Working with CoreLocation

The iOS utilizes several hardware and software technologies to allow you to locate it in the world, tell you which way you’re traveling, and know which way the phone is pointing. The underlying technologies are fairly complex, but Apple abstracts most of this complexity away and gives us an easy API called CoreLocation to get location and heading information.

In this chapter, we’re going to look at working with the location features in the iOS via the CoreLocation API.

All the code in this chapter is taken from the Example_CoreLocation companion code and application. While this example is simple, the application possibilities for this data are endless. For example, you could write an application that computes the distance and direction to known locations from where the user is, or you could create an application that shows which stars are in the sky based on the user’s location and heading. The applications that can be built are limited only by your imagination.

Under the Hood

Under the hood, the iOS utilizes the following location technologies:

- **Wi-Fi Positioning Service (WPS):** Available on all iPhones since an early OS update, and all iPad devices. WPS uses a database lookup of nearby Wi-Fi access point MAC addresses with known locations. It then computes the location based on a proprietary algorithm. In a densely packed urban center, with lots of Wi-Fi access points, the accuracy of WPS is often 20–30 meters.
Global Positioning System (GPS): Available since the iPhone 3G and iPad 3G models, GPS uses a system of orbiting satellites that broadcast time and location signals to Earth. GPS receives these signals and then performs triangulation based on the latency of the signals. GPS is much more reliable than WPS in that it will work anywhere in the world where the GPS receiver has an unobstructed view of at least three satellites. The accuracy of GPS is usually within 20 meters, since the year 2000, when the U.S. government stopped degrading location data for civilian use.

Compass: Available since the iPhone 3Gs and all iPad models, the compass enables the device to know its orientation to the magnetic poles (magnetic heading). Coupled with its location information, it can also tell you its true heading, based on the known magnetic variation of the location.

Having to work with these directly would be pretty tedious, but fortunately for us, all of this functionality is wrapped up in the CoreLocation API. MonoTouch exposes this API via the MonoTouch.CoreLocation namespace, and the most important class is CLLocationManager.

Usage Pattern

The general pattern for working with CoreLocation is as follows:

1. Instantiate a CLLocationManager object.
2. Configure any settings on CLLocationManager, such as the level of accuracy you want.
3. Wire up event handlers to the update methods on the CLLocationManager object to handle the location updates (or assign a CLLocationManagerDelegate if you want to use the delegate pattern as described in Chapter 6).
4. Tell CLLocationManager to start feeding you location and/or heading updates.
5. Do something interesting with the location information.
6. Tell CLLocationManager to stop updating you with information (this saves on battery, as we’ll examine later).

Instantiating CLLocationManager

The CLLocationManager class is the workhorse of the CoreLocation API, but it’s very easy to instantiate, as it has no arguments in its constructor. For example, Listing 15–1 comes from the Example_CoreLocation companion code and application.