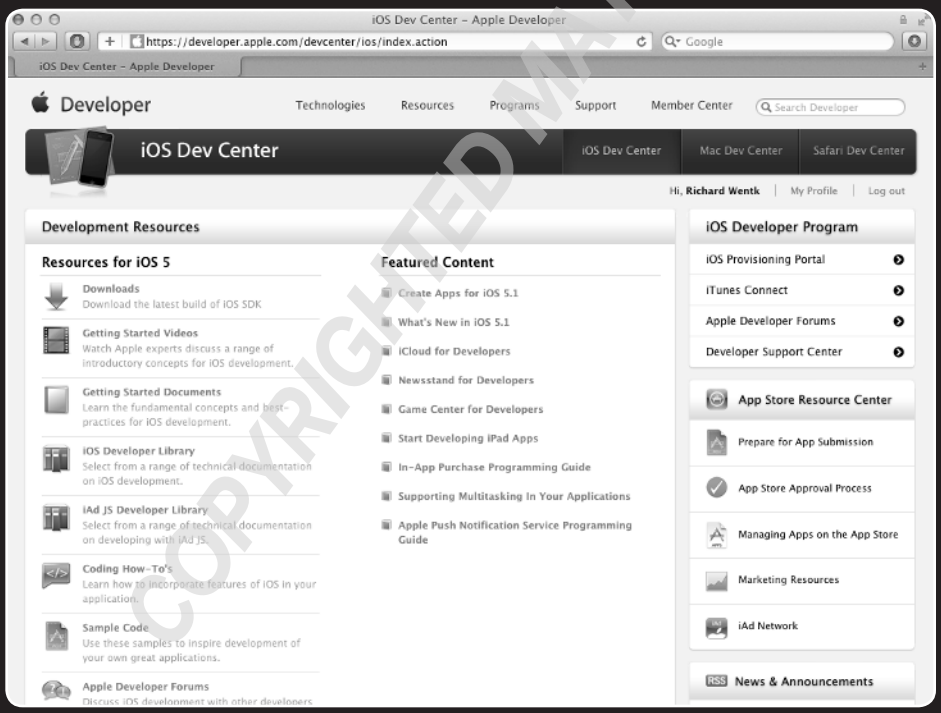
How Can I Start Developing for iOS?

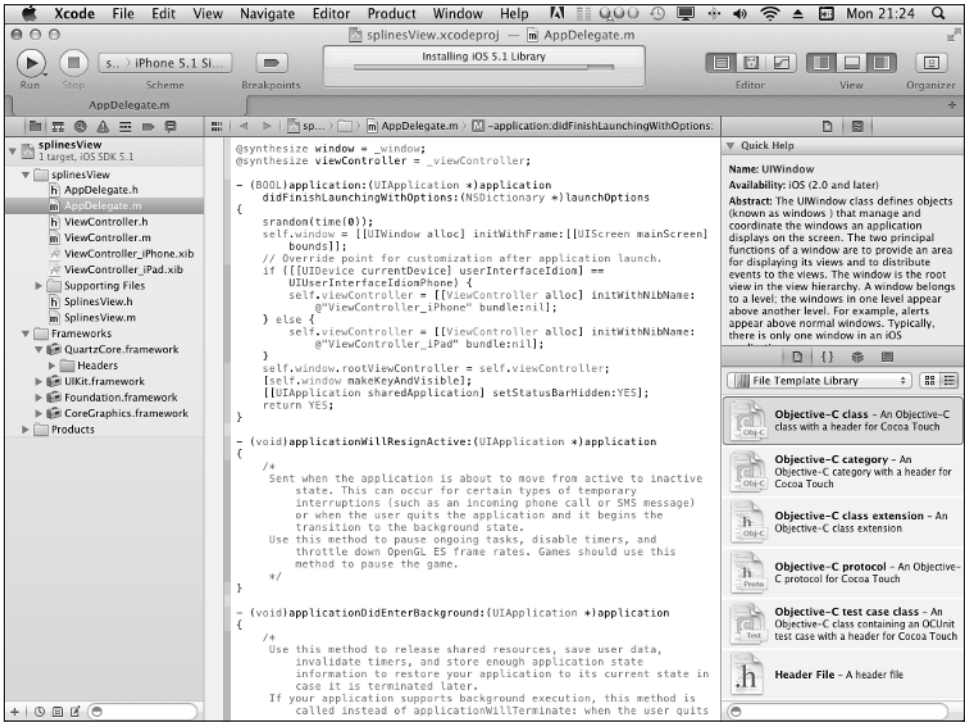


App development relies on a single unified development tool called Xcode. Xcode is free for the curious, but experienced and dedicated developers can get advanced information about updates to iOS by paying a small annual fee to enroll in Apple's iOS Developer Program. This chapter explains how to install Xcode, how to enroll as a developer, why you need lots of bandwidth for downloads, and why sometimes you'll have more than one version of Xcode installed at the same time.

