

Teil I

Der Lebenszyklus einer Mac-OS-X-Anwendung

In diesem Teil:

- **Kapitel 1**
Der Einstieg. 31
- **Kapitel 2**
Einfacher Workflow und passives Debugging 41
- **Kapitel 3**
Einfaches aktives Debuggen. 53
- **Kapitel 4**
Grundlagen der Kompilierung. 63
- **Kapitel 5**
Einstieg in eine Cocoa-Anwendung. 71
- **Kapitel 6**
Eine Cocoa-Anwendung: Views 89
- **Kapitel 7**
Eine Cocoa-Anwendung: Controller 103
- **Kapitel 8**
Versionskontrolle 123
- **Kapitel 9**
Property Lists. 151
- **Kapitel 10**
Libraries und Dependent Targets. 175
- **Kapitel 11**
File Packages und Bundles 187

■	Kapitel 12	
	Unit-Testing	199
■	Kapitel 13	
	Eine anwendungsspezifische View erstellen	213
■	Kapitel 14	
	Dynamic Libraries und Frameworks	235
■	Kapitel 15	
	Dokumentation in Xcode	253
■	Kapitel 16	
	Die Datenmodellierungswerkzeuge verwenden	277
■	Kapitel 17	
	Cross-Development	303
■	Kapitel 18	
	Spotlight (oder wie man ein Plugin erstellt)	317
■	Kapitel 19	
	Letzte Hand anlegen	337

Der Einstieg

In diesem Kapitel

- Ein einfaches Projekt erstellen
- Ein Programm ausführen
- Die Dateien, die von Xcode erstellt werden

1.1 Xcode starten

Nach der Installation können Sie Xcode durch einen Doppelklick auf das Xcode.app-Symbol im Verzeichnis `/Developer/Applications` starten. Xcode meldet sich mit dem Begrüßungsbildschirm `Welcome to Xcode` (siehe Abbildung 1.1).



Abb. 1.1: Der Begrüßungsbildschirm `WELCOME TO XCODE`. Das erste Tab, `GETTING STARTED`, enthält Ressourcen für die Xcode-Entwicklung. Das Tab `XCODE NEWS [RSS]` liefert Ihnen die neuesten Nachrichten über Ergänzungen der Entwickler-Dokumentation.

Von Xcode 3.0 zu Xcode 3.1

Xcode 3.0 hat für diesen Schritt einen separaten Assistenten verwendet (siehe Abbildung 1.2).




Abb. 1.2: Der Begrüßungsbildschirm in Xcode 3.0

Das erste Tab in dem neuen Xcode-3.1-Begrüßungsbildschirm, *Getting Started*, enthält Ressourcen für die Entwicklung mit Xcode. Die interessanteste Funktion des Fensters ist das Tab *Xcode News [RSS]*. Es enthält einen Minireader für einen RSS-Feed mit den neuesten Erweiterungen des Dokumentationssets der Apple Developer Connection (ADC). Apple veröffentlicht fast jede Woche neues Material für Entwickler. Deshalb ist dieses Tool nützlich.

Soll der Begrüßungsbildschirm nicht bei jedem Start von Xcode angezeigt werden, deaktivieren Sie das Kontrollkästchen *Show at launch* und schließen Sie das Fenster. Sie können den Begrüßungsbildschirm jederzeit über den Menübefehl *Help|Xcode News* aufrufen.

FileVault und Xcode

Wenn Sie Ihr Home-Verzeichnis mit Mac OS X FileVault verschlüsseln und FileVault aktiviert ist, kann die Performance von Xcode beim Kompilieren und Linken erheblich beeinträchtigt werden, weil bei jedem Lese- und Schreibvorgang die Verschlüsselung durchlaufen werden muss. Da Compiler-Objekte wahrscheinlich keine wichtigen Geheimnisse enthalten, ist dieser Aufwand überflüssig.

Um dieses Problem zu vermeiden, sollten Sie für Build-Zwischen- und -Endprodukte neue Ordner außerhalb Ihres Home-Verzeichnisses erstellen. Dafür kommt beispielsweise das */Users/Shared*-Verzeichnis in Frage. Öffnen Sie mit dem Menübefehl *Xcode|Preferences ...* () das Dialogfeld *Xcode Preferences* und wählen Sie das *Building*-Panel. Aktivieren Sie bei den Optionsfeldern *Place Build Products in:* und *Place Intermediate Build Files in:* jeweils die Option *Customized location:* und geben Sie Verzeichnisse Ihrer Wahl ein (siehe Abbildung 1.3).

