1

A Quick Glimpse Behind the Canvas

In This Chapter

- Displaying a Canvas on the World Wide Web
- Using JavaScript to draw on your Canvas
- Looking at how it all works

Developers of the Canvas standard could have named their new creation anything they wanted. They could have called it an *area, pad, space,* or any number of other possibilities. They chose *Canvas,* with its obvious artistic implications. There is significant technical detail behind the implementation of Canvas, but at its core, the intent is to provide a new way to satisfy the urge for expression not so different from that behind prehistoric drawings on cave walls.

HTML5 Canvas is a digital version of the surfaces that have teased the imagination of mankind for thousands of years. Adding the power of computing to traditional media concepts creates an amazing combination. No need to imagine horses galloping or geese flying. Now they can move.

In this chapter, you get a first look at how HTML5 Canvas works and how to create your own compelling Canvas applications. You get a glimpse behind the Canvas to see what makes it tick.

Displaying Your Canvas on the Stage of the World Wide Web

Traditional artists working with traditional canvases face a tough task getting their work out into the world. They have to create their paintings, find galleries to show them, hope people notice them, and then maybe sell a few. This is a slow process indeed compared to the global exposure given to an HTML5 Canvas application.

Your Canvas application will sit within the framework of the Internet and World Wide Web. It will have instant and automatic distribution. Get a little buzz going and it could be seen by millions.

So what exactly *is* HTML5 Canvas? Well, stated briefly, HTML5 Canvas is a standard for applications written in JavaScript that run within a web page downloaded from a server and displayed by a browser on a client device. That's quite a mouthful and a bit difficult to absorb. As they say, a picture is worth a thousand words. Figure 1-1 shows how all the HTML5 Canvas pieces fit together.

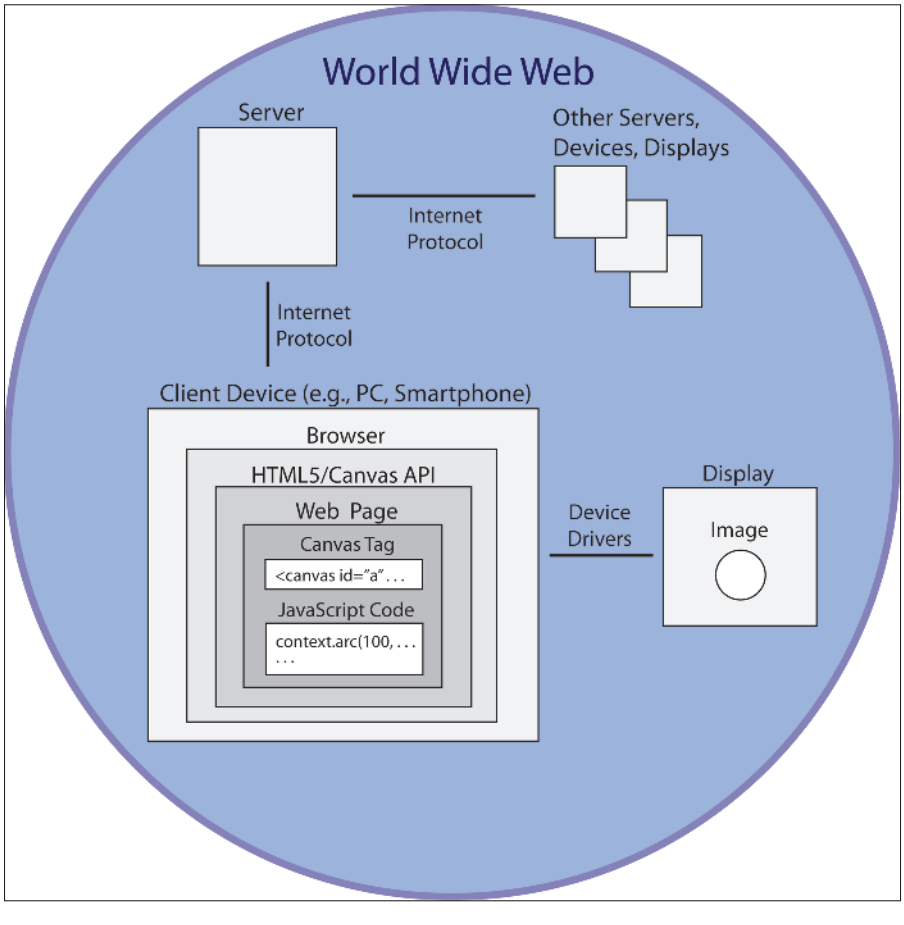

Figure 1-1: HTML5 Canvas on the World Wide Web.