Getting Started with App Development

iOS development doesn't have to be expensive. In fact, you need only two tools to develop iOS apps:

- **A recent Mac.** Although you can use any recent Mac, some models and specifications are more productive than others. You can find more details later in this chapter.
- **A copy of Xcode.** Xcode, shown in Figure 1.1, is Apple's app development toolkit. It has much in common with a standard Mac application, but it has some unusual extra features you'll learn about later in this book. Xcode is available free from the Mac App Store.



1.1 Xcode is a big, complex, powerful application with many features. Fortunately, you don't need to master everything it does to develop apps successfully.

Two further assets are optional, but very useful:

- **An iOS device.** You might expect this to be essential, but it isn't. Xcode includes the *Simulator*—a tool that can run and test iOS apps in an OS X mockup of either an iPhone, shown in Figure 1.2, or an iPad.
- **Membership in the iOS Development Program.** This program for dedicated developers costs $99 a year.

Although membership in the iOS Development Program is optional, it offers important benefits:

- **Pre-release (beta) versions of iOS.** Apple often makes the latest versions of iOS available to developers before they're released to the public. This option also gives developers time to make new apps ready for sale as soon as the new version of iOS is released to the public or to modify existing apps to fix possible incompatibilities.
- **The latest versions of Xcode.** Similarly, Apple makes new versions of Xcode available to developers before they're released to the public.

1.2 The Simulator built into Xcode can run iOS software, but its features are limited, and it isn't a substitute for testing apps on real devices.

- **Live device testing.** Only members of the iOS Development Program can test their apps on real devices. The Xcode Simulator is adequate for basic testing, but it has significant limitations, so this is an essential option for dedicated developers.
- **Access to help and support information.** This includes developer forums, and developer support videos, some of which are taken from the WWDC (World Wide Developer Conference) of the previous year.
- **Access to the App Store.** Only members of the iOS Development Program can sell apps through the App Store.

Genius

Xcode can be used to develop both iOS and OS X applications. OS X developers have a separate paid-for program that gives access to the OS X App Store. But because you don't need special hardware to test OS X apps, you can create and test OS X apps with Xcode before you sign up as an OS X developer.

Choosing a Mac

Although you can use any recent Mac for iOS development, not all Macs are equally productive. If you are developing for fun or curiosity, this may not matter to you, but it's useful to understand the benefits and limitations of each possible choice.

Choosing a processor

Processor speed affects development times, but speed isn't as critical as it is for high-performance applications such as gaming and video editing. Most iOS apps are small and simple, and Xcode doesn't take long to convert your raw instructions—known as *source code*—into a working application.

Note

In Xcode, this process is called *building an application.*

Currently, Xcode runs on OS X Lion, so at a minimum your Mac must have a Core 2 Duo series processor with 64-bit addressing and two cores. An i-Series processor may give better performance, depending on clock speed and number of cores. But using more cores won't make a huge difference to your productivity, especially when starting out. There's no need to invest in an expensive multi-core MacPro, at least not until you're earning enough from your apps to justify spending your development budget on one.

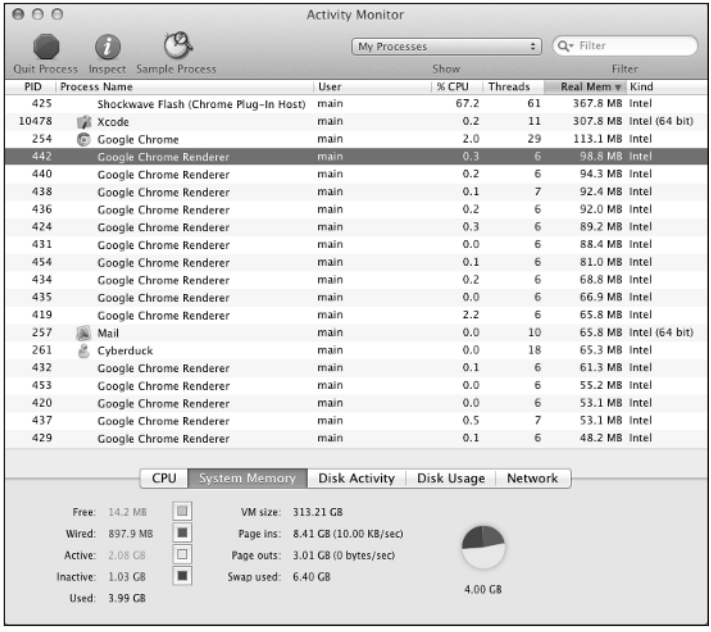
Note

Older versions of Xcode that work under Snow Leopard can be found online. If your Mac can't run OS X Lion, you can experiment with these, but you won't be able to use them to submit apps to the App Store.

Selecting memory

Xcode is a large application. If you're dedicating a Mac to development, it will run comfortably in 4GB. But if you plan to have multiple windows open in Safari or some other web browser (a good way to read Xcode's help files and documentation), you can easily run out of space with 4GB, as shown in Figure 1.3. A more realistic minimum is 8GB.