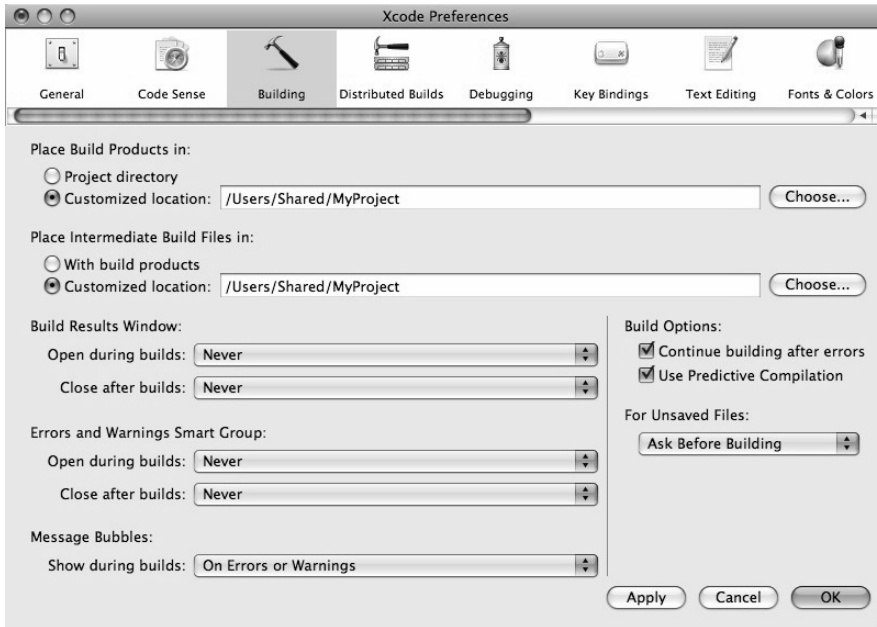


Abb. 1.3: Das XCODE PREFERENCES-Fenster. Wählen Sie hier alternative Verzeichnisse für Ihre Build-Zwischen- und -Endprodukte.

1.2 Hello, World

Traditionell werden neue Programmierwerkzeuge anhand einer Minianwendung eingeführt, die auf dem Terminal die Begrüßung `Hello, World!` ausgibt. Wir wollen diese ehrwürdige Tradition beibehalten. Wählen Sie den Menübefehl `File|New Project` (`⌘` `⌘` `N`). Xcode öffnet das New Project-Dialogfeld (siehe Abbildung 1.4).

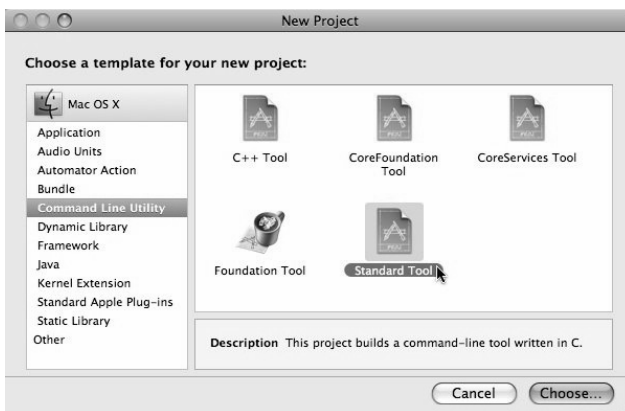


Abb. 1.4: Das NEW PROJECT-Dialogfeld. Wählen Sie links den Eintrag `COMMAND LINE UTILITY`, rechts oben `STANDARD TOOL` und klicken Sie dann auf `CHOOSE...`

Von Xcode 3.0 zu Xcode 3.1

In Xcode 3.0 war dieses Fenster erheblich unübersichtlicher (siehe Abbildung 1.5).



Abb. 1.5: Die Auswahl des Projekttyps in Xcode 3.0

Die Hauptbearbeitungseinheit in Xcode ist das Projekt. Ein *Projekt* umfasst eine Sammlung von Dateien, Tool-Einstellungen und Targets (dt. *Ziele*) zur Lösung einer Aufgabe. Ein Target gibt die Projektdateien an, die zu einem bestimmten Produkt gehören, und enthält die darauf anzuwendenden Tool-Einstellungen. Die meisten Anwendungsprojekte verfügen nur über ein einziges Target, nämlich die Anwendung selbst. Doch komplexere Projekte können mehrere Targets haben: für Libraries, Plugins und Unit-Tests.

