

## Auf einen Blick

### TEIL I WPF-Grundlagen und -Konzepte

1	Einführung in die WPF .....	43
2	Das Programmiermodell .....	77
3	XAML .....	143
4	Der Logical und der Visual Tree .....	193
5	Controls .....	243
6	Layout .....	323
7	Dependency Properties .....	403
8	Routed Events .....	445
9	Commands .....	487

### TEIL II Fortgeschrittene Techniken

10	Ressourcen .....	535
11	Styles, Trigger und Templates .....	573
12	Daten .....	647

### TEIL III Reichhaltige Medien und eigene Controls

13	2D-Grafik .....	791
14	3D-Grafik .....	869
15	Animationen .....	915
16	Audio und Video .....	981
17	Eigene Controls .....	1003
18	Text und Dokumente .....	1059

### TEIL IV Interoperabilität und Apps

19	Standard-Dialoge, Windows Taskbar und mehr .....	1123
20	Interoperabilität .....	1143
21	Universal Windows Apps und WinUI .....	1177

# Inhalt

Materialien zum Buch .....	25
Vorwort .....	27
Hinweise zum Buch .....	31

## TEIL I WPF-Grundlagen und -Konzepte

<b>1 Einführung in die WPF</b> .....	<b>43</b>
<b>1.1 Die WPF und .NET</b> .....	<b>43</b>
1.1.1 .NET Core 3.0 und .NET Framework 4.8 .....	43
1.1.2 Die WPF und Visual Studio 2019 .....	45
1.1.3 Die WPF als zukünftiges Programmiermodell .....	45
1.1.4 Stärken und Eigenschaften der WPF .....	47
1.1.5 Auf Wiedersehen, GDI+ .....	50
<b>1.2 Von Windows 1.0 zur Windows Presentation Foundation</b> .....	<b>50</b>
1.2.1 Die ersten Wrapper um die Windows-API .....	51
1.2.2 Windows Forms und GDI+ .....	51
1.2.3 Die Windows Presentation Foundation .....	52
<b>1.3 Die Architektur der WPF</b> .....	<b>53</b>
1.3.1 MilCore – die »Display Engine« .....	55
1.3.2 WindowsBase .....	57
1.3.3 PresentationCore .....	57
1.3.4 PresentationFramework .....	58
1.3.5 Vorteile und Stärken der WPF-Architektur .....	58
<b>1.4 Konzepte</b> .....	<b>59</b>
1.4.1 XAML .....	60
1.4.2 Dependency Properties .....	62
1.4.3 Routed Events .....	65
1.4.4 Commands .....	69
1.4.5 Styles und Templates .....	71
1.4.6 3D .....	73
<b>1.5 Zusammenfassung</b> .....	<b>74</b>

<b>2</b>	<b>Das Programmiermodell</b>	77
<b>2.1</b>	<b>Grundlagen der WPF</b>	78
2.1.1	Namespaces	78
2.1.2	Assemblies	79
2.1.3	Die Klassenhierarchie	79
2.1.4	Object	80
2.1.5	DispatcherObject	80
2.1.6	DependencyObject	82
2.1.7	Visual	82
2.1.8	UIElement	82
2.1.9	FrameworkElement	83
2.1.10	Control	84
2.1.11	ContentElement	85
2.1.12	FrameworkContentElement	85
2.1.13	Freezable	85
2.1.14	Visual3D	86
2.1.15	UIElement3D	87
<b>2.2</b>	<b>Projektvorlagen in Visual Studio 2019</b>	87
2.2.1	WPF-App	88
2.2.2	WPF-Browser-App (Web)	89
2.2.3	WPF-Benutzersteuerelementbibliothek	90
2.2.4	Benutzerdefinierte WPF-Steuerelementbibliothek	91
<b>2.3</b>	<b>Windows-Projekte mit Visual Studio 2019</b>	92
2.3.1	Ein Windows-Projekt mit XAML und C#	92
2.3.2	MainWindow.xaml	95
2.3.3	MainWindow.xaml.cs (Codebehind)	96
2.3.4	MainWindow.g.cs	97
2.3.5	MainWindow.baml	99
2.3.6	App.xaml	100
2.3.7	App.xaml.cs (Codebehind)	100
2.3.8	App.g.cs	101
2.3.9	Fazit aus den betrachteten Dateien	102
2.3.10	Der Buildvorgang von Dateien	103
2.3.11	Eine reine Codeanwendung (C#)	104
2.3.12	Eine reine kompilierte XAML-Anwendung	107
2.3.13	Best Practice	109