If you have too little memory, your Mac will stall for a minute or two every time you switch applications. If you plan to run Xcode, a browser, Mail, and perhaps some other applications simultaneously, you can avoid these distracting waits by expanding your Mac's RAM to 8GB.



1.3 This Mac is running Xcode and the Chrome web browser. It has just 14.2MB of free RAM, guaranteeing a long wait when swapping between applications.

Genius

You can check how much memory applications are using, and how much memory is free, by running the Activity Monitor utility in /Applications/Utilities.

Selecting a monitor

Monitor size has a direct and immediate influence on productivity in Xcode. The larger your monitor, the more windows you can have open simultaneously, and the less time you'll waste opening, closing, moving, and scrolling between various windows and panes.

When you use Xcode, you'll spend more time editing, searching the web, and typing than you will building. You may also need to run supporting software, including sound, image, and perhaps video editing tools, at the same time as Xcode.

When starting out, expect to spend lots of time searching and reading the Xcode documentation. The more information you can view at once, the more easily you can make sense of what you're learning.

So in practice, you'll be significantly more productive with a large monitor—even a relatively cheap, non-Apple monitor—than with a small one.

Another option is to use multiple affordable 22-inch or 24-inch monitors simultaneously. Two monitors will be more productive than one, and if you can resurrect an old PC or Mac to use as a documentation browser, three or more monitors will help you work even more efficiently.

Selecting other peripherals

Printing out code—the instructions that make up an app—is often seen as an old-fashioned memory aid. But it can be surprisingly helpful, because it can be difficult to hold the entire structure of an app in (human) memory at the same time.

The other essential peripheral is a backup device. Any external hard drive will do for backups. You can simply copy project folders to the hard drive as you go, or you can rely on Time Machine to automate backups for you.

You can also speed up development significantly by using an SSD (Solid State Drive)—a very fast hard drive with no moving parts. Many Mac models support SSD as an option. It's often possible to add an SSD to an existing Mac, although the cost is high compared to a standard hard drive, drive capacities are usually lower, and the upgrade can be physically difficult.

Caution

Code backups are *not optional*. If you have apps for sale, losing the source code for them can be a disaster.

Genius

Experienced developers keep their code online in a *repository*—a web site that stores code safely and manages versions. Optionally, you can set up a repository to share your code with other developers so they can contribute to your projects. There's more information about this in Chapter 13.

Selecting a form factor

The physical size and configuration of your Mac has a direct influence on how easily and quickly you master app development. Although portable development is tempting, a permanent workstation is always more productive.

Running Xcode on a laptop

You can run Xcode on a laptop, including the MacBook Air models shown in Figure 1.4. This can seem an appealing option, but the small screen and relatively slow processor will slow down development. Most developers find they're most productive in a quiet environment or one with customizable music and sound. Coffee shops and other public locations aren't ideal. However, a MacBook Air or MacBook Pro can be useful to demonstrate app development in business meetings.

1.4 Portable development with a laptop can seem attractive, but it's rarely as productive as working in a fixed location with a bank of large monitors.

Running Xcode on an iMac

An iMac, shown in Figure 1.5 is a good choice for development. It's relatively affordable, easy to work with, includes a monitor, and can be expanded with further displays. You'll find large-screen iMacs particularly productive. Make sure you use a model with at least 4GB of memory, but consider expanding it to 8GB if you can, especially if the iMac is your family's main computer and it's used regularly for other tasks.

1.5 An iMac can be an ideal development machine.

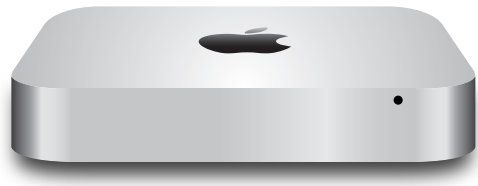
Running Xcode on a Mac Pro

As the name suggests, Apple's Mac Pro line is aimed at professional Mac users. There's no reason *not* to use a Mac Pro, shown in Figure 1.6, if you have access to one. But keep in mind the memory and processor requirements of Xcode and the fact that monitor size has more influence on development speed than raw processor power. It's likely you'll waste most of the power of a Mac Pro unless you pair it with two or more large monitors.

1.6 A Mac Pro is a good choice for professional developers with large budgets, but it's excessive for beginners.

Running Xcode on a Mac Mini

The Mac Mini, shown in Figure 1.7, is a small and convenient solution for small office and home use. Mac Mini models aren't outstandingly powerful, but they're small enough to be unobtrusive, and they can easily be paired with two monitors. Most models have 2GB or 4GB of RAM; consider expanding this for dedicated development.



1.7 A Mac Mini is ideal for light and medium performance applications, but may struggle as a professional web server for e-commerce.

Genius

It's well known among developers that the cheapest way to expand a Mac's memory is to use third-party memory products. They're significantly cheaper than Apple's own memory, but offer identical performance. If you want to do the job yourself, Apple's support web pages include detailed specifications and instructions. For example, http://support.apple.com/kb/ht1423 includes information about expanding the memory of an iMac.