Verschiedene Target-Typen erfordern verschiedene Tool-Einstellungen und System-Libraries. Xcode vereinfacht die Konfiguration eines neuen Projekts, indem es die gebräuchlichsten Target-Typen als erstes Target in dem Projekt anbietet. Wir wollen ein einfaches Command-Line-Utility (Befehlszeilenprogramm) erstellen, das in dem BSD-UNIX-Subsystem von Mac OS X läuft. Wählen Sie deshalb in dem Dialogfeld links den Eintrag Command Line Utility und oben rechts die Projekt-Template Standard Tool aus (siehe Abbildung 1.3). Klicken Sie dann auf Choose...

Hinweis

Die Liste der verfügbaren Projekt- und Target-Typen ist recht umfangreich. Einen gründlichen Überblick finden Sie in Anhang B, *Projekt- und Target-Templates*.

Im nächsten überlagerten Dialogfeld (siehe Abbildung 1.6) können Sie den Projektnamen eingeben und den Speicherort des Projekts wählen. Geben Sie hier den Projektnamen **Hello-**

World in das Save as-Feld ein. Wählen Sie im zweiten Feld das Verzeichnis. Hier wurde **xcodeuser**, das Home-Verzeichnis unseres fiktiven Xcode-Anwenders, als Projektverzeichnis gewählt. Xcode erstellt das Verzeichnis **HelloWorld**, falls es noch nicht existiert. Andernfalls werden Sie gefragt, ob Sie das bestehende Verzeichnis überschreiben wollen. Dann speichert Xcode die Projektdatei **HelloWorld.xcodeproj** und andere Dateien in diesem Verzeichnis, die zur Erstellung der Targets Ihres Projekts benötigt werden.



Abb. 1.6: Geben Sie im Feld **SAVE AS** den Namen Ihres neuen Projekts ein. Wählen Sie in dem Feld **WHERE** den Speicherort des Projekts. Das Projekt wird dort automatisch in einem separaten Ordner mit dem Namen aus dem ersten Feld gespeichert.

Von Xcode 3.0 zu Xcode 3.1

In Xcode 3.0 wurde dieser Schritt über ein separates Fenster des Assistenten abgewickelt (siehe Abbildung 1.7).



Abb. 1.7: Eingabe des Projektnamens in Xcode 3.0

Klicken Sie auf den Save-Button. Xcode erstellt das HelloWorld-Verzeichnis, kopiert einige Dateien hinein und öffnet das Projektfenster (siehe Abbildung 1.8). Bei einem BSD-Command-Line-Utility besteht das Projekt aus einer `main.c`-Datei mit der `main()`-Funktion, einer `HelloWorld.1`-Template-Datei für die *man*-Page und dem HelloWorld-Produkt. Das Produkt wird ursprünglich in Rot angezeigt, weil Xcode die zugehörige Datei nicht findet. Dies ist natürlich, weil wir sie noch nicht erstellt haben.