<b>2.4</b>	<b>Application, Dispatcher und Window</b>	109
2.4.1	Die Klasse »Application«	110
2.4.2	Übersicht der Application-Events	111
2.4.3	Events der Application-Klasse verwenden	111
2.4.4	Die Properties »Windows« und »MainWindow«	114
2.4.5	Das Shutdown-Verhalten Ihrer Applikation	115
2.4.6	Die Klasse »Dispatcher«	117
2.4.7	Fenster mit der Klasse »Window«	124
2.4.8	Methoden der Klasse »Window«	124
2.4.9	Die Komponenten eines Fensters	125
2.4.10	Übersicht der Properties der Klasse »Window«	127
2.4.11	Dialogspezifische Properties	133
2.4.12	Modale Dialoge	133
2.4.13	Nicht-modale Dialoge	137
2.4.14	Events der Klasse »Window«	138
<b>2.5</b>	<b>Zusammenfassung</b>	139
<b>3</b>	<b>XAML</b>	143
<b>3.1</b>	<b>&lt;Warum&gt;XAML?&lt;/Warum&gt;</b>	143
<b>3.2</b>	<b>Elemente und Attribute</b>	145
<b>3.3</b>	<b>Namespaces</b>	147
3.3.1	Der XML-Namespace der WPF	148
3.3.2	Der XML-Namespace für XAML	150
3.3.3	Über den Namespace-Alias	151
3.3.4	XAML mit eigenen CLR-Namespaces erweitern	151
<b>3.4</b>	<b>Properties in XAML setzen</b>	155
3.4.1	Die Attribut-Syntax	155
3.4.2	Die Property-Element-Syntax	156
3.4.3	Die Content-Property (Default-Property)	158
3.4.4	Die Attached-Property-Syntax	159
<b>3.5</b>	<b>Type-Converter</b>	160
3.5.1	Vordefinierte Type-Converter	161
3.5.2	Eigene Type-Converter implementieren	162
3.5.3	Type-Converter in C# verwenden	167

<b>3.6 Markup-Extensions</b> .....	169
3.6.1 Verwenden von Markup-Extensions in XAML und C# .....	169
3.6.2 XAML-Markup-Extensions .....	172
3.6.3 Markup-Extensions der WPF .....	173
<b>3.7 XAML-Spracherweiterungen</b> .....	175
<b>3.8 Collections in XAML</b> .....	181
3.8.1 Collections, die IList implementieren .....	181
3.8.2 Collections, die IDictionary implementieren .....	183
<b>3.9.XamlReader und.XamlWriter</b> .....	185
3.9.1 XAML mit.XamlReader dynamisch laden .....	186
3.9.2 Objekte mit.XamlWriter in XAML serialisieren .....	188
<b>3.10 Zusammenfassung</b> .....	189
<b>4 Der Logical und der Visual Tree</b> .....	193
<b>4.1 Zur Veranschaulichung verwendete Komponenten</b> .....	196
4.1.1 Der InfoDialog von FriendStorage .....	196
4.1.2 Die Anwendung »XAMLPadExtensionClone« .....	197
<b>4.2 Der Logical Tree</b> .....	199
4.2.1 Der Logical Tree des InfoDialogs .....	199
4.2.2 Für den Logical Tree verantwortliche Klassen .....	203
4.2.3 Die Klasse LogicalTreeHelper .....	207
4.2.4 NameScopes, FindName und FindLogicalNode .....	212
4.2.5 NameScopes – die Namensbereiche .....	212
4.2.6 INamespace, FrameworkElement und FrameworkContentElement .....	216
4.2.7 Die Methode »LogicalTreeHelper.FindLogicalNode« .....	218
<b>4.3 Der Visual Tree</b> .....	220
4.3.1 Der Visual Tree des InfoDialogs .....	220
4.3.2 Eigene Klassen im Visual Tree .....	223
4.3.3 Die Klasse »VisualTreeHelper« .....	227
4.3.4 Der Visual Tree und das Rendering .....	230
<b>4.4 Der Live Visual Tree in Visual Studio 2019</b> .....	235
4.4.1 Den Live Visual Tree beim Debuggen anzeigen .....	236
4.4.2 Elemente in der Anwendung auswählen .....	237
4.4.3 Eigenschaften zur Laufzeit ansehen und anpassen .....	239
<b>4.5 Zusammenfassung</b> .....	240