Joining the iOS Developer Program

Enrollment is optional for casual developers who are experimenting with iOS, but essential for more dedicated developers who want to test apps on real iOS devices and also want access to the App Store.

Understanding enrollment options

There are two enrollment options. The $99 annual fee is the same for both, but they have different benefits:

- **Individual.** For this option, you need an existing Apple ID, although it can be useful to create a new one even if you already have one, because there are occasional incompatibilities between developer and public Apple IDs. Individual enrollment is usually completed within 48 hours. Personal ID is confirmed via your credit card details.
- **Company.** Enrolling as a company is more complex. In addition to an Apple ID, you'll also be asked to provide proof of incorporation, with a valid address. The company name must be legally registered: Trade names aren't allowed. Business enrollment can take up to two weeks, although it's usually completed more quickly.

Individual developers have simplified access to the *provisioning* process used to manage test devices. As an individual, you work as a team of one and all possible team roles are available to you.

This is a benefit when getting started, but may become a limitation if your apps become very successful and you try to hire other developers to help you.

Support for team development is available only for company enrollment. A designated team leader can add or remove team members. He or she can also limit access to test devices and control which members are allowed to upload finished apps to the App Store.

Each option has legal and financial implications as well. If you enroll as an individual, Apple pays you as an individual. This may affect your tax status. Some countries, especially the U.S., withhold a percentage of the earnings of foreign developers unless they apply for them to be released. This process can take between three months and a year to complete.

If you enroll as a company, you are taxed as a company, which is usually simpler and more direct.

To summarize, if you already have a company, the extra wait at the start of the development process is a relatively small price to pay for the extra team options and the simpler financial processing.

If you don't have a company or you have no need for team development, individual enrollment is a simpler option. It may cause issues later if your apps become very successful, but few app developers reach the stage where team development becomes necessary.

Enrolling in the iOS Developer Program

For individuals, enrollment isn't a complicated process. To begin, open a web browser and navigate to https://developer.apple.com/programs/ios, shown in Figure 1.8.

1.8 The iOS Developer Program sign-up page.

Click the Enroll Now button, and work your way through the questions. You'll be asked for your Apple ID, name, address, e-mail, nationality, and other basic information. Once enrollment is

complete and you pay the enrollment fee, you'll receive confirmation within 24 hours— although the confirmation e-mail usually arrives more quickly. You can then access the Developer Program features.

For companies, the process is similar, but you'll be asked to confirm details of incorporation, usually by faxing a copy of your documents to Apple's HQ in Cupertino. Company applications are checked manually, so the process takes longer. You may also be contacted by an Apple representative with some basic questions. Once enrollment is complete, you can access the Program features, but your account is set up with the extra team management features that aren't offered to individuals.

Note

Apple offers a separate Enterprise Program for large corporations that want to develop apps in-house and distribute them internally. Details are outside the scope of this book, but you can find the latest information at http://developer.apple.com/programs/ios/enterprise.

Will apps make me rich?

Realistically, it's unlikely that app sales will make you rich. iOS apps continue to be more successful than apps for any other platform, including Android. But the iOS app store is heavily saturated, and many users already have more apps than they want or need.

App sales are heavily concentrated among a small number of super-sellers, who can be extremely successful. A slightly larger number of app developers make four or five figures a year, especially if they combine app sales with ad revenue captured from Apple's iAD program. But many sellers are lucky to break even.

A successful app has four ingredients: a genuinely original idea or a popular existing idea; aggressive marketing, which may include a sizeable marketing budget; impressive graphic design; and a certain amount of luck. If you can adapt your apps for novel markets—for example, relatively few apps are written for the Chinese market—you can increase your chances of financial success.

However, at the time of writing, app developer skills are very much in demand among employers. While you may not be able to make a fortune selling your apps directly, you'll certainly be able to increase your employability. Some developer compensation packages are very generous, so it's well worth investigating this career option.

Downloading and Installing Xcode

The App Store and Developer Program versions of Xcode are different—sometimes significantly different—and they're installed in different ways. Before you get started with downloading and installation, let's look at a critical element of Xcode, the SDK (Software Development Kit).

Caution

Xcode is a continual work in progress. The versions of iOS it supports change with almost every release, but Apple also varies key features without notice, and delivery and installation options also change. The information that follows is valid for the public release of Xcode 4.3 for iOS 5.1. Don't be surprised if some of the details have changed again by the time you read this.

Understanding SDKs and betas

Xcode has two main components. The current toolset includes the tools you use to edit, build, test, and distribute apps. Xcode is a fairly mature application now, so these tools change relatively slowly.

As iOS develops, new features are added and old features become obsolete. The other component in Xcode is called the SDK (Software Development Kit), and it manages this information.

