Chapter 1

Adding Silverlight to Your Web Development Toolkit

In This Chapter

▶ Understanding Rich Internet Applications (RIA)

▶ Seeing how Silverlight enables RIA development

▶ Understanding what you need to develop Silverlight applications

▶ Seeing what Silverlight can do beyond just creating a stylish user interface

*W*atch out, Adobe — Silverlight 4 is here to give you a run for your money. Web site designers and developers alike can find much to be excited about in Microsoft's answer to Adobe Flash. As we tell you in more detail in the Introduction to this book, Silverlight 4 is one of a new family of Web site applications that provide graphics, animation, audio, and video — all the features necessary for the kind of rich user experience people have come to expect on the Web. These applications are called Rich Internet Applications (RIA). Adobe Flash was one of the first in the RIA arena; more recently, Microsoft made its entry into this space with the launch of Silverlight.

In this chapter, we give you an overview of what Silverlight is, what it can do, and the tools you need to develop Silverlight applications.

Obtaining and Running Silverlight

As does Adobe Flash, Silverlight runs as a plug-in within the browser and needs to be installed the first time any Silverlight application is run. A plug-in is a piece of software that is not part of the Web browser but can be added to it to run additional applications. When a user visits a Web site that needs the Silverlight plug-in, the Web browser will prompt the user to install it.

After the plug-in is installed, visiting any site that hosts a Silverlight application causes Silverlight to start up in the browser seamlessly. The plug-in itself is around 4MB, and in addition to the Silverlight runtime (which is responsible for running the Silverlight application), the application itself needs to be downloaded from the site that hosts the application. Modern popular browsers such as Internet Explorer, Firefox, Safari, and Chrome support Silverlight while running on Microsoft's Windows operating systems 2003, XP, Vista, and 7) or on Apple's Mac OS X. Recently, Silverlight has also been rewritten to run on browsers in Linux-based operating systems. With Silverlight 4, the application can also run as a stand-alone application — that is, it can be run outside the browser.

Obtaining and installing Silverlight on your computer is quite simple. All you have to do is visit a Silverlight-enabled site and it will prompt you to install Silverlight. Figure 1-1 shows a site that uses Silverlight. As soon as you visit the site using a Web browser, the browser prompts you to install Silverlight.

Figure 1-1:
Site prompting Silverlight installation.

All you have to do is click the button to install and you are done. A Silverlight installation is around 4.7MB and needs to be installed only once. If you visit the same site again or another site that uses Silverlight, you are not prompted to reinstall it.

Checking Out Some Silverlight-Enhanced Sites

So, what do Silverlight applications look like? You can find a good example at the Hard Rock Café Memorabilia site (`http://memorabilia.hardrock.com/`), shown in Figure 1-2. This site contains photographs of some of the memorabilia that Hard Rock Café owns.

When the page comes up, the screen displays numerous small images of the memorabilia, but each time you click an image, you zoom in to it. As you zoom in to the image, it first appears blurred and progressively becomes sharper. This is because the whole photograph (which can be several megabytes in size) does not load in one go; instead, it loads progressively as you need it. You can also see several animations on the site. Creating an application like this using plain HTML would be very difficult and would lack the rich interactivity that is provided by Silverlight.
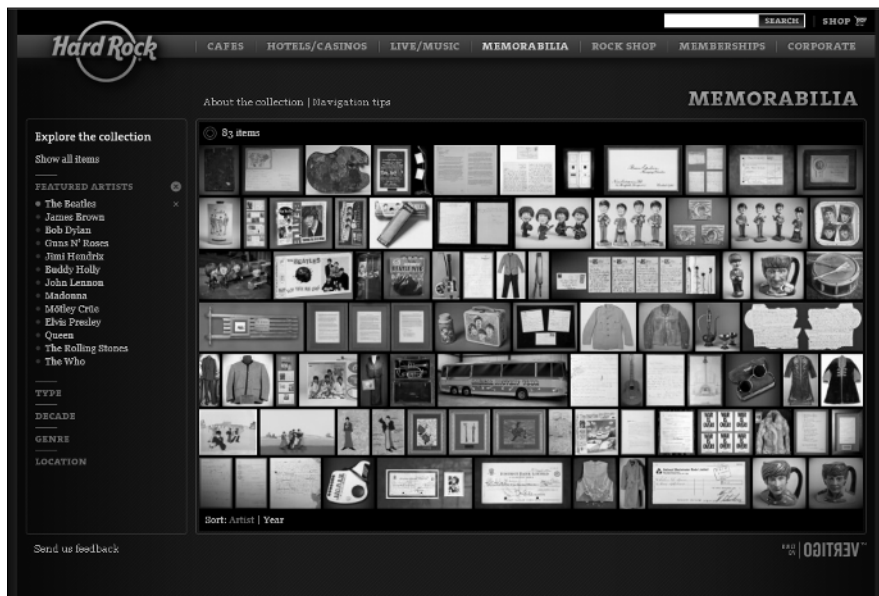

Figure 1-2:
Silverlight running the Hard Rock Café Memorabilia site.

