

Inhaltsverzeichnis

Einführung	17
1 Eine kurze Geschichte der Windows-Anwendungsentwicklung	29
1.1 Das Leben von Windows	29
1.1.1 Von Windows 3.1 zu 32-Bit	30
1.1.2 Windows XP und Windows Vista	31
1.1.3 Windows 7 bügelt das Vista-Fiasko aus	31
1.1.4 Der Paradigmenwechsel von Windows 8	32
1.2 Geschichte der APIs und Werkzeuge	36
1.2.1 Die Leistungsstärke von C	36
1.2.2 C++ verdrängt C	39
1.2.3 Visual Basic	42
1.2.4 Delphi	43
1.2.5 Die Einführung von .NET	43
1.2.6 Neue UI-Technologien	45
1.3 Fallstricke der Windows-Anwendungsentwicklung	48
1.4 Zusammenfassung	50
2 Mit Windows 8 arbeiten	53
2.1 Zwei Welten, ein Betriebssystem	53
2.2 Eingabemethoden	57
2.2.1 Multitouch-Eingabe (Mehrfingereingabe)	57
2.2.2 Die Software-Tastatur	59
2.2.3 Andere Eingabegeräte	60
2.3 Anmeldung	61
2.4 Die Startseite	62
2.4.1 Die Entwicklung der Startseite	63
2.4.2 In installierten Apps browsen und Apps suchen	66

| Inhaltsverzeichnis

2.4.3	Mit Live-Kacheln arbeiten	71
2.4.4	Mit Windows-8-Style-Apps arbeiten	79
2.5	Die Windows-Charms-Leiste	84
2.5.1	Die Charms-Leiste	85
2.5.2	Der Charm »Start«	86
2.5.3	Der Charm »Suchen«	86
2.5.4	Der Charm »Teilen«	87
2.5.5	Der Charm »Geräte«	89
2.5.6	Der Charm »Einstellungen«	90
2.6	Der Windows-Desktop	92
2.6.1	Die Desktop-App	92
2.6.2	Zwischen Desktop-Programmen wechseln	93
2.6.3	Wo ist die Start-Schaltfläche?	93
2.7	Zusammenfassung	94
3	Die Windows-8-Architektur aus Entwicklersicht	97
3.1	Windows-8-Entwicklungsarchitektur	98
3.1.1	Desktopanwendungsschichten	100
3.1.2	Windows-8-Style-Anwendungsschichten	102
3.2	Windows Runtime verstehen	105
3.2.1	Überblick über die Windows-Runtime-Architektur	106
3.2.2	Metadaten in der Windows-Runtime	109
3.2.3	Language Projections	116
3.2.4	Vorteile der Windows-Runtime	118
3.2.5	Was die Windows-Runtime nicht enthält	119
3.3	.NET Framework 4.5	121
3.3.1	Das Installationsmodell von .NET Framework 4.5	121
3.3.2	Windows-Runtime-Integration	122
3.3.3	Unterstützung der asynchronen Programmierung	123
3.3.4	Andere neue Funktionen	124
3.4	Die passende Technologie für ein Projekt auswählen	124
3.4.1	Der Windows Store	124
3.4.2	Windows-8-Style-Anwendungen oder Desktopanwendungen?	125
3.4.3	Eine Programmiersprache wählen	126
3.5	Zusammenfassung	127
4	Einführung in die Entwicklungsumgebung	131
4.1	Einführung in den Werkzeugsatz	132
4.1.1	Visual Studio 2012	132