Five great things about HTML5 Canvas

Canvas is a much anticipated feature of HTML5. Here are just a few of the great things about it:

- **It's interactive.** HTML5 Canvas can listen for and respond to user actions. Moving a mouse, pressing a button, tracing with a finger — these can all be sensed by your JavaScript code and used to drive application actions.
- **It's animated.** Objects can be made to move on an HTML5 Canvas . . . from simple bouncing balls to complex animations.
- **It's accessible.** All the major browsers now support HTML5 Canvas. Your Canvas application can be used on devices ranging from large computers to smartphones and tablets.
- **It's flexible.** An HTML5 Canvas can be used to display text, lines, shapes, images, videos . . . all with or without animation. It's a super-flexible medium.
- **It's growing rapidly.** HTML5 and the Canvas feature are steadily gaining popularity.

Client devices

Client devices include computers such as desktop PCs and mobile devices such as smartphones, tablets, and laptops. The client device is where your web browser resides and your Canvas is displayed. The website defining your Canvas is hosted on a server. Your web pages are downloaded from the server and displayed by your web browser.

Web browsers

Web browsers are software applications that construct and display web pages based on HyperText Markup Language (HTML) instructions. Major web browsers and their download sites are described next.

Desktop browsers

Desktop browsers can be downloaded from these developer sites:

- **Internet Explorer:** `www.windows.microsoft.com/en-US/internet-explorer/downloads/ie`
- **Firefox:** `www.mozilla.org/firefox`
- **Chrome:** `www.google.com/chrome`
- **Safari:** `www.apple.com/support/safari`
- **Opera:** `www.opera.com`

Mobile browsers

Mobile browsers are loaded on to mobile devices as apps that are downloaded from stores that are specific to the brand of device. To find browsers on app stores, search the store for *browser* or for a specific browser name, such as

- Internet Explorer
- Firefox
- Chrome
- Safari
- Opera

Websites and web pages

Websites are made up of web pages defined by HTML elements called *tags*. HTML tags define web page layout and content, including your Canvas. Listing 1-1 shows the code used to create the display in Figure 1-2. This is a simple example, but it demonstrates the basics needed to create a Canvas display.

Figure 1-2: A simple Canvas display.

Listing 1-1: Saying Hi on a Canvas

```
<!DOCTYPE HTML>
<html>
<head>
<script>

// WINDOW LOAD function.
window.onload = function()
{
   // CANVAS definition standard variables.
   canvas  = document.getElementById("canvasArea");
   context = canvas.getContext("2d");

   // MESSAGE details centered on canvas.
   var mText = "Hi!"
   var xPos  = canvas.width/2;
   var yPos  = canvas.height/2;

   // TEXT format details.
   context.font         = "80pt Comic Sans MS";
   context.fillStyle    = "lime";
   context.textAlign    = "center";
   context.textBaseline = "middle";

   // FILL text.
   context.fillText(mText, xPos, yPos);
}
</script>
</head>
<body>

<div style = "width:200px; height:200px;
              margin:0 auto; padding:5px;">

<!-- CANVAS area definition -->
<canvas  id    = "canvasArea"
         width = "200"  height ="200"
         style = "border:2px solid black">
<!-- MESSAGE if browser doesn't support canvas -->
Your browser doesn't currently support HTML5 Canvas.
Please check www.caniuse.com/#feat=canvas for
information on browser support for canvas.
</canvas>

</div>
</body>
</html>
```

Here are some of the HTML tags used in Listing 1-1 and throughout the book:

- `<!DOCTYPE HTML>`: Declares the document for a web page
- `<html>`: Delineates HTML code
- `<head>`: Defines code containing information about your web page
- `<script>`: Delineates code areas such as your Canvas JavaScript
- `<body>`: Defines the main area of your webpage
- `<div>`: Provides web page formatting information
- `<canvas>`: Defines the Canvas area

HTML5

HTML5 is the latest version of the HyperText Markup Language. HTML defines how web pages function and how they're displayed. HTML5 contains many new and exciting features added to the previous version, HTML4. Major aspects of HTML5 include:

- **Improved interaction with the user:** Provides for fancier forms and more flexible user input.
- **Improved support of audio and video:** Provides native support for audio and video.
- **Geolocation:** Your application can determine where the client device is located if the device has location-sensing hardware.
- **Client-side data storage:** Your application can temporarily store data on the client device.
- **Canvas:** Powerful graphics display. Canvas is one of the most anticipated and important new HTML5 features.