Another site that showcases Silverlight's user interface is the Woodgrove Financials sample application, shown in Figure 1-3. This application uses fly-out menus, dynamic graphs, and slick animation.

See the list of images to the left of the screen? On the site, as you move your mouse cursor over them, a menu slides out, giving additional information about what you can do. Clicking the Mortgage option, for example, brings up an interactive graph that shows you how much money you can save on your mortgage. Clicking Trade History brings up an interactive display of graphs and tables, where you can filter your stocks, find information on them, and review the stock prices over a period of time.

The sample application has several features, which you can explore by visiting `http://cookingwithxaml.com/meals/financials/default.html`. The application is highly interactive. As noted previously, creating a rich user interface for an application such as this would be extremely difficult using plain ol' HTML.
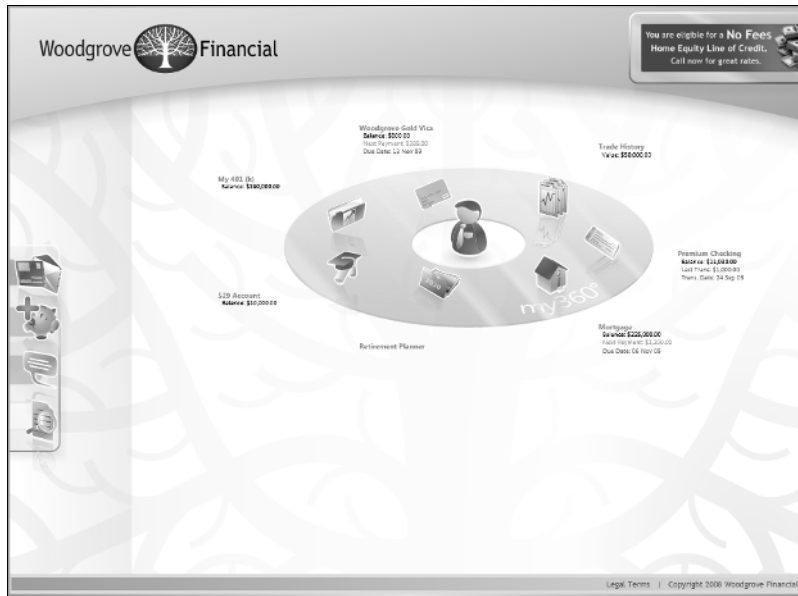
Figure 1-3:
An example of a dynamic site with a rich user interface.

Grasping the Potential of Silverlight

One of the main advantages of Silverlight is that if you are a developer and have already been creating applications using Microsoft technologies, you do not have to learn something new to create Silverlight applications. You can program Silverlight applications using .NET languages such as C# and VB.NET. In fact, the Silverlight runtime is a scaled-down version of .NET, which is Microsoft's primary environment under which most applications run. The .NET framework contains all the libraries needed to run these applications and provides the Application Programming Interface (API) that programmers can call to use the various features.

The Silverlight team worked hard to trim down the .NET framework so that it contains all the good bits of the framework while at the same time ensuring that it doesn't get too big to be sent across the Internet when you download it for the first time. The user interface framework, which is a key part of the RIA user experience, is based on Windows Presentation Foundation (WPF) and Extensible Application Markup Language (XAML). We tell you more about XAML in Chapter 2 and show you many examples of it throughout this book. Silverlight contains a powerful graphics and animation engine, and the UI framework provides the following features:

- ✔ Support for drawing vector graphic images and doing 3-D perspective transformations
- ✔ Support for creating storyboards and animations
- ✔ A full suite of controls, such as text boxes and buttons, and support for creating your own controls
- ✔ Layout management support, which allows you to lay out controls in a variety of ways
- ✔ Styling of controls, which allows the properties of your controls to be standardized and reused
- ✔ Skinning, which allows you to change the complete appearance of controls
- ✔ Support for *data binding,* which seamlessly binds data and control properties
- ✔ Multi-touch support, which allows you to use hand gestures and touch interactions on hardware that supports it
- ✔ Support for hooking on to webcams and microphones to share video and audio with others
- ✔ Support for media streaming, allowing audio and video to be streamed according to the optimal bit rate based on the network speed
- ✔ The ability to create Silverlight applications to be run on mobile phones that are part of the Windows Phone 7 series.

