

Chapter 1

Introducing Facebook and the Facebook Platform

In This Chapter

- ▶ Getting to know Facebook
 - ▶ Exploring the Facebook Platform
 - ▶ Understanding what the Facebook API, FBML, FQL, and FBJS are
 - ▶ Figuring out where to get online developer information
-

We hear all of the time about “platforms” in various walks of life. Political parties put their policies and agendas into documents called *platforms*. An Olympic diver performs a perfect dive off of a 10 meter *platform* on the way to a gold medal. Heck, *platform tennis* is a form of tennis that people play with paddles.

Facebook too has its own platform — cleverly called the Facebook Platform. Fortunately for your coworkers, it is a platform that you don’t need a swimsuit or a paddle to use. You do need to know a Web programming language such as PHP or Java, however.

Facebook gained popularity because of its structured environment and social network, but its Facebook Platform is proving to be a critical means of preventing it from becoming the latest “flavor of the month.” Because of third-party applications, Facebook now offers a compelling reason for users to invest themselves in Facebook.com in a way that they were never really able to do with social networking sites like MySpace. In fact, upon the Platform’s release, it was only a matter of weeks before users began to see thousands and thousands of Facebook-inspired applications from all sorts of developers — from major corporations to hobbyists working in their basements.

In this chapter, I introduce you to the basics of Facebook and its development platform. If you are an application developer just coming to this social networking site as a newcomer, I get you up to speed by surveying the core concepts and components of Facebook itself. Next, I survey the Facebook Platform and its various parts and show you how they work together to form a cohesive solution.

Get your platform shoes on and let's take a hike!

Discovering Facebook

MySpace and Friendster may have been the early “go to” places for online social interaction, but Facebook has overtaken them as the fastest growing social networking site on the Web. Its structured environment, enjoyable user experience, and expandable platform for third-party applications have helped it gain this level of importance and popularity.

Before you begin to develop applications for Facebook, you should get to know all of the ins and outs of Facebook itself to ensure you fully understand the potential of how your application can tap into its platform.

If you are a newcomer to Facebook, you need to get your arms around two important concepts: the *News Feed* and the *profile*.

News Feed

After you are logged in to Facebook, the Facebook home page (www.facebook.com) displays the News Feed, as shown in Figure 1-1. Think of the News Feed as your own personalized news channel — something like a FNN (Friends News Network), if you want to get clever.

The News Feed contains a live list of announcements or stories about the activity of your network of friends on Facebook — whom they befriended, what apps they added, what their Status is. For example, if my friend Jared Allen became friends with Ally Horinsky, I would receive the following story:

```
Jared Allen and Ally Horinsky are now friends.
```

Facebook compiles this list of news stories based on several factors — the activity of your friends, your preferences of story types, frequency settings on specific friends, the privacy levels of your friends, a user's opinion on the quality of a story (known as “thumbs up” and “x” votes) — all mixed together into a behind-the-scenes, super-secret algorithm. A user can determine the frequency of certain news stories, but Facebook ultimately retains control over what is placed on the News Feed. Facebook also places social ads inside of the News Feed.

In this book, you discover how your application can add news stories to the News Feed.

The News Feed



Figure 1-1:
The News Feed contains a list of news stories about your network of friends.



Requests and notifications

The Facebook home page devotes the right sidebar column to listings of requests, notifications, birthday reminders, pokes, status updates, and more. You can use this space to quickly glance

through items requiring your attention. You can also access these notifications through the profile.

Profile

Each Facebook user has an individualized profile page, such as the one shown in Figure 1-2. A profile page is a user's "real estate" on Facebook.com, a home page for the user, friends, and others. Users decide what information to show or restrict based on who the visitor is.