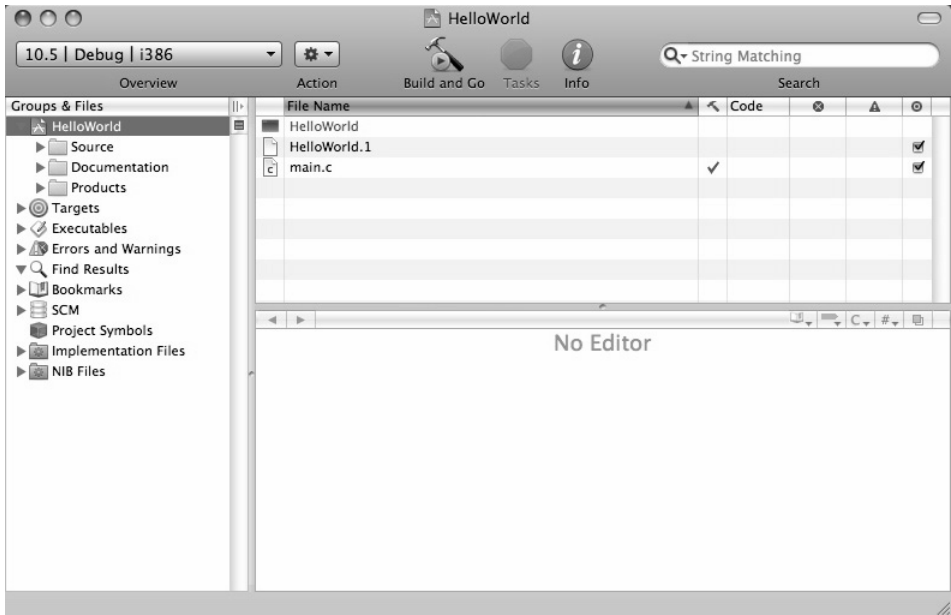


Abb. 1.8: Das HELLOWORLD-Projektfenster. In der Spalte GROUPS & FILES auf der linken Seite werden diverse zu Gruppen zusammengefasste Dateien angezeigt. Die Liste rechts oben zeigt Details zu dem Eintrag an, den Sie in der GROUPS & FILES-Spalte auswählen. Der Bereich links unten ist der Editor-Bereich (falls die Datei editierbar ist). Der Editor-Bereich kann mit dem Splitter vergrößert oder verkleinert werden.

Die in der linken Groups & Files-Spalte ausgewählte Datei wird ausführlich auf der rechten Seite des Projektfensters angezeigt. Wenn Sie links den ersten Eintrag unter der Überschrift auswählen (hier HelloWorld), zeigt die Detail-Liste alle Dateien in dem Projekt an. Unter dem Projekt-Icon befinden sich Ordner-Icons, die Untergruppen von Dateien repräsentieren. Wenn Sie ein Ordner-Icon anklicken, werden rechts die Dateien in dieser Gruppe angezeigt.

Links neben den Projekt- und Ordner-Icons befinden sich die vertrauten Dreiecksymbole, mit denen Sie die Untereinträge der Ordner anzeigen oder verbergen können.

Von Xcode 3.0 zu Xcode 3.1

In Xcode 3.0 enthält die Toolbar des Editor-Fensters zusätzlich die beiden Icons Build und Editor (siehe Abbildung 1.9). Sie können die ältere Version wiederherstellen, indem Sie den Menübefehl `View|Customize Toolbar...` wählen und die fehlenden Icons in die Toolbar ziehen. Doch im Rest dieses Buches wird nur noch die Xcode-3.1-Version verwendet!

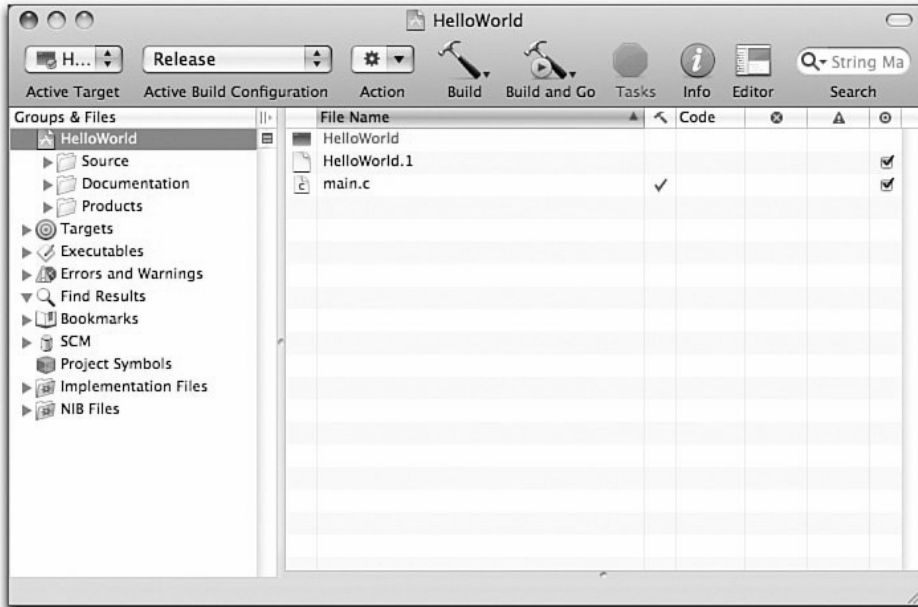


Abb. 1.9: Das Editor-Fenster mit der Toolbar von Xcode 3.0

Klicken Sie, falls erforderlich, auf das Projekt-Icon am Anfang der Groups & Files-Spalte, um die Dateien in der Detail-Liste auf der rechten Seite anzuzeigen. Doppelklicken Sie jetzt auf `main.c`. Die zugehörige Datei wird in einem separaten Fenster angezeigt (siehe Abbildung 1.10).