<b>5 Controls</b> .....	243
<b>5.1 Die Klasse »Control«</b> .....	246
<b>5.2 ContentControls</b> .....	248
5.2.1 Buttons .....	251
5.2.2 Der »normale« Button .....	251
5.2.3 Der RepeatButton .....	253
5.2.4 Der ToggleButton .....	254
5.2.5 Der RadioButton .....	255
5.2.6 Die CheckBox .....	257
5.2.7 Labels .....	258
5.2.8 ToolTips anzeigen .....	260
5.2.9 Die Klasse »ToolTip« .....	261
5.2.10 Die Klasse »ToolTipService« .....	263
5.2.11 Scrollen mit ScrollViewer .....	264
5.2.12 WPF- und HTML-Inhalte mit Frame darstellen .....	266
5.2.13 ContentControls mit Header .....	268
5.2.14 Inhalte mit dem Expander auf- und zuklappen .....	268
5.2.15 Inhalte mit der GroupBox gruppieren .....	270
5.2.16 TabItems für ein TabControl .....	270
<b>5.3 ItemsControls</b> .....	271
5.3.1 ItemsControls mit Header .....	274
5.3.2 Navigation mit einer ToolBar .....	274
5.3.3 TreeViewItem und MenuItem .....	278
5.3.4 Baumansicht mit der TreeView .....	278
5.3.5 Navigation über Menüs .....	282
5.3.6 Das gewöhnliche Menü .....	283
5.3.7 Das Kontextmenü .....	285
5.3.8 Elemente mit einem Selector auswählen .....	287
5.3.9 Listen mit der ComboBox anzeigen .....	288
5.3.10 Eine Liste in der ListBox anzeigen .....	291
5.3.11 Die ListView als erweiterte ListBox .....	292
5.3.12 Mehrere Tabs mit einem TabControl .....	295
5.3.13 Das Ribbon .....	295
5.3.14 Eine StatusBar mit Informationen .....	300
5.3.15 »Alternating Rows« mit AlternationCount .....	300
<b>5.4 Controls zur Textdarstellung und -bearbeitung</b> .....	302
5.4.1 »TextBox« zum Editieren von Text .....	302
5.4.2 RichTextBox für formatierten Text .....	304
5.4.3 PasswordBox für maskierten Text .....	304