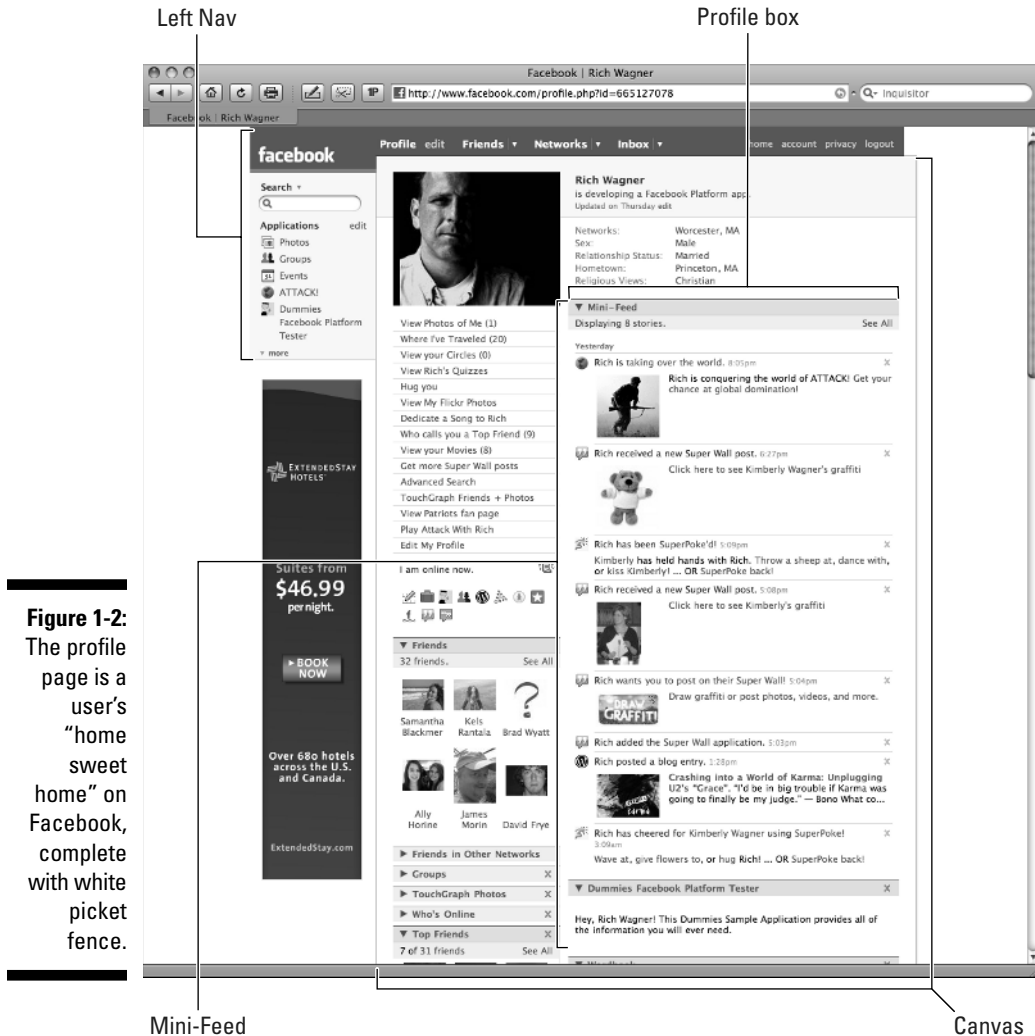


Figure 1-2: The profile page is a user's "home sweet home" on Facebook, complete with white picket fence.

The Facebook page contains a variety of elements. Some of the most notable for application developers include the following:

- ✔ **Left Nav:** The Left Nav sidebar (measuring 150 pixels in width) contains an Applications menu in which users can quickly access applications. Users can drag and drop their five favorite apps into the topmost section.
- ✔ **Canvas:** The canvas is the 650-pixel section to the right of the left side menu and below the top navigation menu. Not only is the profile page contained within the canvas, but every Facebook application is included as well.
- ✔ **Mini-Feed:** Mini-Feed contains a snapshot summary of the activities of a user. Notice the difference from the News Feed: Whereas News Feed on Facebook.com displays stories of friends, a profile's Mini-Feed presents stories only related to the user. As a Facebook developer, you can publish a Mini-Feed story using the API.
- ✔ **Profile box:** Profile boxes on a user's profile page consist of built-in Facebook boxes (Mini-Feed, Friends, Education, and Work) and Facebook applications. They can either be wide (400 pixels) and be displayed in the main profile section, or narrow (200 pixels) on the left sidebar. A user can drag and drop profile boxes to various parts of the page and can remove an application's profile box without deleting the application. Figures 1-3 and 1-4 show the My Flickr profile box in wide and narrow views, respectively.