For a complete list of new HTML5 features, visit

```
www.w3.org/TR/html5-diff/#new-elements
```

Canvas

The Canvas feature of HTML5 enables you to add dynamic displays within defined areas of your web pages. These displays can include sophisticated shapes, colors, text, video, audio, animation, and more. It's limited only, as they say, by your imagination.

The evolution of computer graphics

Computer graphics have been around since as far back as the 1960s. In the 1970s, video games began a push in the sophistication of graphics that continues today. However, until the introduction of Canvas, browser-based graphics relied mainly on vector manipulation. Vector graphics, such as Scalable Vector Graphics (SVG), draw images based on lines and curves defined by sets of data. Canvas, by contrast, is a "bit map" technology in which images are drawn based on the definition of the individual pixels (picture elements) of objects. This provides a greater degree of control over the display images. In practical terms, the Canvas bit map technology results in faster and more efficient rendering of displays that have large numbers of objects. This makes the development of browser applications such as games much more feasible.

The initial versions of Canvas were implemented by individual browser developers. The Apple WebKit browser was the first in 2004, followed by the Gecko (Firefox) browser in 2005 and the Opera browser in 2006. The HTML5 implementation of Canvas creates a common standard across all browsers. The most recent releases of all major web browsers support HTML5 Canvas.

To define Canvas areas (single or multiple) within your web page, use the new HTML5 `<canvas>` tag. Identify each Canvas with a unique `id`, as in this example from Listing 1-1:

```
<canvas  id    = "canvasArea"
         width = "200"  height ="200"
         style = "border:2px solid black">
</canvas>
```

JavaScript code

It's your JavaScript code that will draw images on your Canvas. Without JavaScript, a Canvas is just a blank space.

JavaScript, developed by Netscape in the mid-1990s, is a different language than Java, although its developers were influenced by Java, which was developed by Sun Microsystems in the early 1990s.

Here is the JavaScript from Listing 1-1 that created the display in Figure 1-2:

```
// WINDOW LOAD function.
window.onload = function()
{
   // CANVAS definition variables.
   canvas  = document.getElementById("canvasArea");
   context = canvas.getContext("2d");

   // MESSAGE details centered on canvas.
   var mText = "Hi!"
   var xPos  = canvas.width/2;
   var yPos  = canvas.height/2;

   // TEXT format details.
   context.font         = "80pt Comic Sans MS";
   context.fillStyle    = "lime";
   context.textAlign    = "center";
   context.textBaseline = "middle";

   // FILL text.
   context.fillText(mText, xPos, yPos);
}
```

The JavaScript code in the sample listings is structured to be as easy as possible to understand. The focus of this book is on Canvas features and capabilities, not on programming languages. I've commented the code heavily and avoided complex coding structures. For a concise JavaScript language reference, see

```
http://docs.webplatform.org/wiki/javascript/tutorials
```

Device drivers

A *device driver,* which is usually built into client computers, smartphones, and tablets, is a software/firmware layer between an application and a device, such as a display. The display driver does the work of displaying the individual pixels that form the images on your Canvas.

Displays

Displays can be built into the client device or function as separate devices attached to the client. There's a huge variety of displays ranging from those that are measured in feet to those measured in inches. Displays are made up of individual picture elements (pixels), each of which is controlled by the device driver to show a color specified by software on the client device, such as your Canvas application code.

The Interplanetary Internet

The World Wide Web may need a new name. It's about to reach out beyond our planet. NASA is developing a technology that extends the Internet and web to extraterrestrial locations. Known as the "Interplanetary Internet," it uses Disruption-Tolerant Networking (DTN) to sustain super long-distance connections without data loss. Get ready to draw images of "ET" on your Canvas. (Search for *DTN* on `www.nasa.gov` to get the latest updates.)

Images

Canvas images are constructed pixel by pixel *(bit mapped),* as opposed to *vector graphics,* which are drawn based on points along specified lines. The dimensions of a Canvas are specified in pixels, and the images you create with your application code are based on pixel dimensions.

Remote devices

A *remote* device is located away from the client device containing your HTML and JavaScript code.

Internet Protocol

Client devices communicate with servers and other devices around the world via the *Internet Protocol* suite. Internet Protocol is the communications glue that holds the Internet and World Wide Web together. It's a layered structure of messaging and rules for exchanging information across telecommunications systems.

Servers

Servers are the computing devices that host the website that contains your Canvas application. Even though the task of constructing and displaying a web page and Canvas is delegated to the client device, the server plays an important role. The server stores your HTML and JavaScript code and downloads it to the client when a user selects one of your web pages for display. The server can also host images and data that can be retrieved and used by your Canvas application.