5.4.4	TextBlock zur Anzeige von Text .....	305
5.4.5	Zeichnen mit dem InkCanvas .....	305
<b>5.5</b>	<b>Datum-Controls</b> .....	<b>307</b>
5.5.1	Calendar .....	307
5.5.2	DatePicker .....	310
<b>5.6</b>	<b>Range-Controls</b> .....	<b>312</b>
5.6.1	Bereich mit Slider auswählen .....	313
5.6.2	ProgressBar zur Statusanzeige .....	314
5.6.3	Scrollen mit der ScrollBar .....	314
<b>5.7</b>	<b>Sonstige, einfachere Controls</b> .....	<b>315</b>
5.7.1	Decorator zum Ausschmücken .....	315
5.7.2	Einfacher Rahmen mit der Border-Klasse .....	315
5.7.3	Elemente mit der Viewbox skalieren .....	316
5.7.4	Bilder mit der Image-Klasse darstellen .....	317
5.7.5	Einfaches Popup anzeigen .....	318
<b>5.8</b>	<b>Zusammenfassung</b> .....	<b>321</b>
<b>6</b>	<b>Layout</b> .....	<b>323</b>
<b>6.1</b>	<b>Der Layoutprozess</b> .....	<b>323</b>
6.1.1	Die zwei Schritte des Layoutprozesses .....	324
6.1.2	Schritt 1: Measure .....	324
6.1.3	Schritt 2: Arrange .....	325
6.1.4	Fazit aus den beiden Schritten .....	325
6.1.5	»MeasureOverride« und »ArrangeOverride« .....	326
6.1.6	Ein eigenes Layout-Panel (DiagonalPanel) .....	327
6.1.7	Zusammenfassung des Layoutprozesses .....	332
<b>6.2</b>	<b>Layoutfunktionalität von Elementen</b> .....	<b>333</b>
6.2.1	Width und Height .....	333
6.2.2	Margin und Padding .....	334
6.2.3	Margin .....	334
6.2.4	Padding .....	336
6.2.5	Alignments .....	337
6.2.6	»HorizontalAlignment« und »VerticalAlignment« .....	337
6.2.7	»HorizontalAlignment« und »VerticalContentAlignment« .....	338
6.2.8	Die Visibility-Property .....	339
6.2.9	Die UseLayoutRounding-Property .....	339
6.2.10	Transformationen .....	341

6.2.11	»LayoutTransform« vs. »RenderTransform« .....	343
6.2.12	»RotateTransform« .....	346
6.2.13	»ScaleTransform« .....	347
6.2.14	»SkewTransform« .....	349
6.2.15	»TranslateTransform« .....	349
6.2.16	»MatrixTransform« .....	350
6.2.17	»TransformGroup« .....	352
<b>6.3</b>	<b>Panels</b> .....	<b>354</b>
6.3.1	Die Klasse »Panel« .....	354
6.3.2	Canvas .....	355
6.3.3	StackPanel .....	358
6.3.4	WrapPanel .....	360
6.3.5	DockPanel .....	361
6.3.6	Grid .....	363
6.3.7	Die Höhe von Zeilen und die Breite von Spalten .....	365
6.3.8	GridSplitter .....	368
6.3.9	Geteilte Größen – »SharedSizeGroup« .....	372
6.3.10	Das Grid als Alleskönner .....	375
6.3.11	Primitive Panels .....	377
6.3.12	»UniformGrid« .....	377
6.3.13	»VirtualizingPanels« .....	378
6.3.14	TabPanel .....	380
6.3.15	ToolBarOverflowPanel .....	380
6.3.16	Übersicht der Alignments in den verschiedenen Panels .....	380
6.3.17	Wenn der Platz im Panel nicht ausreicht .....	383
<b>6.4</b>	<b>Das Layout von FriendStorage</b> .....	<b>385</b>
6.4.1	Das Hauptfenster aus Benutzersicht .....	385
6.4.2	Das Hauptfenster aus Entwicklersicht .....	387
6.4.3	Animation des Freunde-Explorers .....	396
<b>6.5</b>	<b>Zusammenfassung</b> .....	<b>401</b>
<b>7</b>	<b>Dependency Properties</b> .....	<b>403</b>
<b>7.1</b>	<b>Die Keyplayer</b> .....	<b>404</b>
7.1.1	»DependencyObject« und »DependencyProperty« .....	404
7.1.2	Was ist die Property Engine? .....	406
<b>7.2</b>	<b>Dependency Properties</b> .....	<b>406</b>
7.2.1	Eine Dependency Property implementieren .....	408