Figure 1-3:
My Flickr
application
displayed
as a wide
profile box.

Figure 1-4:
My Flickr
displayed
in the left
sidebar.



✓ **Wall:** You can think of the Wall as a “bulletin board” of sorts that people can use to post notes, comments, or other feedback about a person. As a Facebook developer, you cannot post directly to the Wall inside of your application. However, you can provide a Wall attachment for your users. A Wall attachment provides a way for users of your application to add application-specific content to their Walls.

For example, the iLike application provides an Attachment that makes it easy for users to add music to their Walls. For example, by clicking the Add Music button (see Figure 1-5), a user can select a song to add to his or her Wall to share with friends (see Figure 1-6).

Figure 1-5:
Facebook
developers
can add
Attachments
to the Wall.

The iLike Add Music button



Nearly everything about Facebook is oriented toward social networking with friends. This topic is so key to Facebook that I devote the next section to it. So, as you begin to think about developing an application for Facebook, you should always consider how you can best take advantage of the social, viral nature of the Facebook Platform.

Figure 1-6:
Attachments help
get your
application
noticed.



Discovering the Pulse of Facebook: The Social Graph

The term *social network* has become so popular and familiar during the past couple of years that even non-techie people understand what the term means. Facebook, however, has its own vernacular when it comes to understanding social networking. They call it the *social graph*. The social graph can be defined as the interconnections that exist among family, friends, and acquaintances that every person has.

Take me as an example. I have a wife and three boys, parents who live in Indiana, a sister who lives in Ohio, many friends from my church, acquaintances from my town, coworkers at previous places of employment, and so on. Some of these people have interconnections of their own. My wife knows my sister, and my kids know my parents. Others members of my network do not know each other. My boys, for instance, have never met my former boss, Sal.

Social relationships like these are bound together by some type of interdependency between each other. It can be family, faith, friends, common interests, geography, or line of work. But there is always some sort of tie between each other, however loose.

In daily life, people place great value on their social graph. If I am looking for a job, I probably begin by networking my friends and former coworkers or acquaintances in the high tech world. If I am looking for someone to watch our dog while we are on vacation, I am willing to trust the niece of a close friend to do the job well, even though I barely know the niece personally. Moreover, when I am looking for a dentist, I ask around and get recommendations from people whose opinions I value within my social graph.



Online communities like Facebook and MySpace seek to re-create this natural human phenomenon in the digital world. Within these communities, you establish these links between people based on some sort of interdependency. Through the community services, you are able to get to know what your friends are up to, the music they are listening to, the movies they recommend, and what apps they are using.

However, the practices, behavior, and culture of these various social networking sites impact how well they emulate the real world. On MySpace, for example, I may collect hundreds or even thousands of connections, but only a handful are truly “friends” with any degree of interdependency. In contrast, in an environment like Facebook, which does not promote “friendship collecting” per se, I may currently only have a few dozen friends, but nearly all of those connections are backed by real-world relationships. As a result, I place much more weight on the opinions and activities of my friends on Facebook than I would on a site like MySpace. This value of the social interconnection is significant for the Facebook application developer creating social-enabled apps.

One issue that is curious to many is the use of the word “graph” in the term *social graph*. The term *network* in *social network* is intuitive enough, but the word *graph* conjures up images of something you create in Microsoft Excel for your boss.

Although *graph* may be a synonym for *chart* in everyday usage, Facebook uses mathematical definition of the term. In this context, you can think of a graph as an abstraction for modeling relationships between sets of nodes.

What Is the Facebook Platform?

The Facebook Platform is a standards-based programming framework that enables developers to create applications that interact and integrate with core Facebook services.

Points of integration

When you create an application for the Facebook Platform (see Chapter 2 for how to get started), you can integrate it at several points in the framework. Take, for example, Super Wall, one of the most popular third-party Facebook applications available. Consider the ways in which it is integrated into Facebook itself — so much so it becomes harder and harder to distinguish between apps created by Facebook and those developed by third parties. Here are several points of integration:

- ✔ **Display in the Application directory:** As Figure 1-7 shows, Super Wall is listed in the application directory after the developer adds it to Facebook.
- ✔ **Provide an About page:** An application provides an About page to explain to users what the application is and what it can do for them. Facebook also lets potential users know which of their friends have already added this application. The Super Wall about page (see Figure 1-8) provides an image and brief product description.