This set of features helps developers easily provide a rich interactive user experience.

Although the base class library in Silverlight is a cut-down version of the .NET framework, it still uses the same namespaces as the full version and supports a wide variety of features such as multithreading, generics, and even Language Integrated Query (LINQ). In addition, the framework adds support for WCF RIA Services, which helps build Silverlight applications in a multitier environment while allowing applications to work in offline scenarios at the same time. Other feature highlights of Silverlight include support for connecting to Web services, peer-to-peer communication, and access to local storage.

In spite of all these features, Silverlight itself runs in a sandboxed environment. This means that for security reasons, the Silverlight application cannot access system resources or invoke API calls on the machine it is running on. This feature prevents malicious applications from taking over your computer without your knowledge.

A brief history of the name "Silverlight"

Silverlight was initially called WPF/e (for "WPF everywhere"), which would have been a terrible name for it. It needed a catchy and simple name, and eventually the name Silverlight was chosen. Version 1 was released in September 2007, and it mainly accomplished the delivery of rich media and programming that couldn't be done using languages like C# and VB.NET.

Around the same time that version 1 was released, Microsoft also released a preview version of Silverlight. This 1.1 version of the program allowed developers to write Silverlight applica-tions using .NET languages such as C# and VB.NET. It also made use of the Framework API.

When version 1.1 was finally released in Oct 2007, it was rebranded as version 2.0. Silverlight 3 was announced in early 2009 and finally released in September 2009, soon followed by Silverlight 4 in the spring of 2010. In March 2010, Microsoft also announced that Silverlight will also be used to deploy applications for Windows Phone 7 series. Having four major releases in such a short time shows Microsoft's commit-ment to Silverlight and the RIA scene in general.

Hosting a Silverlight Application in a Web Page

Because Silverlight applications are run from a Web browser, they have to be hosted n a Web page. Silverlight applications have the extension `.xap` (pronounced "zap"), but the Silverlight file is actually nothing more than a `.zip` file. If you rename the file to have a `.zip` extension and open the file, you can still see all the compiled libraries, markup images, and other resources that the file contains.

To understand how a Silverlight application is hosted in HTML, you can navi-gate to a Web site that contains a Silverlight application and view the HTML source code by right-clicking and choosing View Source from the menu. The HTML source code will look something like the following:

```
<object data="data:application/x-silverlight-2,"
  type="application/x-silverlight-2" width="100%"
        height="100%">
    <param name="source" value="ClientBin/
        SilverlightApplication.xap"/>
    <param name="onerror" value="onSilverlightError" />
    <param name="background" value="white" />
    <param name="minRuntimeVersion" value="4.0. 50303.0"
        />
    <param name="autoUpgrade" value="true" />
    <a href="http://go.microsoft.com/
        fwlink/?LinkID=149156&v=4.0. 50303.0"
        style="text-decoration: none;">
```

```
       <img src="http://go.microsoft.com/
            fwlink/?LinkId=161376"
       alt="Get Microsoft Silverlight" style="border-style:
            none"/>
       </a>
</object>
```

The HTML tag is an `object` tag that specifies the data attribute and the type. These elements signify that a Silverlight object is being created. The `param name="source"` tag contains the relative location of the `.xap` file that will be run when the application starts.

Some of the other parameters contain values that specify where to get the latest versions of Silverlight.

If all this looks too complicated for you, don't despair. The `object` tags are created automatically when you use Visual Studio and Expression Blend.

All the Stuff You Need to Create Silverlight Applications

Microsoft provides all the programs needed to create Silverlight applications. Expression Blend and Visual Studio are the most commonly used applications to create Silverlight applications, and we cover these two products in depth in this book.

To start developing applications in Silverlight 4, you need to install the following programs and associated tools:

- ✔ Visual Studio 2010
- ✔ Silverlight 4 Tools for Visual Studio 2010
- ✔ Expression Blend for .NET 4
- ✔ Silverlight Toolkit
- ✔ WCF RIA Services
- ✔ Deep Zoom composer

Other than Visual Studio and Expression Blend, all the tools listed here are free for you to download and install. If you do not have Visual Studio 2010 installed on your machine, you can also use the free Visual Web Developer 2010 Express. There is, however, no free edition of Expression Blend.

Links to download the free tools can be obtained from `www.silverlight.net`.

Visual Studio

To write, compile, and debug programs, you need an Integrated Development Environment (IDE). Visual Studio is the IDE of choice for building applications that run in Microsoft's .NET environment. Visual Studio supports languages such as VB.NET and C#, and you can use it to create Silverlight applications, too.