HTML5 Canvas applications

The applications for HTML5 Canvas are limitless. Here are a few important ones:

- **Advertising:** The interactivity and animation of HTML5 Canvas are ideal for attracting the attention needed for successful advertising.
- **Art & decoration:** HTML5 Canvas provides intricate control of color and images for creating artistic and decorative surfaces.
- **Education & training:** Text, images, videos, diagrams, and other HTML5 Canvas features can be combined to produce effective education and training applications.
- **Entertainment:** The web is growing as a platform for delivering entertainment. HTML5 Canvas video, images, and graphics are a great base for developing entertaining applications.
- **Gaming:** The HTML5 Canvas fine grain, pixel level control of displays, and the many methods for creating animation offer lots of possibilities for gaming applications.
- **Data representation:** HTML5 Canvas provides features to combine the power of access to global data sources with the imagery of graphs and charts.

Seeing a Canvas Application in Action

One of the great things about developing Canvas applications is how quickly you can see your work produce results. Here's an overview of the Canvas development sequence, which is covered in detail in Chapter 2:

1. **Create your HTML5 and JavaScript code using a text editor.**

   For example, you can use a text editor such as Notepad on a PC or TextEdit on a Mac.

2. **Save your Notepad or TextEdit file in a directory on your computer with the .HTML or .HTM extension.**

3. **Double-click the .HTML or .HTM file in your directory to display your Canvas.**

   Your Canvas appears in your default browser.

To get you jump-started with this process, access the examples for this book, as described in the following sections. The examples are self-contained, and each one includes all the code necessary to display a Canvas.

Using your browser to display a sample Canvas

You can download the sample applications in this book two ways:

- **Download the files from** `http://www.dummies.com/go/html5canvas`. After the download has completed, double-click the downloaded folder to open it and access the individual `.HTM` files. You can then move these files to any other folder you choose. To start an individual example, double-click the `.HTM` file.
- **Access individual example web pages:** Go to `http://donkcowan.com/html5-canvas-for-dummies#examples/` and click individual example page links.

As an example, to display the Canvas in Figure 1-2, access the code from Listing 1-1, as follows:

1. **Point your browser to** `http://donkcowan.com/html5-canvas-for-dummies#examples/` **to access the sample code for individual listings in the book.**

   You should see a list of samples, including Listing 1-1 Saying Hi on Canvas.

2. **Click the listing (for example, Listing 1-1 Saying Hi on Canvas).**

   You should see the Canvas display appear (refer to Figure 1-2). That's it — a simple two-step process to reach the sample code.

Using the sample code

The example listings have a number of uses:

- **Experimenting with the code:** The sample code is a great way to understand how Canvas JavaScript code functions. Modify the code and watch the effect on the display.
- **Using the code to seed your applications:** The samples provide a base for developing your own applications.
- **Referencing the code:** It's easy to forget how to code a particular Canvas task. A quick check of the sample code can be a big aid during application development.

To use the sample code directly on `http://donkcowan.com/html5-canvas-for-dummies#examples/`, do the following:

1. **Use your browser to access the sample listings from the book.**

   Follow the two preceding steps to access the sample you're interested in.

2. **Right-click (Ctrl+click on the Mac) on the page displaying the Canvas.**

   The menu of options for the web page appears (see Figure 1-3.)

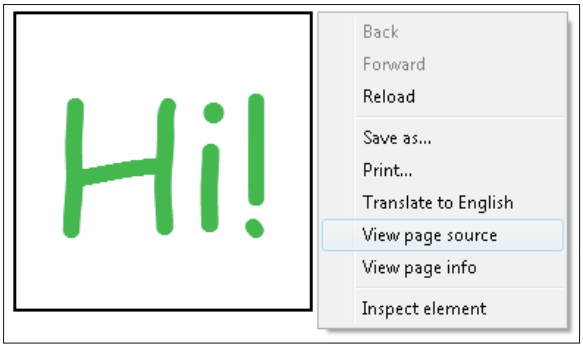

Figure 1-3: Canvas displays a menu of options.

3. **Click View Page Source to display the HTML and JavaScript code generating the Canvas display.**
4. **Highlight the code that you want to copy.**
5. **Right-click (Ctrl+click on the Mac) on the page displaying the code and select Copy from the options, as shown in Figure 1-4.**
6. **Open your text editor and paste the code into your application.**

   Right-click (Ctrl+click on the Mac) the page displaying the code and select Paste from the options.