The Super Wall app

Figure 1-7: Every Facebook application can be listed in the directory.

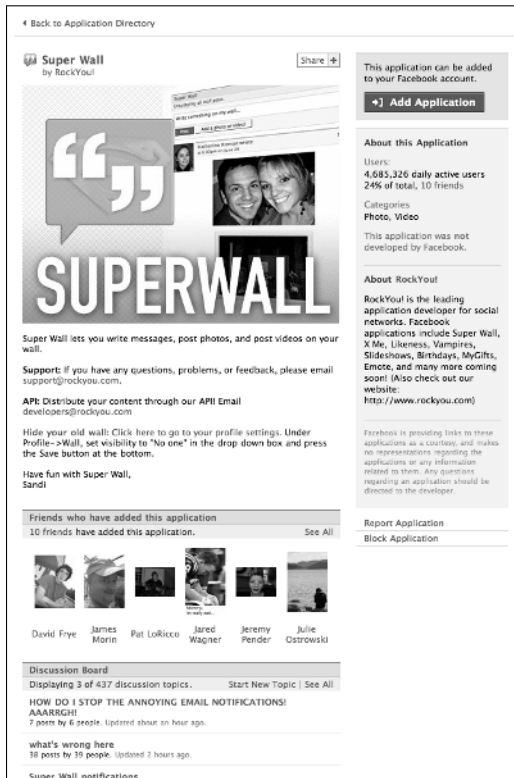
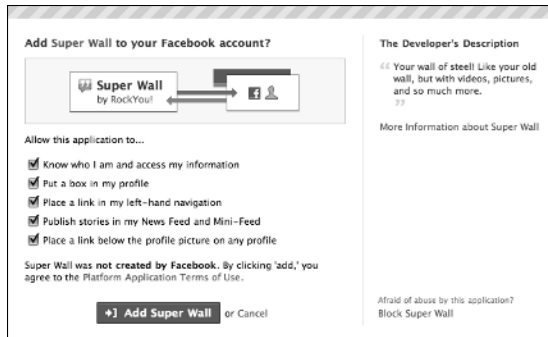


Figure 1-8: The Super Wall About page invites users to install the application.

➤ **Give the user a choice:** Users always have the last word on determining the level of integration that your application is permitted to have with their accounts. By default, Facebook encourages users to provide full integration. (See Figure 1-9.)

Figure 1-9:
Users
decide
how much
integration
an
application
can have.



- **Invite friends:** With access to the friends data of a user, an application can ask users if they wish to invite their friends to add the application as well. You can easily understand the viral growth nature of this functionality. Figure 1-10 shows the Super Wall friends invitation request.
- **Display inside Facebook Canvas Pages:** Facebook applications are displayed within the Facebook canvas. As you can see from Figure 1-11, Super Wall gets the entire Facebook page, except for the top menu and Left Nav. This presentation layer is provided to Facebook either as Facebook Markup Language (FBML, an HTML-like language) or as an `iframe` element.

Figure 1-10:
Application
developers
encourage
users to
invite their
friends.

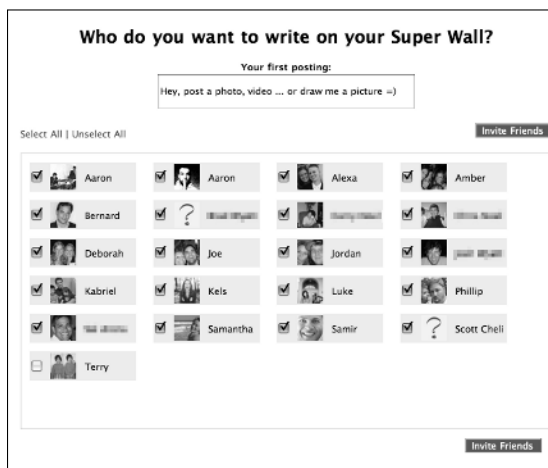




Figure 1-11: Facebook Canvas Pages allow third-party apps to seamlessly integrate into the Facebook presentation environment.