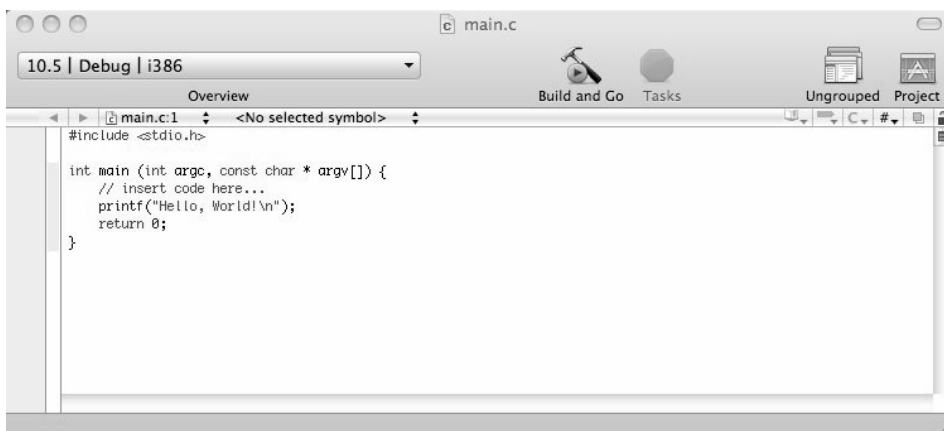


Abb. 1.10: Ein Editor-Fenster mit der Standarddatei `main.c` aus der Xcode-Template für das Command-Line-Utility

Die Standarddatei `main.c` für das Command-Line-Utility enthält eine `main()`-Funktion. Der Befehl `printf("Hello,World\n");` dient als Platzhalter, damit das Programm überhaupt

etwas Erkennbares macht. Später werden Sie ihn durch eigene Befehle ersetzen. Durch den Platzhalterbefehl wird die Ausführung unseres ersten Xcode-Programms beträchtlich erleichtert. Damit Sie die Ausgabe des Programms auch sehen können, wählen Sie zunächst den Menübefehl Run|Console aus oder drücken Sie `⇧⌘R`. Xcode öffnet ein Konsolenfenster für standardmäßige Texteingaben und -ausgaben.

Das Konsolenfenster verfügt (wie alle anderen Fenster) am oberen Rand über eine Toolbar. Klicken Sie den Build and Go-Button in dieser Toolbar an. Es können mehrere Sekunden vergehen – der erste Build eines Targets dauert immer länger. Doch bald werden Sie mit einem Hello, World! in dem Konsolenfenster belohnt (siehe Abbildung 1.11). Erfolg.

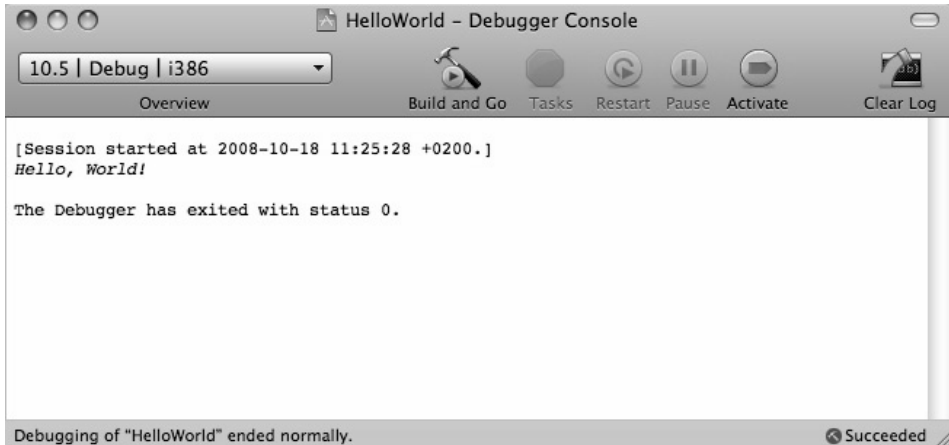


Abb. 1.11: Das Konsolenfenster mit dem Output des Programms: Hello, World!

Verlassen Sie Xcode (indem Sie die Tasten `⌘Q` drücken oder den Menübefehl Xcode|Quit Xcode auswählen). Da es nichts zu speichern gibt, werden Projekt- und Editor-Fenster sofort geschlossen.