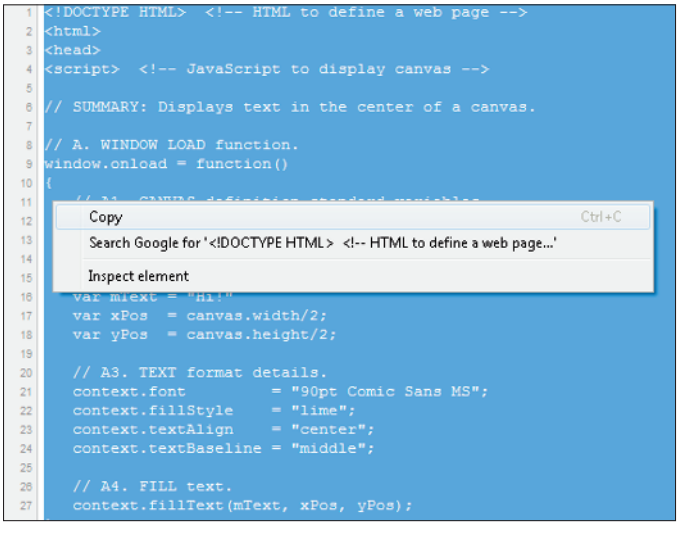

Figure 1-4: The Canvas code menu of options.

To download, save, and then use the sample code on your computer, do the following:

1. **Download the Zip file from `www.dummies.com/go/html5canvas` and store the files where you want them on your computer.**

   Move the entire folder or individual file to a new location if you choose to.

2. **Right-click (Ctrl+click on the Mac) the desired file and select Open With *your text editor* (Notepad, for example) from the options, as shown in Figure 1-5.**

   The file containing the code from the sample application opens in the text editor.

For your convenience, the image, audio, and video files referenced by the sample JavaScript code are stored on a server at `www.marketimpacts.com`. This means that the sample applications will execute without your having to adjust the code for file location.

The image, audio, and video files are also available in the Zip file you downloaded from `www.dummies.com/go/html5canvas`. To reference image, audio, or video files that you reuse from the download or create yourself, you'll need to change the JavaScript file references to point to the server where you store the files. I explain this process in more detail in Chapter 4.

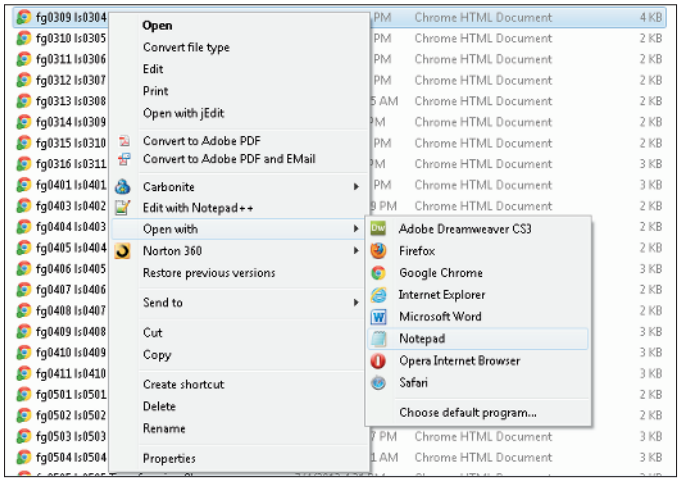

Figure 1-5: Opening the code file on your computer.

Why develop for HTML5 Canvas

HTML5 Canvas is an attractive platform for software development. Here are five reasons why.

- **Develop once, run anywhere:** HTML5 Canvas is supported by recent releases of the major web browsers, which run on a wide variety of devices from large computers to mobile devices. Code written using HTML5 tags and JavaScript code will work on all these devices.
- **Toolkit availability:** The tools needed to develop a Canvas application are not extensive or expensive — a computer, browser, text editor, and code debugger. For advanced development, such as sophisticated games, a number of third-party libraries are available to facilitate coding.
- **It's a well-accepted standard:** Although it will take time for all the features of HTML5 to be implemented in all browsers, HTML5 Canvas is a solid and accepted standard that will be around for many, many years.
- **Demand for interaction and animation:** Web users today want to interact with websites and see entertaining movement and animation. HTML5 Canvas gives developers a solid platform for serving these needs.
- **The mobile market:** HTML5 Canvas is increasingly supported on mobile devices. It offers a way to develop applications for smartphones and tablets without having to program for individual operating systems such as iOS and Android.