- ✓ **Appear in the Left Nav:** Applications can have their name and icon displayed in the Left Nav menu on the left side of the page (see Figure 1-12). Applications can be launched by clicking these links. Users can drag and drop applications, placing their top five apps in the topmost section.
- ✓ **Display a profile box:** Applications can have a profile box that is displayed on the user's profile. Although in this case a user does not run your application inside of the profile box, you can provide updated information useful to the user or actions for launching the application itself (see Figure 1-13).

Super Wall icon

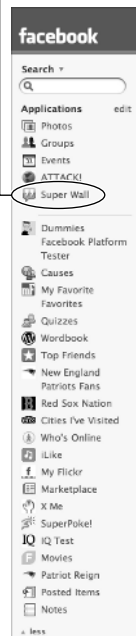


Figure 1-12:
The Left Nav is the Facebook application launcher.

Super Wall profile box

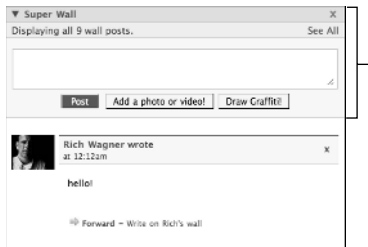


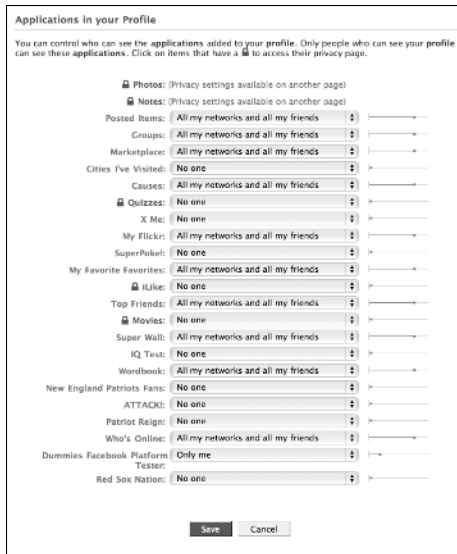
Figure 1-13:
The Super Wall profile box provides frequently requested actions for a user.

➤ **Create profile action links:** Also on a user's profile page, an application can add items in the Profile action links list, located just under the user's picture. For example, as Figure 1-14 shows, Super Wall adds an action link for inviting friends.



- ✔ **Access and post to News Feed:** Applications can post stories to the user's News Feed to communicate with a user's friends. For example, Super Wall sends a story out to my friends when I write something new on my wall. Each application has a maximum number of stories that can be on a News Feed at a time.
- ✔ **Send e-mail alerts:** If a user allows this activity, applications can send e-mail alerts and notifications to users.
- ✔ **Generate requests:** Applications can create new requests that require a response from a user. Often a request takes the form of an invitation from one user to another to add an application or perform an action.
- ✔ **Create message attachments:** As I describe in the "Discovering Facebook" section earlier in this chapter, message attachments can be added to Facebook message composers, such as the Wall. Using this point of integration, applications can provide app-prepared content into the composer.
- ✔ **Integrate into Facebook privacy settings:** In the privacy settings of their profiles, users can specify who can view their applications' profile boxes — everyone, some networks, only friends, only themselves, or no one. Application developers can also add additional privacy settings inside of their own applications. (See Figure 1-15.)

Figure 1-15:
Users can
configure
the level
of privacy
desired for
applications.



Facebook application architecture

Facebook applications are not installed directly onto the Facebook server. Instead, they are placed on the developer's server and then called by Facebook when the application URL is requested.

To interact with applications, Facebook uses a *callback metaphor*. Let me explain. The URL of your application is associated with a registered application in Facebook. When the Facebook application URL (such as `www.facebook.com/dummies`) is requested, Facebook redirects the request to your server. Your application processes the request, communicates with Facebook using the Facebook Application Programming Interface (API) or Facebook Query Language (FQL), and returns Facebook Markup Language (FBML) to Facebook for presentation to the user inside of its canvas.