1.3 Was wurde wo gespeichert?

Rufen Sie im Finder den HelloWorld-Ordner in Ihrem Home-Verzeichnis auf (siehe Abbildung 1.12). Das HelloWorld-Verzeichnis befindet sich an der Stelle, die Sie bei der Anlage des Projekts spezifiziert haben. Es enthält unter anderem die Dateien HelloWorld.1 und main.c, die bei der Anlage des neuen Projekts automatisch angelegt wurden. Das blaue HelloWorld.xcodeproj-Icon repräsentiert die Projektdokumentdatei. Mit einem Doppelklick auf das Icon können Sie Xcode öffnen und das HelloWorld-Projekt in dem Zustand anzeigen, in dem Sie es verlassen haben.

Das build-Verzeichnis enthält ein Release-Verzeichnis mit dem HelloWorld-Tool und einen Ordner namens HelloWorld.build. Letzterer enthält etwa ein Dutzend Dateien, einschließlich des kompilierten Objektcodes von main.c und mehrerer Indexdateien, die das Navigieren in großen Projekten und in System-Libraries erleichtern. Sie können das HelloWorld.build-Verzeichnis ignorieren. Es wird ausschließlich intern in Xcode verwendet.

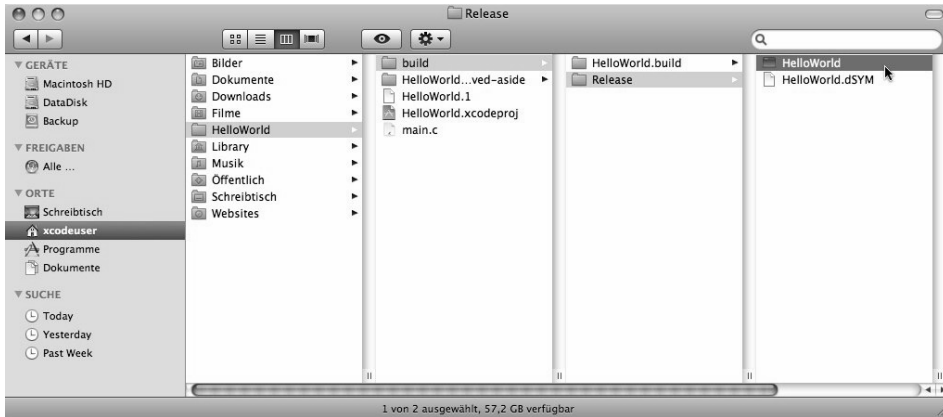


Abb. 1.12: Das HelloWorld-Projekt im Finder. Wenn Sie ein Command-Line-Utility in Xcode erstellen, werden ein Ordner für das Projekt, eine Projektdatei, einige Template-Dateien und ein build-Verzeichnis mit dem fertigen Tool erstellt.

Das HelloWorld-Tool ist eine ausführbare UNIX-Datei, die Sie in einem Terminal aufrufen können. Öffnen Sie die Anwendung `Terminal.app` im Verzeichnis `Programme/Dienstprogramme` und ziehen Sie das Icon des HelloWorld-Tools aus dem Finder in das Terminal-Fenster. Dies hat denselben Effekt, als würden Sie den kompletten Pfad in das Terminal-Fenster eintippen. Drücken Sie auf die `[↵]`-Taste. Das Tool wird ausgeführt, zeigt `Hello, World!` an und kehrt zur Befehlseingabe zurück:

```
xcodeuser$ /Users/xcodeuser/HelloWorld/build/Release/HelloWorld
Hello, World!
xcodeuser$
```

An diesem Punkt sind wir mit dem HelloWorld-Projekt fertig. Sie können es mit sämtlichen Dateien in den Papierkorb ziehen. In Xcode bleibt kein Indiz zurück, dass Sie HelloWorld erstellt oder ausgeführt haben.

1.4 Zusammenfassung

In diesem Kapitel haben Sie die ersten Grundlagen der Ausführung von Xcode kennen gelernt und ein einfaches Programm mit Xcode erstellt und ausgeführt.

Sie haben Xcode ausgeführt und für Ihre Zwecke angepasst. Dann haben Sie ein neues Projekt erstellt, um ein Command-Line-Utility zu entwickeln. Sie haben das Tool erstellt und es in einem Konsolenfenster ausgeführt und überprüft.

Sie haben gelernt, wie Xcode ein Projektverzeichnis erstellt und strukturiert und wie Sie ein Projekt entsorgen können, wenn Sie es nicht mehr benötigen.