7.2.2	Klassische FontSize-Property .....	408
7.2.3	»FontSize« als Dependency Property .....	408
7.2.4	Metadaten einer Dependency Property .....	411
7.2.5	Die Callback-Methoden der Metadaten .....	415
7.2.6	Metadaten in Subklassen überschreiben .....	416
7.2.7	Default-Metadaten .....	417
7.2.8	Validieren einer Dependency Property .....	417
7.2.9	Die FontSize-Property als Ziel eines Data Bindings .....	420
7.2.10	Existierende Dependency Properties verwenden .....	422
7.2.11	Read-only-Dependency-Properties implementieren .....	425
7.2.12	Den Wert einer Dependency Property ermitteln .....	427
7.2.13	Lokal gesetzte Werte löschen .....	429
7.2.14	Überblick über die Quellen mit »DependencyPropertyHelper« .....	430
7.2.15	Auf Änderungen in existierenden Klassen lauschen .....	430
<b>7.3</b>	<b>Attached Properties</b> .....	<b>431</b>
7.3.1	Eine Attached Property implementieren .....	432
7.3.2	Ein einfaches Panel mit Attached Properties .....	435
7.3.3	Bekannte Vertreter .....	439
<b>7.4</b>	<b>Zusammenfassung</b> .....	<b>441</b>
<b>8</b>	<b>Routed Events</b> .....	<b>445</b>
<b>8.1</b>	<b>Die Keyplayer</b> .....	<b>446</b>
8.1.1	Die Klassen »RoutedEvent« und »EventManager« .....	446
8.1.2	Die Routing-Strategie .....	447
8.1.3	Das Interface »InputElement« .....	450
8.1.4	Die Klasse »RoutedEventArgs« .....	451
8.1.5	Das Event System .....	452
<b>8.2</b>	<b>Eigene Routed Events</b> .....	<b>453</b>
8.2.1	Ein Routed Event implementieren .....	453
8.2.2	Klassisches Click-Event .....	453
8.2.3	Click-Event als Routed Event .....	454
8.2.4	Das Routed Event als Attached Event verwenden .....	458
8.2.5	Existierende Routed Events in eigenen Klassen nutzen .....	460
8.2.6	Instanz- und Klassenbehandlung .....	461
<b>8.3</b>	<b>Die »RoutedEventArgs« im Detail</b> .....	<b>466</b>
8.3.1	Sender vs. Source und OriginalSource .....	466
8.3.2	Die Handled-Property .....	469

<b>8.4</b>	<b>Routed Events der WPF</b> .....	<b>471</b>
8.4.1	Tastatur-Events .....	473
8.4.2	Maus-Events .....	476
8.4.3	Mouse-Capturing .....	478
8.4.4	Stylus-Events (Stift) .....	480
8.4.5	Multitouch-Events .....	481
8.4.6	Die statischen Mitglieder eines FrameworkElements .....	484
<b>8.5</b>	<b>Zusammenfassung</b> .....	<b>485</b>
<b>9</b>	<b>Commands</b> .....	<b>487</b>
<b>9.1</b>	<b>Die Keyplayer</b> .....	<b>488</b>
9.1.1	Das Interface »ICommand« .....	488
9.1.2	Das Interface »ICommandSource« .....	489
<b>9.2</b>	<b>Eigene Commands mit »ICommand«</b> .....	<b>490</b>
9.2.1	Ein Command implementieren .....	490
9.2.2	Das Command verwenden .....	491
9.2.3	Das Command von der Logik entkoppeln .....	492
<b>9.3</b>	<b>Die »wahren« Keyplayer</b> .....	<b>495</b>
9.3.1	Die Klassen »RoutedCommand« und »RoutedUICommand« .....	495
9.3.2	Der »CommandManager« .....	497
9.3.3	Die Klasse »CommandBinding« .....	498
9.3.4	Elemente mit einer CommandBindings-Property .....	499
9.3.5	Das Zusammenspiel der Keyplayer .....	500
<b>9.4</b>	<b>Eigene Commands mit der Klasse »RoutedUICommand«</b> .....	<b>504</b>
9.4.1	Die eigenen Commands in FriendStorage .....	504
9.4.2	Commands mit »InputGestures« versehen .....	505
9.4.3	Die Klasse »KeyGesture« .....	505
9.4.4	Die Klasse »MouseGesture« .....	506
9.4.5	Die FriendCommands-Klasse mit InputGestures erweitern .....	506
9.4.6	CommandBindings zum Window-Objekt hinzufügen .....	507
9.4.7	Die Commands im Menü und in der ToolBar verwenden .....	509
<b>9.5</b>	<b>Built-in-Commands der WPF</b> .....	<b>514</b>
9.5.1	Built-in-Commands in FriendStorage .....	515
9.5.2	Bestehende Commands mit InputBindings auslösen .....	517
9.5.3	Controls mit integrierten CommandBindings .....	520