Figure 1-16 shows the architecture of a typical Web application, whereas Figure 1-17 displays the architecture of a Facebook application.

Figure 1-16:
Architecture
of a standard
Web
application.

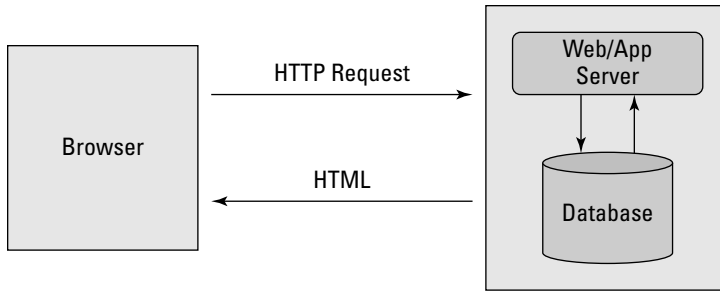
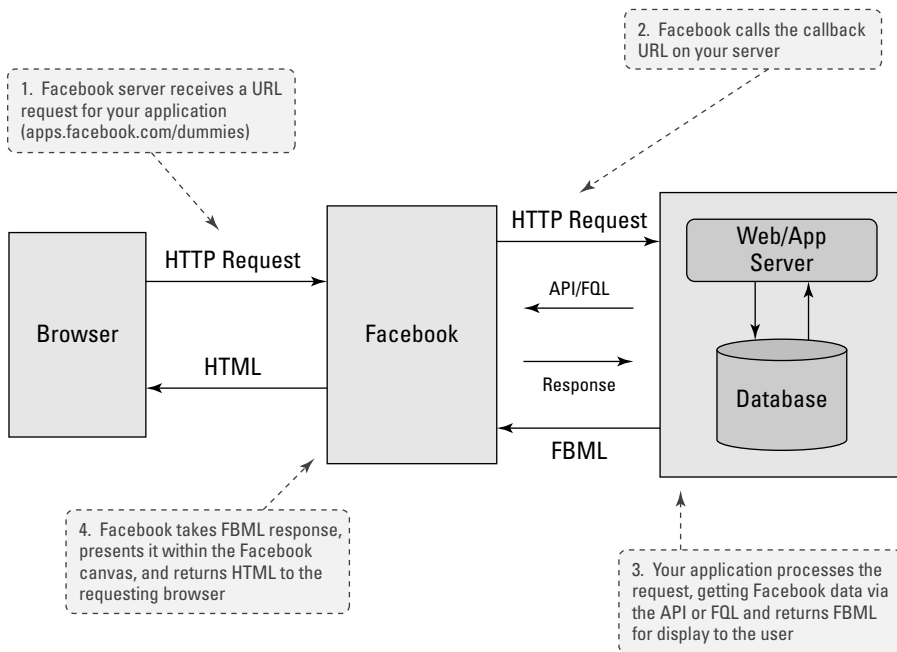


Figure 1-17:
Facebook
application
architecture.



Exploring the Framework Components

The Facebook Platform consists of four components that you work with inside your code:

- ✓ Facebook API
- ✓ FBML (Facebook Markup Language)
- ✓ FQL (Facebook Query Language)
- ✓ Facebook JavaScript

Each of these is described in the following sections.

Facebook API

The Facebook API is a Web services programming interface for accessing core services (profile, friends, group, event, photo) and performing other Facebook-centric functionality (log in, redirect, update view). The API is based on a REST-based architecture (which I explain in Chapter 3).

Facebook officially supports client libraries for PHP (4 and 5) and Java. These can be downloaded at developers.facebook.com/resources.php. Several unofficial client libraries have been developed for most every major Web programming language, including ASP.NET, ASP (VBScript), ColdFusion, C++, C#, D, Emacs List, PHP4, Lisp, Perl, Python, Ruby on Rails, VB.NET, and Windows Mobile. To download these unofficial versions, go to wiki.developers.facebook.com.