Internally, the SDK is a collection of files and supporting documentation that defines all the features in a single version of iOS. Whenever iOS is updated, Apple releases a new version of Xcode with an updated SDK.

This seems simple, but there are complications. Basically, Xcode is always available in two versions:

- **The public version in the App Store includes the SDK for the most recent public version of iOS.**
- **The Developer Program version usually includes the SDK for the *next* version of iOS.** This version is a work in progress. It includes provisional features that may change before the final release.

A critical fact is that the Developer Program version goes through multiple updates. Each update is called a beta or preview, and is available for a period that can vary between a week and a month. When a new beta version is released, it replaces the previous preview, which becomes obsolete.

Just before a final public release, Apple announces a special version called a Release Candidate (RC) version. Unlike earlier previews, the RC version can usually be used to submit apps with the new features to the App Store. Occasionally, the RC version goes through one or two further updates to fix last-minute issues. More typically, it's released "as is" after a couple of weeks. This gives developers a chance to submit compatible apps to the App Store and gives Apple time to review them.

Table 1.1 summarizes the process.

Table 1.1 Xcode and SDK versions

Xcode Release	Availability	SDK	App Store Submissions?
Public	App Store and Developer Portal	Current public version of iOS	Yes
Early Preview	Developer Portal only	Preview of next version of iOS	No
Release Candidate	Developer Portal only	Advanced preview—imminent public release	Usually yes

Caution

It would make sense for Apple to make each SDK update downloadable as a separate small file that could be installed in Xcode. Unfortunately, Apple doesn't do this. Whenever the SDK is updated, you have to download a complete new version of Xcode.

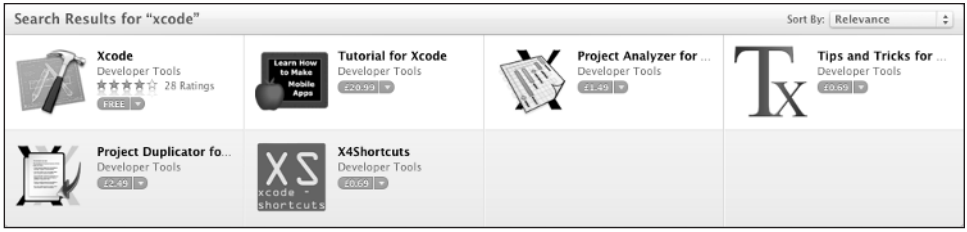
Note

Occasionally, Apple makes major changes to Xcode to update its editors and build tools. The last time this happened (in 2011), Xcode went through its own set of beta preview versions. Instead of an RC version, Apple eventually released a version called a GM (Gold Master) Seed. This process may happen again during the lifetime of this book.

Installing Xcode from the App Store

To install the current public version of Xcode from the Mac App Store, follow these steps:

1. **Launch the App Store application on your Mac.**
2. **Type** Xcode **into the search box at the top right, as shown in Figure 1.9.**
3. **Click the Free button.**
4. **Wait while the App Store downloads an extremely large file.**



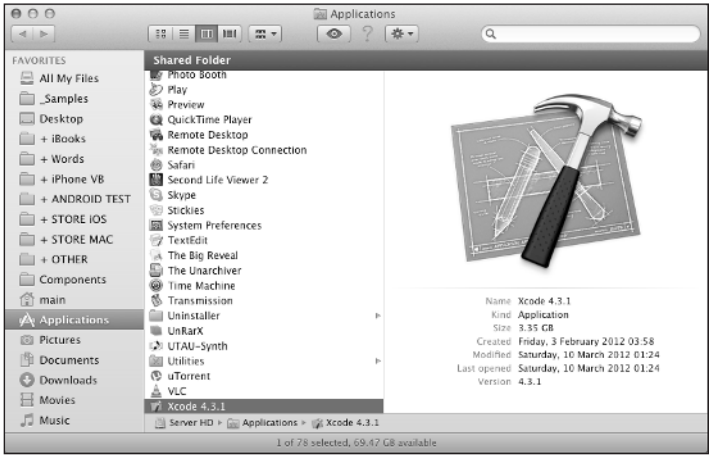
1.9 You can get Xcode for free in the Mac App Store, but beware of the download size!

5. **After the download completes, the App Store installs Xcode in /Applications, as shown in Figure 1.10.**

Caution

Xcode is currently 3.4GB. This is a big, big download, and if you don't have fast broadband, you'll probably need to leave it running overnight. If you don't have broadband at all, you won't be able to download it directly. If you have a laptop, it can be worth visiting a local WiFi hotspot with good connectivity, such as an Internet café, and downloading Xcode there. (Not all café owners appreciate very large downloads, so it's polite to check that this is okay before you start!)

6. **Double-click Xcode to launch it.** For convenience, you can also drag the application icon to your Dock.



1.10 The App Store downloads a separate installer and copies it into /Applications.

Installing Xcode from a developer download

Many developers find the App Store process inconvenient. From version 4.3, Apple rolled the complete Xcode package into a single, giant application that includes Xcode itself and assorted other tools you'll meet briefly later in this book.