Silverlight Tools for Visual Studio

Silverlight Tools for Visual Studio is responsible for adding all the necessary elements to get Visual Studio to build Silverlight applications. It adds the following: Silverlight project templates; the Silverlight Toolbox, which contains all the Silverlight controls such as buttons and text boxes; and the Silverlight Software Development Kit (the SDK). It also offers other useful features such as debugger support.

Expression Blend

You use Expression Blend to create user interfaces using visual tools for both WPF and Silverlight. It provides a much better environment for creating user interfaces than Visual Studio does. For instance, you can create a user interface just by dragging and dropping controls onto the design surface, whereas you would have to manually code most of this using Visual Studio.

Expression Blend is actually part of a suite of applications collectively known as Expression Studio. Although you can install just the applications you need, designers generally tend to install the entire suite of the Expression Studio applications because these tools help in visually designing the user interface. They complement each other quite well. The other applications in this suite are as follows:

✔ **Expression Web:** This application allows you to not only author Web pages in HTML but also design, build, and manage entire Web sites. It helps in creating Web sites that use Cascading Style Sheets (CSS), and it integrates with Microsoft's ASP.NET and PHP programming environments. Expression Web, shown in Figure 1-4, complies with Web

standards endorsed by the World Wide Web Consortium (W3C). It also contains something called Super Preview, which enables you to see how pages look in different browsers at design time. Traditionally, designers have struggled to make their applications look the same across multiple browsers, and Super Preview allows them to look at these pages side by side or even overlap each other to observe and fix differences.

✓ **Expression Design:** This application allows designers to create graphics and artwork for use in Web and desktop applications. You can even export these designs into XAML so that they can be used in Silverlight applications. Figure 1-5 shows Expression Design in action.

✓ **Expression Encoder:** This application helps in preparing media files (such as videos and audios) so that they can be used with Silverlight. A video file being edited in Expression Encoder is shown in Figure 1-6.
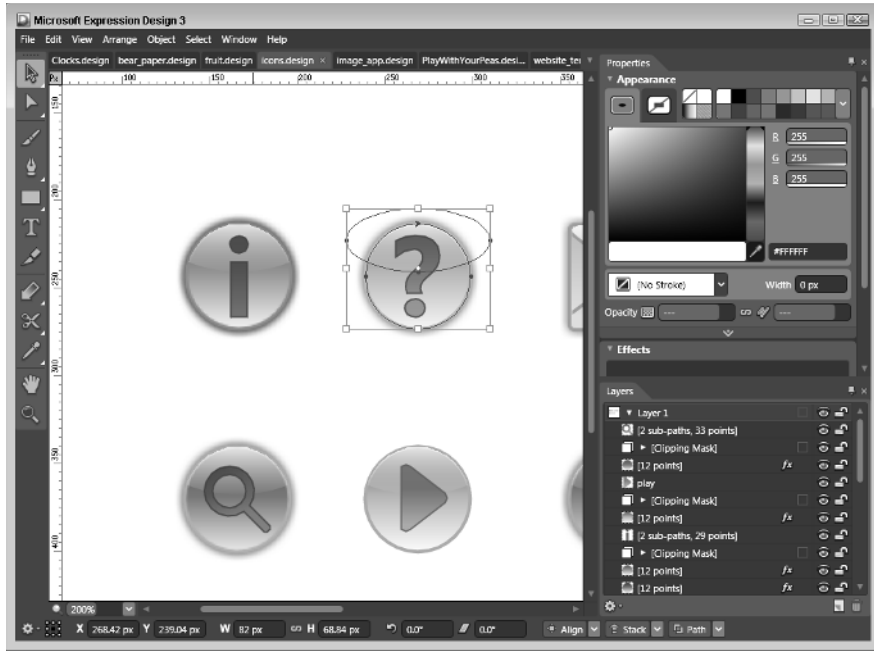

Figure 1-4: Expression Web in use.

Figure 1-5:
Graphics created using Expression Design.

Figure 1-6:
A video file being edited in Expression Encoder.

Silverlight Toolkit

The Silverlight Toolkit is a collection of controls and utilities for Silverlight that supplements what is already present in the core Silverlight libraries. The Silverlight Toolkit includes additional controls (such as charts) that are very useful but aren't used daily by applications. If Microsoft had included these controls in the main Silverlight runtime application, it would have become unwieldy and too big to download in a reasonable amount of time. By separating this set of controls from the main application, Microsoft left it up to developers to decide which parts to include in their application. In addition, it allows Microsoft to make frequent updates to the toolkit outside the release cycle of Silverlight.

