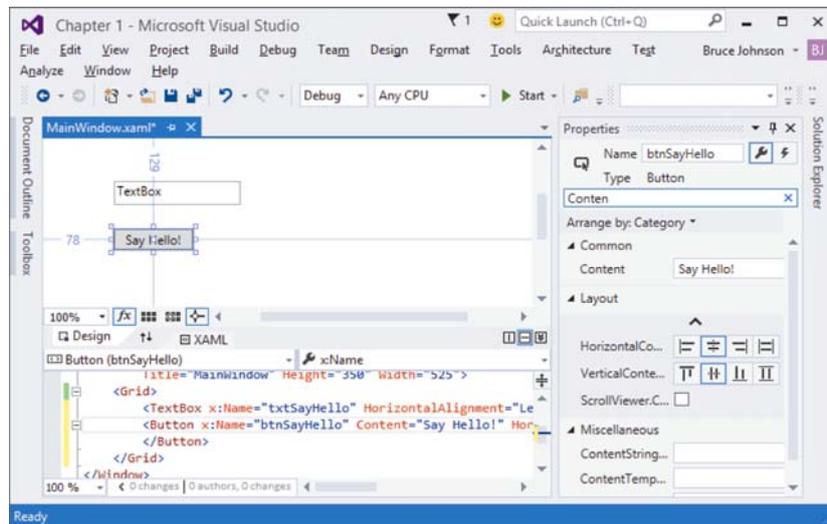


图 1-13

在 Name 字段下面的搜索字段中输入一个属性名，就可以快速定位该属性。在图 1-13 中输入 Conten，以缩短 Properties 列表，更容易找到 Content 属性。

在窗口上添加控件后，选项卡的文本后面就会加上星号(\*)，表示这个选项卡有未保存的修改。如果试图在修改内容处于挂起状态时关闭这个选项卡，Visual Studio 就会询问是否要保存这些修改。生成应用程序时，任何未保存的文件都会自动保存为生成过程的一部分。



需要注意的是，在 Visual Studio 2015 中进行修改时，一些文件也会改变，如解决方案文件，但不显示任何已改变的指示。如果要退出应用程序或关闭解决方案，Visual Studio 仍会提示保存这些修改。

(4) 取消对所有控件的选择(单击屏幕上的空白区域即可), 再双击按钮。这不仅会在代码编辑器中打开这个窗体的隐藏代码文件, 还会给按钮创建 Click 事件的处理程序。添加一行代码, 给用户回应一条消息, 代码窗口如图 1-14 所示。

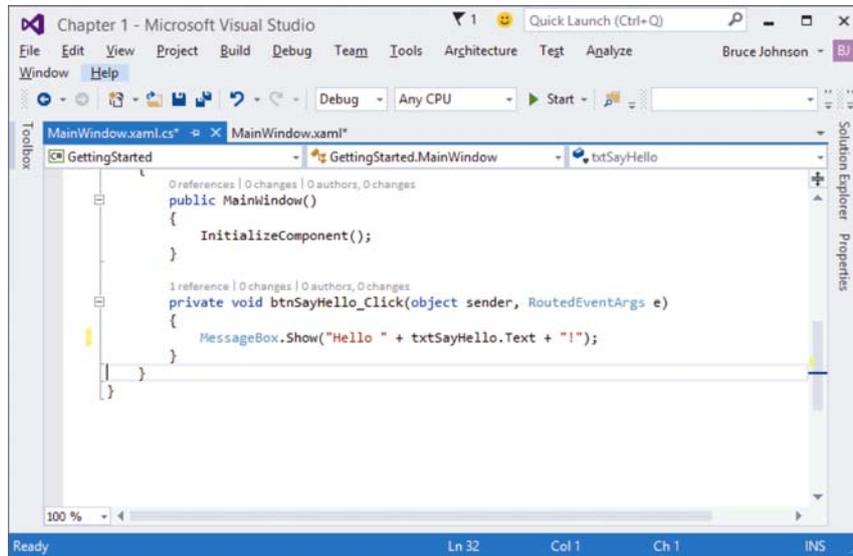


图 1-14

(5) 在生成并执行应用程序之前, 把光标放在包含 MessageBox.Show 的代码行上, 按下 F9 键。这将设置一个断点——按下 F5 键运行应用程序, 然后单击“Say Hello!”按钮后, 会在这一行上暂停应用程序的执行。图 1-15 图示了到达这个断点。把鼠标指针悬停在这一断点行上, 就会出现一个数据提示, 显示 txtSayHello.Text 属性的内容。