The giant application is bundled into a single .dmg file. Installation is much simpler than it was: You simply mount the .dmg file in Finder and drag a single Xcode application file to a folder of your choice.

It's not yet clear if Apple is reserving this new installation method for developers or if future public App Store versions will work the same way. Whatever the outcome, the following steps summarize this alternative installation process:

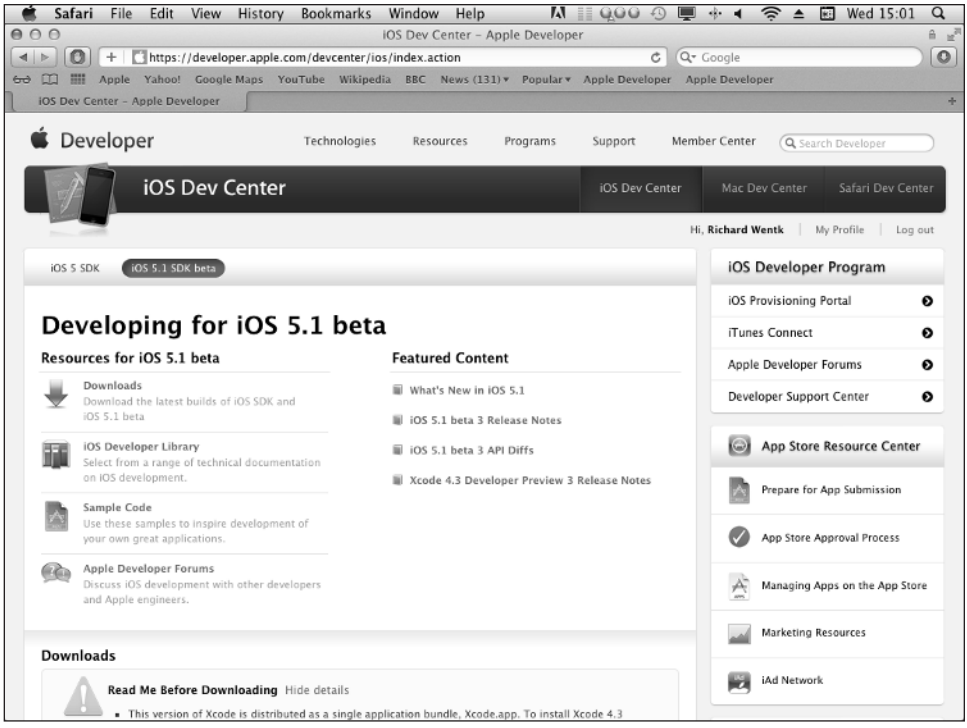
Caution

You must be enrolled in the iOS Developer Program to access the web pages and files shown in this section.

1. **Open a web browser, and navigate to the iOS Dev Center landing page at https://developer.apple.com/devcenter/ios.**
2. **Click the Log in button near the top of the page, and enter your Apple ID and password.**
3. **When a beta (developer preview) version is available, it appears as a link or button to the right of the main public version; click the button to access the beta area, shown in Figure 1.11.**
4. **Scroll down to the Downloads links.** You can also click the Downloads link near the top left.
5. **Click the Xcode and iOS download link to begin downloading Xcode.** Depending on your browser and your individual preferences, you may choose to download the file to a specific location. Otherwise, the file will be copied to /Downloads on your Mac.
6. **Wait…** until the download completes.
7. **Create a new folder on the root of your system disk.** The name doesn't matter, but "iOS Beta" is a good choice.
8. **Navigate to the downloaded .dmg file in Finder.**

Note

While you can install this version manually into /Applications, it's better to keep it in a different custom folder. You also may want to rename to avoid confusion with the current public version, especially if you also move it to the Dock.



1.11 The iOS beta/preview area is available only to members of the Developer Program.

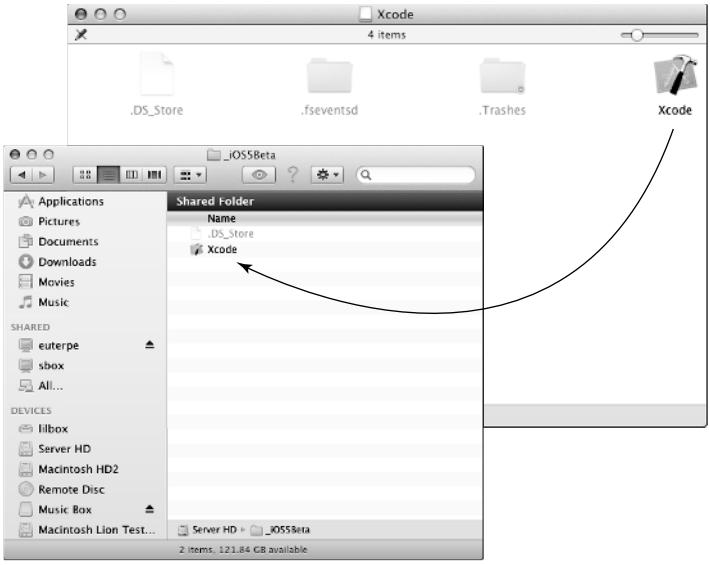
9. **Double-click the .dmg file.** The contents should appear in a new window. You'll see the .dmg file appear under the Devices list at the left of Finder.
10. **Drag the Xcode file from the new window to the folder you created in Step 7, as shown in Figure 1.12.**
11. **Double-click the Xcode file to launch it.** You can also drag its icon to the Dock for convenience.