4.1.2	Visual Studio Express 2012 für Windows 8 installieren	135
4.2	Ein kurzer Überblick über die Visual-Studio-IDE	137
4.2.1	Ein neues Projekt erstellen	137
4.2.2	Eine Seite entwerfen	139
4.2.3	Ereignisse verarbeiten	142
4.2.4	Die Anwendung ausführen	145
4.2.5	Mit Beispielen und Erweiterungen arbeiten	148
4.2.6	Erweiterungen installieren und nutzen	150
4.2.7	Einige nützliche Fakten über die IDE, die Sie kennen sollten ..	153
4.3	Anwendungen mit Expression Blend aufpeppen	156
4.3.1	Expression Blend mit einer Visual-Studio-Lösung starten . .	157
4.3.2	Werkzeugfenster in Blend	158
4.3.3	Ein animiertes Objekt zur UI hinzufügen	159
4.3.4	Ein UI-Element animieren.	161
4.3.5	Die Animation starten	164
4.3.6	Visual Studio und Expression Blend zusammen verwenden . .	165
4.4	Zusammenfassung	165
5	Prinzipien der modernen Windows-Anwendungsentwicklung	171
5.1	Windows-8-Style-Anwendungen	171
5.1.1	Was ist die Windows-8-Entwurfssprache?	172
5.1.2	Allgemeine Designprinzipien für Windows-8-Anwendungen..	173.
5.1.3	Anwendungsstruktur und Navigationsmodelle	175
5.1.4	Eine einfache Windows-8-Style-Anwendung	179
5.2	Die Entwicklungsplattform für asynchrone Programme	182
5.2.1	Einführung in die asynchrone Programmierung	183
5.2.2	Die Evolution der asynchronen Programmierung auf der .NET-Plattform	186
5.2.3	Asynchrone Programmierung mit C# 5.0	191
5.2.4	Asynchrone Callbacks	197
5.2.5	Die neuen asynchronen Schlüsselwörter in C#	200
5.2.6	Langlaufende asynchrone Operationen in C# 5.0 abbrechen ..	202
5.2.7	Asynchrone Entwicklung mit der Windows-Runtime	206
5.2.8	Unterschiede zwischen der Windows-Runtime und .NET . . .	210
5.2.9	Asynchrone Programmierung mit JavaScript Promises	211
5.2.10	Mit Promises arbeiten	217
5.2.11	Promises als Rückgabewerte	218
5.2.12	Promises verketten.	218
5.3	Zusammenfassung	222

| Inhaltsverzeichnis

6	Windows-8-Style-Anwendungen mit HTML5, CSS und JavaScript erstellen	225
6.1	HTML5 und CSS im Web	226
6.1.1	HTML5-Technologien	227
6.1.2	Erste Schritte mit HTML	228
6.1.3	Eine Webseite mit HTML5 erstellen	230
6.1.4	Den Seitenstil mit CSS definieren	234
6.1.5	Erste Schritte mit CSS	235
6.1.6	Eine Webseite mit CSS-Funktionen formatieren	241
6.1.7	Clientseitigen Code ausführen	244
6.1.8	Erste Schritte mit JavaScript	245
6.1.9	Clientseitigen Code zu einer Webseite hinzufügen	248
6.2	HTML5-Anwendungen mit Windows-Runtime	252
6.2.1	Neu in Windows 8	253
6.2.2	Die Windows Library for JavaScript (WinJS)	253
6.3	Windows-8-Style-Anwendungen mit JavaScript erstellen	254
6.3.1	Auf das Dateisystem zugreifen	255
6.3.2	Daten verwalten	261
6.3.3	Eine Einfachbindung in einer HTML5-Anwendung	262
6.3.4	Die Listenbindung in einer HTML5-Anwendung benutzen	265
6.3.5	Die Datenquelle mit einer Listenbindung aktualisieren	269
6.3.6	Das Gerät des Benutzers respektieren	272
6.3.7	Ein flexibles Layout erstellen	273
6.3.8	Scrollen, zoomen und schwenken	282
6.3.9	Zoomen und schwenken implementieren	283
6.3.10	Scrollen implementieren	285
6.3.11	Canvas-Grafiken in Windows-8-Style-Anwendungen	287
6.3.12	Canvas-Grafiken in Windows-8-Style-Anwendungen zeichnen	288
6.3.13	Mit der Windows-8-Animationsbibliothek arbeiten	293
6.4	Zusammenfassung	300
7	Mit XAML Windows-8-Style-Benutzerschnittstellen erstellen	303
7.1	Die Benutzerschnittstelle mit XAML beschreiben	303
7.2	Mit Namespaces arbeiten	307
7.2.1	Die »GreetMe«-Anwendung mit XAML erstellen	307
7.3	Das Layout-Managementsystem verstehen	313
7.3.1	Ein neues Konzept: Abhängigkeitseigenschaft	313
7.3.2	Über Abhängigkeitseigenschaften hinaus: Angefügte Eigenschaften	313