When you use a client library, you use language-specific methods to access Facebook services. For example, here's how you can retrieve the name and picture of a user and then display it on a page:

```
$user_data = array('name', 'pic', 'activities');  
$current = $facebook->api_client->users_getInfo($user, $user_data);  
echo "<p>Your name is {$current[0]['name']} </p>";  
echo "<p>Here is what you look like: <img src='{$current[0]['pic']}' /></p>";
```



Don't concern yourself with the details for now. Just notice that the user-specific information is accessed using the `$facebook->api_client->users_getInfo()` call.

Chapter 3 provides the details you need to work with the Facebook API.

Facebook Markup Language

Facebook Markup Language (FBML) is an HTML-like language used to display pages inside of the Facebook canvas. Here are three basic facts about FBML:

- ✓ FBML contains a subset of HTML elements. Such common elements as `p`, `ul`, and `h1` are part of FBML.
- ✓ FBML provides qualified support for `script` and `style` elements. Normal JavaScript is not supported in a `script` element, although Facebook JavaScript (see the “Facebook JavaScript” section below) does permit some scripting capabilities. You cannot use the `style` element to link to external style sheets, but you can use it for internal CSS.
- ✓ FBML also includes several proprietary extensions for Facebook-specific UI elements as well as program flow.

For example, the `fb:name` element displays the name of the user based on the value of the `uid` attribute:

```
<p>Your name is: <fb:name uid="665127078" useyou="false" /></p>
```

Some FBML elements help control the flow of a program. For example, `fb:if-can-see` only displays the content inside of it if the user has permission to see a specified feature (the `what` attribute) for the specified user (the `uid` attribute). Consider the following:

```
<fb:if-can-see uid="665127078" what="profile">  
  <p>You are granted access, you lucky one!</p>  
  <fb:else>Go somewhere else, you unfortunate one.</fb:else>  
</fb:if-can-see>
```

See Chapter 4 for full details on working with FBML.

Facebook Query Language

Facebook Query Language (FQL) is a SQL-based interface into Facebook data. You can access many Facebook database tables, including `user`, `friend`, `group`, `group_member`, `event`, `event_member`, `photo`, `album`, and `photo_tag`.

The language itself is similar to standard SQL, though with a few restrictions:

- ✔ `SELECT` statements must be performed one table at a time.
- ✔ Join queries are not permitted.
- ✔ A query must be indexable.

You can actually perform most of the same data access routines using either API or FQL. To that point, several of the API calls are nothing more than wrappers for FQL queries. FQL even has some advantages over the standard API interface. It reduces bandwidth and parsing overhead and can help minimize the number of data requests in complex situations.

Here's an example of using FQL to retrieve a list of friends of the current user (where the `$user` variable is the current user):

```
$friends = $facebook->api_client->fql_query("SELECT uid, name FROM user WHERE uid IN (SELECT uid2 FROM friend WHERE uid1=$user)");
```

If you want to discover more about FQL queries, turn to Chapter 5.

Facebook JavaScript

FBML does not support the use of standard JavaScript using the `script` element. However, Facebook JavaScript (or FBJS, for short) allows for limited scripting functionality inside a Facebook application. Here are some of the differences between FBJS and JavaScript:

- ✔ Instead of using the standard JavaScript DOM, Facebook provides its own alternative DOM implementation.
- ✔ Many FBJS object implementations are the same as standard JavaScript, although there are some differences. For example, instead of accessing a JavaScript property (such as `document.href`), FBJS uses a pair of `get` and `set` methods instead (`getHref`, `setHref`).
- ✔ When Facebook processes scripting code inside of `script` elements, it tacks on the application ID to function and variable names. Facebook does this to create an application-specific scope.

Facebook JavaScript does provide AJAX and Facebook dialog objects for developers to take advantage of.

Chapter 6 provides full details on how to work with Facebook JavaScript.



Three amigos: Developer centers on Facebook.com

You should bookmark these three useful developer links on Facebook.com:

- ✓ `facebook.com/developers` is the “home base” for Facebook developers. You can set up a new application, access your list of installed applications, and read the latest news and developer discussions.
- ✓ `developers.facebook.com` is the official documentation center for Facebook Platform developers. You can access all of the Facebook-approved reference material here.
- ✓ `wiki.developers.facebook.com` is a user-based Facebook developer’s Wiki, providing access to documentation, FAQs, sample code, and more.