Genius

If you enroll as a developer, you'll often have two versions of Xcode in the Dock at the same time—the public version, used for current app development, and a beta/preview. You can rename the application files by hand to tell them apart. For example, you can rename the preview version to Xcode Beta.

Caution

In addition to the /Developer and /Beta folders, you should also create one or more project folders for your apps. *Do not* save apps inside the Xcode folders; they can be deleted without notice when you install an update.



1.12 Currently, you can install the developer version of Xcode by dragging it to a folder of your choice.

Understanding other installation requirements

There's more to know about the preview process—and more files to download. In addition to the most recent version of Xcode with the newest SDK, you must also download the following:

- **A firmware update file for each of your test devices.** This file updates the device to run the latest preview/beta version of iOS.
- **Xcode's internal documentation files.** This process is described later in this chapter.
- **Optionally, a new version of iTunes.** Apple is moving to on-air updates of iOS devices, which means that iTunes is no longer required for iOS updates. However, this option isn't always available, so you may still need to download a special beta/preview version of iTunes before you can update the firmware in your devices.


Note

Developer versions of iTunes download and install like any other .dmg packaged application: Double-click the .dmg file to mount it, and then double-click the installer. One critical difference is that preview versions of iTunes usually include a built-in expiration date. After a preview expires, you can download the most recent version of iTunes from www.apple.com/itunes.


Note

Firmware is the name given to a device's internal software. Firmware is usually stored in a special memory that isn't dependent on battery power, so it isn't lost if the device loses all power. Technically, you *flash* firmware when you update it.

It should be obvious from this list that updating your development tools to the most recent SDK can mean lots of downloading. The total bandwidth for a new beta/preview SDK with Xcode, firmware for a single device, and associated documentation can total more than 5GB. If you have multiple test devices and need to update to a preview of iTunes, this can climb toward 7GB.

Updates can happen weekly. It's not obligatory to download every preview/beta; you can usually skip some of the earlier betas, but expect to download at least 15-20GB a month.


Caution

Some broadband ISP contracts limit the amount of data you can download each month. If your broadband supplier implements a bandwidth cap, consider switching to a supplier that doesn't limit you in this way.

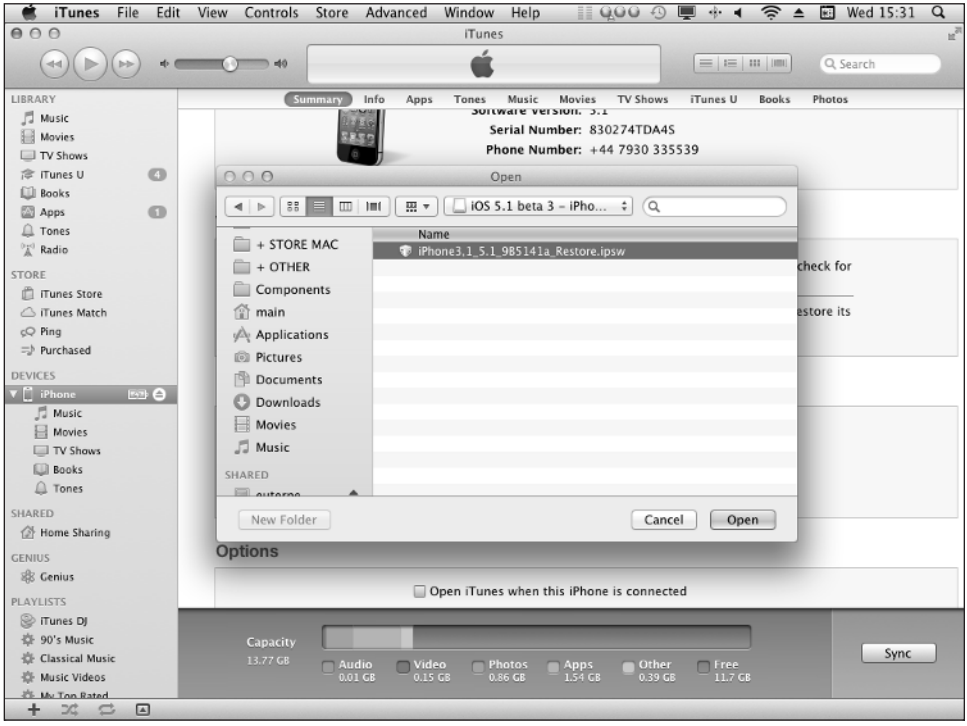
Updating devices

Firmware is supplied either as a .dmg file or as an .ipsw file. (To find the .ipsw file inside a .dmg file, double-click it to open it.)