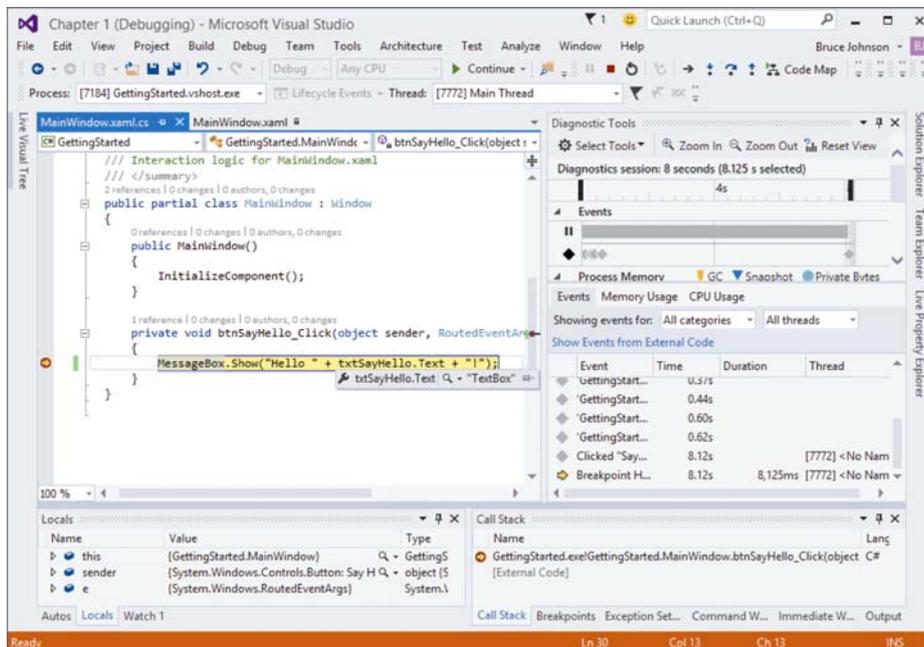


图 1-15

在图 1-15 中，Visual Studio 的布局与前面的屏幕截图完全不同，因为在这个屏幕的下半部分显示了许多新的工具窗口，在顶部显示了新的命令栏。另外，IDE 底部的状态栏是橙色的，而当处于设计模式时，显示为蓝色。当停止运行或调试应用程序时，Visual Studio 会返回以前的布局。Visual Studio 2015 维护着两个分开的布局：设计时布局和运行时布局。当编辑项目时，菜单、工具栏和各个窗口使用默认布局；而执行和调试项目时，它们都定义了一个不同的设置。可以修改这些布局，以适应自己的风格，Visual Studio 2015 会记住这些修改。

(6) 最后一步是部署应用程序。无论是使用 Windows Forms 或 WPF 生成富客户端应用程序还是生成 Web 应用程序，Visual Studio 2015 都可以发布该应用程序。在 Solution Explorer 中双击 Properties 节点，选择 Publish 节点，就会显示发布应用程序的选项，如图 1-16 所示。

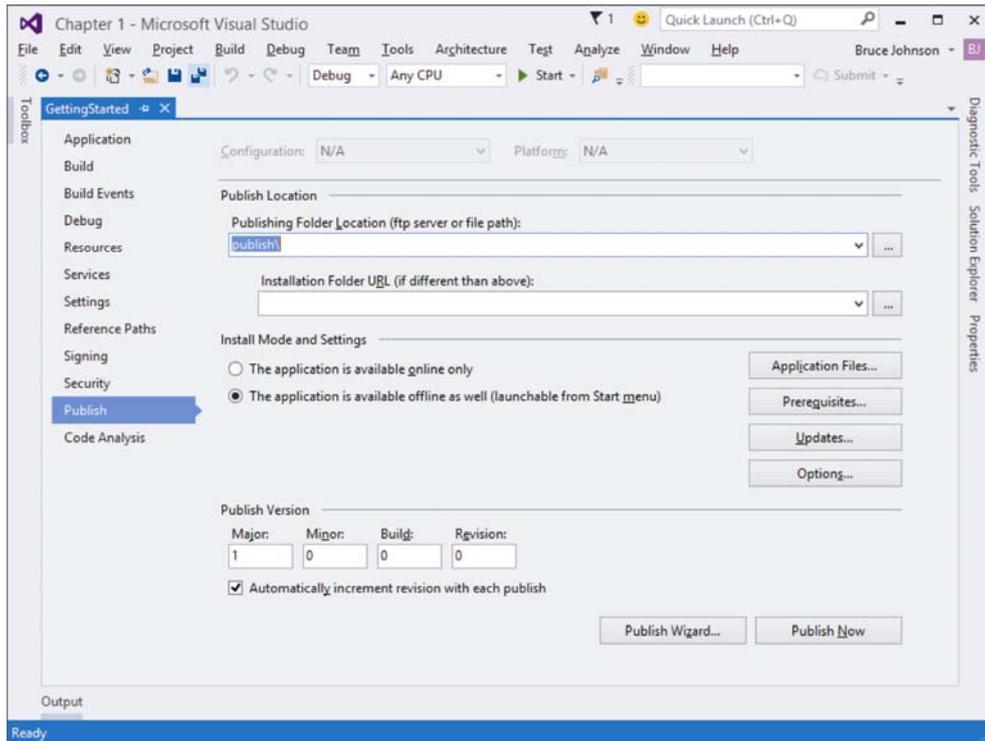


图 1-16

在图 1-16 中，发布文件夹被设置为本地路径(默认情况下，该路径相对于项目所在的目录)，但可以指定网络文件夹、IIS 文件夹或 FTP 站点。一旦指定要发布的位置，单击 Publish Now 按钮就会把应用程序发布到该位置。

### 1.3 小结

本章介绍了 Visual Studio 2015 的各个组件如何协调工作以生成应用程序。下面列出了创建解决方案的一般过程：

- (1) 用 File 菜单创建解决方案。
- (2) 用 Solution Explorer 定位需要编辑的窗口，双击该项，在主工作区域显示它。

- (3) 把需要的组件从 Toolbox 拖放到窗口上。
- (4) 依次选择窗口和各个组件，在 Properties 窗口中编辑属性。
- (5) 双击窗口或控件，访问组件图形化界面背后的代码。
- (6) 用主工作区域编写代码，并设计图形化界面，在该区域的顶部通过选项卡切换它们。
- (7) 用工具栏启动程序。
- (8) 如果出错，就在 Error List 和 Output 窗口中复查。
- (9) 用工具栏或菜单命令保存项目，并退出 Visual Studio 2015。

后续章节将介绍如何定制 IDE，使之更符合自己的工作风格，还将说明 Visual Studio 2015 完成了应用程序开发过程的大量工作。本书还会介绍作为开发人员使用 Visual Studio 2015 时可重用的许多最佳实践。