<b>9.6</b>	<b>Das Model-View-ViewModel-Pattern (MVVM)</b> .....	523
9.6.1	Die Idee des Model-View-Controller-Patterns (MVC) .....	523
9.6.2	Die Idee des Model-View-ViewModel-Patterns (MVVM) .....	525
9.6.3	Ein MVVM-Beispiel .....	526
<b>9.7</b>	<b>Zusammenfassung</b> .....	531

## TEIL II Fortgeschrittene Techniken

### 10 Ressourcen 535

<b>10.1</b>	<b>Logische Ressourcen</b> .....	535
10.1.1	Logische Ressourcen definieren und verwenden .....	536
10.1.2	Die Suche nach Ressourcen im Detail .....	539
10.1.3	Die Suche im Application-Objekt .....	540
10.1.4	Systemweite Ressourcen .....	540
10.1.5	Elemente als Ressourcen verwenden .....	543
10.1.6	Statische Ressourcen .....	546
10.1.7	Dynamische Ressourcen .....	548
10.1.8	Ressourcen in separate Dateien auslagern .....	552
10.1.9	Logische Ressourcen in FriendStorage .....	555
<b>10.2</b>	<b>Binäre Ressourcen</b> .....	558
10.2.1	Binäre Ressourcen in .NET .....	558
10.2.2	Binäre Ressourcen bei der WPF .....	560
10.2.3	Die Pack-URI-Syntax .....	563
10.2.4	Auf Dateien im Anwendungsverzeichnis zugreifen .....	564
10.2.5	In C# auf binäre Ressourcen zugreifen .....	565
10.2.6	Eine binäre Ressource als SplashScreen .....	568
<b>10.3</b>	<b>Zusammenfassung</b> .....	571

### 11 Styles, Trigger und Templates 573

<b>11.1</b>	<b>Styles</b> .....	573
11.1.1	Grundlagen und Keyplayer .....	574
11.1.2	Styles als logische Ressourcen definieren .....	577
11.1.3	Einen Style für verschiedene Typen verwenden .....	581
11.1.4	Bestehende Styles erweitern .....	583