7.3.3	Eigenschaften für Größe und Layout eines Steuerelements	315
7.3.4	Das Canvas-Panel	315
7.3.5	Das StackPanel-Panel	317
7.3.6	Das Grid-Panel	318
7.3.7	Ein komplexes Layout entwerfen	321
7.4	Wiederverwendbare Ressourcen in XAML	326
7.4.1	Ressourcen referenzieren	326
7.4.2	Die Hierarchie der Ressourcen	327
7.4.3	Ressourcen-Dictionarys	327
7.4.4	Mit Systemressourcen arbeiten	328
7.5	Grundlegende Steuerelemente in Windows-8-Style-Anwendungen	330
7.5.1	Steuerelemente mit einfachen Werten	331
7.5.2	Content-Steuerelemente	335
7.5.3	Die UI mit grundlegenden Steuerelementen erstellen	337
7.6	Mit Daten arbeiten	339
7.6.1	Abhängigkeitseigenschaften und Benachrichtigungen der Datenbindung	340
7.6.2	Bindungsmodi und -richtungen	342
7.6.3	Die DataContext-Eigenschaft	343
7.6.4	Datenwerte in der Bindungspipeline mit konvertieren	343
7.6.5	Bindungen an Auflistungen	345
7.6.6	Ein einfaches Master-Details-Szenario	346
7.7	Zusammenfassung	350
8	Mit XAML-Steuerelementen arbeiten	353
8.1	Animationen in Anwendungen verwenden	353
8.1.1	Animationsbibliothek	354
8.1.2	Der visuelle Status von Animationen	358
8.1.3	Mit der Animationsbibliothek arbeiten	360
8.1.4	Anwendungsspezifische Animationen	364
8.2	Das Aussehen eines Steuerelements entwerfen	371
8.2.1	Das Steuerelement mit dem Inneren verbinden	372
8.2.2	Auf Interaktionen reagieren	373
8.2.3	Mit Expression Blend arbeiten	377
8.2.4	Vorlagen mit Expression Blend anpassen	382
8.3	Mit komplexen Steuerelementen arbeiten	387
8.3.1	Die ListViewBase-Steuerelemente	387
8.3.2	Das GridView-Steuerelement	387
8.3.3	Das ListView-Steuerelement	391

| Inhaltsverzeichnis

8.3.4	Das FlipView-Steuerelement	393
8.3.5	Mit semantischem Zoom arbeiten	395
8.3.6	Komplexe Steuerelemente in Windows 8 verwenden	396
8.3.7	Mit dem AppBar-Steuerelement arbeiten	399
8.4	Zusammenfassung	400
9	Windows-8-Style-Anwendungen erstellen	403
9.1	Der Lebenszyklus einer Windows-8-Anwendung	403
9.1.1	Lebenszykluszustände einer Anwendung	404
9.1.2	Anwendungszustandsänderungen verwalten	405
9.1.3	Anwendungen anhalten, fortsetzen und schließen	407
9.1.4	Mit Lebenszyklusereignissen von Anwendungen arbeiten	408
9.2	Windows-8-Apps bereitstellen	415
9.2.1	Anwendungspakete	415
9.2.2	Ein Anwendungspaket erstellen	416
9.2.3	Das Anwendungspaketmanifest	418
9.2.4	Installation, Update und Deinstallation	421
9.3	Befehlsoberflächen	422
9.3.1	Mit dem Kontextmenü arbeiten	423
9.3.2	Mit der App-Leiste arbeiten	426
9.3.3	Ereignisse von Schaltflächen mit der App-Leiste integrieren	431
9.3.4	Das Verhalten der App-Leiste optimieren	432
9.3.5	Mit Benachrichtigungen arbeiten	434
9.3.6	Befehle mit dem Einstellungen-Charm integrieren	438
9.4	Anwendungsdaten persistieren	440
9.4.1	Anwendungsdatenspeicher	441
9.4.2	Die ApplicationData-Klasse	442
9.5	Anwendungen und die Startseite	446
9.5.1	Anwendungslogo und der Begrüßungsbildschirm	446
9.5.2	Ein breites Logo zu einer Anwendung hinzufügen	447
9.5.3	App-Kacheln mit Benachrichtigungen animieren	449
9.6	Zusammenfassung	454
10	Mehrseitige Anwendungen erstellen	457
10.1	Navigationsmuster	457
10.1.1	Hubnavigation	458
10.1.2	Direkte Navigation	459
10.1.3	Hierarchische Navigation	459
10.1.4	Semantischer Zoom	460