To update a device to the latest preview version of iOS with iTunes, follow these steps:

1. **Connect your device to your Mac using a cable, and launch iTunes.** (Although iTunes offers backup over WiFi and to iCloud, cable connections are more reliable.)
2. **Select the device from the list at the left.**
3. **Click the option labeled Back up to this computer in the Backup pane.** Optionally, you can also set a password.

4. **Click the Sync button at the bottom right.** Wait for the backup to complete.
5. **Hold down the option button on your Mac's keyboard, and click Restore.**
6. **Navigate to the .ipsw file you downloaded, select it, and click Open, as shown in Figure 1.13.** If the .ipsw file was packed inside a .dmg file, select the .dmg file from the list of Devices at the left, and then select the .ipsw file.



1.13 Selecting a preview/beta version of iOS in iTunes to install it.

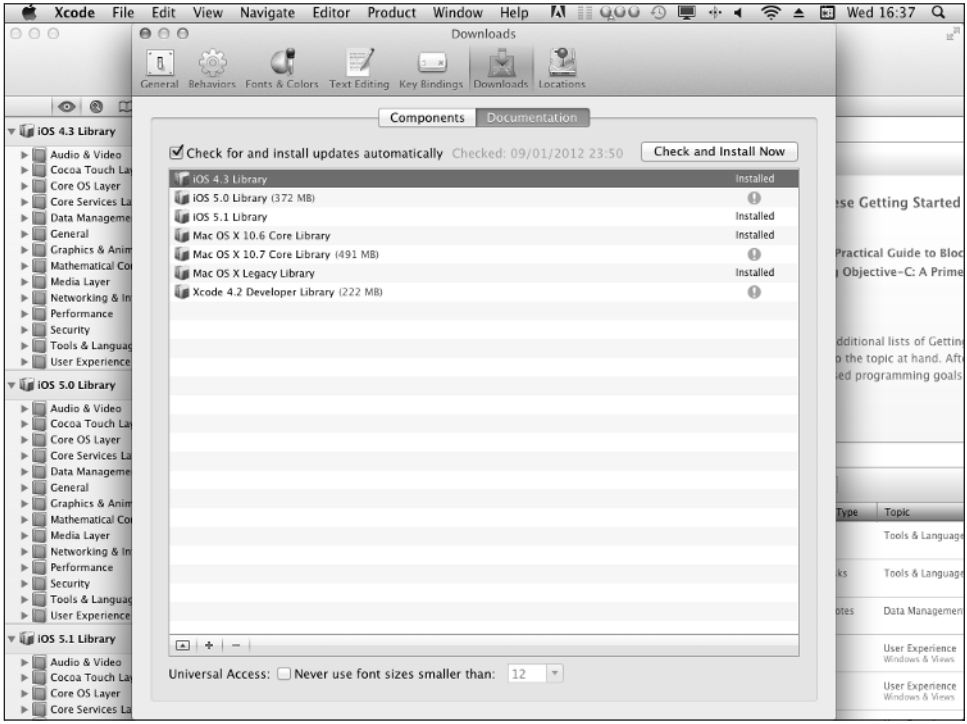
7. **Wait while iTunes flashes the new version of iOS to your device.** This can take 10-15 minutes.
8. **The Set Up Your (iOS Device) page appears automatically in iTunes.** Select your most recent backup from the menu on this page, and click Continue.
9. **Wait while iTunes restores your apps, data, and settings.** This can take another 15-20 minutes.
10. **Your device can now be used for testing with the latest developer version of Xcode.**

If you have multiple devices—for example, more than one iPhone, an iPhone and an iPad, an iPhone and an iPod Touch—you must update all devices you want to use for testing with the new version of iOS.

Installing documentation

Although Xcode is a huge download, it doesn't include all the documentation and help files you need. To install these files, follow these steps:

1. **Launch Xcode.**
2. **Select Xcode➪Preferences from the main menu.**
3. **Click the Downloads icon near the top of the pane.**
4. **Click the Documentation tab.**
5. **Click the Check and Install Now button, as shown in Figure 1.14.**



1.14 Reviewing the installed documentation. You don't need every possible file, just the more recent ones.

Each file is 300-400MB. You don't need to install every file; for example, you don't need the OS X 10.6 Core Library if you're developing iOS apps. But you do need the files for the current version of iOS and the Xcode Developer Library for help with Xcode itself.

Note

If you check the check box labeled Check for and install updates automatically at the top left of this page, Xcode downloads all the documentation it needs as soon as it can. However, the download process isn't bullet-proof, so you should always check manually that files have downloaded correctly.

Now that you've learned about the Developer Program, SDKs, and Xcode, you're ready to start exploring Xcode's developer-friendly features in more detail.