WCF RIA Services

When you start creating large applications that have multiple screens and that connect to a database to read and write data, you need to design your application well. As part of this design, you need to divide your application into multiple layers, or *tiers*. An application is typically divided into the following tiers:

- ✔ **Presentation:** Focuses on the user interface
- ✔ **Application:** Focuses on application logic, such as validating user input or performing calculations
- ✔ **Data:** Focuses on how to read and write the data used by the application

WCF RIA Services is a framework for creating multitier applications in Silverlight, incorporating data operations such as authentication, authorization, data validation, and other essential services across the different tiers. It integrates with ASP.NET and provides ways to share code among the different tiers. WCF RIA Services is explained in more detail in Chapters 12 and 13.

Deep Zoom Composer

Deep Zoom Composer helps in creating the zooming and panning effect of images that you see in the Hard Rock Café Memorabilia site mentioned earlier in this chapter. Figure 1-7 shows Deep Zoom Composer being used to create a similar application. We tell you more about using Deep Zoom Composer in Chapter 15.

Figure 1-7:
Deep Zoom
composer in
action.

Creating Rich User Experiences in Silverlight

Rich Internet applications such as Silverlight aim to provide a dynamic experience for Web site users. This dynamism comes in the form of graphics, animation, audio, and video, and Silverlight supports all these features. In fact, Silverlight 1.0 was all about playing audio and video files in a browser, and nothing more.

So what kind of audio and video files work in Silverlight? In addition to MP3 files for audio, Silverlight supports a range of audio and video formats, including High Definition (HD)-quality video. Even the formats that it does not currently support can be easily converted into a supported format using Expression Encoder.

Graphics is another key area for providing rich user interfaces. Silverlight provides shape objects such as an ellipse, a line, a polyline, a rectangle, and a polygon that designers can modify to suit their needs. In addition, you can draw complex shapes using an object in XAML called `Path`. You can apply transformations on these vector graphic images to create 3-D effects and animations.

In addition to all these features, Silverlight, out of the box, contains around 60 controls. *Control* is another name for tools such as text boxes, buttons, and other elements that appear on Web sites. Part II of this book covers all the fundamentals you need to get started on working with the most commonly used controls. We even show you how to create your own!

Accessibility for people with special challenges has become a very important aspect of modern Web sites, and Silverlight supports this user accessibility by providing features that allow applications to be read using screen readers and by helping developers create functionality and features, within applications, that do not depend on the mouse alone but can be accessed using keyboard shortcuts. So, for example, if you have a menu in an application that usually pops up when you right-click the mouse, the user can also access it using a keyboard shortcut.

Silverlight also supports internationalization and localization, which allow applications to be written to support other languages and cultures. Not all countries speak or even use English, of course, and even the countries that do use English have varying date formats and currencies.

Silverlight Plays Well with Others

Silverlight does not actually replace HTML, ASP.NET, or JavaScript. In fact, it can complement these technologies to provide a pleasing, rich user experience for existing Web sites. The Document Object Model (DOM), which is a standard model for accessing all the user interface objects in an HTML page, can be manipulated from Silverlight, and some Silverlight functionality can be triggered from JavaScript. All these technologies actually work well together. You can find out how to integrate these technologies together in Chapter 15.

Silverlight Has More to Offer Than Just a Pretty Face

The user interface is the primary focus of Silverlight. It is also the most visible part as far as the users are concerned. But there is more to Silverlight than just creating pretty user interfaces.

In Part III of this book, we tell you how you can use Silverlight to access data that is available on the Web. This data can be in the form of Web services, which are small units of functionality that can be accessed by other applications. Such a unit of functionality can be anything from providing weather information for a certain region to complex business functionality such as managing product inventory or an employee database.

Silverlight does not have any API calls to talk to the database directly. It can use Web services or use another technology called WCF Data Services. Using WCF Data Services, Silverlight can perform all the necessary database operations such as creating, reading, updating, and deleting (known collectively as CRUD) data using calls to a Web service.

TECHNICAL STUFF

WCF Data Services is accessed by applications as RESTful services. REST stands for Representational Transfer State, and RESTful services present data from the database to applications as resources that can be accessed using unique Universal Resource Indicators (URIs), which are HTTP addresses. In addition, commands in the HTTP protocols (such as `GET`, `POST`, `PUT`, and `DELETE`) are used to specify the type of action that needs to be performed on the data. For example, to get information about a book with an ISBN whose last six digits are 524657, the REST address could look something like `http://servername/Book/524657`. Rather than return an HTML page for this address, the REST service would return an XML document containing all the information about the book.