10.2	Mit Seiten arbeiten	461
10.2.1	Rückwärts und vorwärts navigieren	463
10.2.2	Parameter und Navigationsereignisse	467
10.2.3	App-Leisten zur Navigation verwenden	470
10.2.4	Dateien und Webseiten starten	477
10.3	Die Anwendungsvorlagen »Geteilte App« und »Raster-App«	480
10.3.1	Die Struktur der Vorlagen	482
10.3.2	Beispieldaten und Laufzeitdaten verwalten	486
10.3.3	Layout-Management	488
10.3.4	Andere studierenswerte Funktionen	491
10.4	Zusammenfassung	492
11	Verbundene Anwendungen erstellen	495
11.1	Apps mit dem Betriebssystem und anderen Apps verbinden	495
11.1.1	Picker: einheitliches Design der Dateiauswahl	497
11.1.2	Mit einem Picker arbeiten	501
11.1.3	Was sind Verträge?	505
11.1.4	Mit dem »Vertrag für »Suche«« arbeiten	509
11.1.5	Mit einem Zielfreigabe-Vertrag arbeiten	514
11.2	Auf das Internet zugreifen	519
11.2.1	Änderungen der Internet-Connectivity entdecken	519
11.2.2	Netzwerkinformationen abrufen	519
11.2.3	Mit Feeds arbeiten	522
11.2.4	Mit Feeds arbeiten	524
11.2.5	Andere SyndicationClient-Aspekte	527
11.3	Auf Windows Live zugreifen	528
11.3.1	Werkzeuge und Berechtigungen	529
11.3.2	Eine App in Windows Live Services integrieren	530
11.4	Zusammenfassung	539
12	Tablet-Funktionen nutzen	541
12.1	Anpassung an Tablet-Geräte	542
12.2	Standortabhängige Anwendungen erstellen	543
12.2.1	Mit dem Standort arbeiten	544
12.2.2	Mit der Standort-API arbeiten	546
12.3	Mit Sensoren arbeiten	553
12.3.1	Mit dem Beschleunigungsmesser arbeiten	554
12.3.2	Gyroskopwerte auslesen und verarbeiten	564
12.3.3	Mit Sensor-Fusion-Daten arbeiten	574

| Inhaltsverzeichnis

12.4	Zusammenfassung	579
13	Windows-8-Style-Anwendungen mit C++ erstellen.	583
13.1	Microsoft und die Sprache C++	584
13.1.1	Warum C++?	584
13.1.2	Sauber und sicher	585
13.2	C++ und Windows-8-Apps	589
13.2.1	Privilegien von C++ in Windows-8-Apps	590
13.2.2	Die Zusammenarbeit von Windows-Runtime und C++	591
13.2.3	Windows-Runtime-Objekte in C++ verwalten	592
13.2.4	Laufzeitklassen definieren	594
13.2.5	Ausnahmen	596
13.3	C++-Funktionen in Visual Studio entdecken	599
13.3.1	C++-Projekte erstellen	600
13.3.2	Elemente eines C++-Projekts untersuchen	601
13.3.3	Mit dem Typ Platform::String arbeiten	604
13.3.4	Mit Runtime-Auflistungen arbeiten	605
13.3.5	Mit asynchronen Operationen arbeiten	608
13.3.6	Mit Accelerated Massive Parallelism arbeiten	610
13.4	Zusammenfassung	614
14	Fortgeschrittene Programmierkonzepte	619
14.1	Lösungen mit mehreren Sprachen erstellen.	620
14.1.1	Hybridlösungen	620
14.1.2	Eine Hybridlösung mit C#- und C++-Projekten erstellen	622
14.1.3	Windows-Runtime-Komponenten erstellen und nutzen	625
14.2	Hintergrundaufgaben.	629
14.2.1	Hintergrundaufgaben verstehen	631
14.2.2	Hintergrundaufgaben implementieren	637
14.3	Eingabegeräte	648
14.3.1	Die Funktionen von Eingabegeräten abfragen	648
14.3.2	Tastaturfunktionen	649
14.3.3	Mausfunktionen	649
14.3.4	Fingereingabegerätfunktionen	650
14.3.5	Zeigegerätinformationen abfragen	651
14.4	Zusammenfassung	653
15	Windows-8-Anwendungen testen und debuggen	657
15.1	Software-Qualität	657

15.2	Mit dem Debugging vertraut werden	658
15.2.1	Den Programmablauf im Unterbrechungsmodus kontrollieren	660
15.2.2	Variablen überwachen und bearbeiten	661
15.2.3	Code beim Debugging ändern	665
15.2.4	Bestimmte Szenarien für Windows-8-Style-Anwendungen	666
15.3	Einführung in das Software-Testing	669
15.3.1	Einführung in das Unit-Testing	669
15.3.2	Unit-Testing von Windows-8-Style-Anwendungen	670
15.4	Zusammenfassung	674
16	Einführung in den Windows Store	677
16.1	Den Windows Store kennen lernen	677
16.1.1	Wie Kunden eine App im Windows Store finden	678
16.1.2	Anwendungsdetails	679
16.1.3	Mit Apps Geld verdienen	680
16.1.4	Eine Testphase und In-App-Einkäufe zu einer App hinzufügen	686
16.2	Sich als Entwickler registrieren	690
16.2.1	Die Anwendung übermitteln	691
16.2.2	Die Anwendungszertifizierung	692
16.2.3	Das Windows-App-Zertifizierungskit	693
16.3	Zusammenfassung	694
	Anhang A: Nützliche Links	699
	Stichwortverzeichnis	703