11.1.5	Setter und EventSetter .....	585
11.1.6	Styles und Trigger .....	588
<b>11.2</b>	<b>Trigger</b> .....	588
11.2.1	Property-Trigger .....	590
11.2.2	Die Reihenfolge der Trigger .....	592
11.2.3	»EnterActions« und »ExitActions« aus »TriggerBase« .....	593
11.2.4	DataTrigger .....	595
11.2.5	EventTrigger .....	596
11.2.6	»EventTrigger« und »FrameworkElement« .....	598
11.2.7	Komplexe Bedingungen mit Triggern .....	601
11.2.8	Logisches Oder .....	601
11.2.9	Logisches Und .....	602
<b>11.3</b>	<b>Templates</b> .....	603
11.3.1	Arten von Templates .....	603
11.3.2	Layout mit dem ItemsPanelTemplate .....	605
11.3.3	Daten mit DataTemplates visualisieren .....	606
11.3.4	Das Aussehen von Controls mit ControlTemplates anpassen .....	610
11.3.5	Das Default-ControlTemplate eines Controls .....	613
11.3.6	Die Verbindung zwischen Control und Template .....	617
11.3.7	Der Inhalt von ContentControls .....	620
11.3.8	Die Items von ItemsControls .....	620
11.3.9	Two-Way-Contract zwischen Control und Template .....	621
11.3.10	VisualStateManager statt Trigger verwenden .....	625
11.3.11	Ein Template mit Visual States implementieren .....	627
11.3.12	Zustandsübergänge mit Transitions definieren .....	631
11.3.13	Das komplette Template im Überblick .....	633
11.3.14	Templates in C# .....	636
<b>11.4</b>	<b>Styles, Trigger und Templates in FriendStorage</b> .....	637
11.4.1	Der Next-Button .....	638
11.4.2	Die Image-Objekte der Toolbar-Buttons .....	640
11.4.3	Die DataGridRows des Freunde-Explorers .....	642
<b>11.5</b>	<b>Zusammenfassung</b> .....	645

### 12 Daten 647

<b>12.1</b>	<b>Data Binding</b> .....	648
12.1.1	Data Binding in XAML .....	648
12.1.2	Data Binding in C# .....	649

12.1.3	Die Binding-Klasse im Detail .....	650
12.1.4	Der DataContext .....	653
12.1.5	Die Path-Property im Detail .....	656
12.1.6	Die Richtung des Bindings .....	657
12.1.7	Der UpdateSourceTrigger .....	659
12.1.8	Die Delay-Property des Bindings .....	660
12.1.9	Die BindingExpression .....	661
12.1.10	Bindings entfernen .....	663
12.1.11	Debugging von Data Bindings .....	663
<b>12.2</b>	<b>Datenquellen eines Data Bindings .....</b>	<b>666</b>
12.2.1	Binding an die Dependency Properties eines Elements .....	666
12.2.2	Binding an einfache .NET-Properties .....	667
12.2.3	»NotifyPropertyChanged« mit dem CallerMemberName-Attribut .....	669
12.2.4	Binding an statische Properties .....	670
12.2.5	Binding an logische Ressourcen .....	672
12.2.6	Binding an Quellen unterschiedlichen Typs .....	673
12.2.7	Binding an relative Quellen mit »RelativeSource« .....	676
12.2.8	Mit dem Mode »Self« an eine Property des Target-Elements binden .....	676
12.2.9	»TemplatedParent« für Templates .....	677
12.2.10	Mit »FindAncestor« an im Element Tree höher liegende Elemente binden ....	678
12.2.11	Mit »PreviousData« an das vorhergehende Datenobjekt binden .....	679
12.2.12	Binding der Target-Property an mehrere Quellen .....	679
12.2.13	Die Klasse »MultiBinding« .....	679
12.2.14	Die Klasse »PriorityBinding« .....	682
12.2.15	DataSourceProvider für Objekte und XML .....	683
12.2.16	Die Subklasse »ObjectDataProvider« .....	684
12.2.17	Die Subklasse »XmlDataProvider« .....	686
12.2.18	Binding an X.Linq .....	690
<b>12.3</b>	<b>Data Binding an Collections .....</b>	<b>692</b>
12.3.1	Der Fallback-Mechanismus .....	692
12.3.2	Die CollectionViews der WPF .....	695
12.3.3	Das Interface »ICollectionView« .....	696
12.3.4	Klassen, die »ICollectionView« implementieren .....	698
12.3.5	Die DefaultView .....	700
12.3.6	Daten filtern, sortieren und gruppieren .....	701
12.3.7	Das Filtern, Sortieren und Gruppieren in C# .....	702
12.3.8	Das Filtern, Sortieren und Gruppieren in XAML .....	706
12.3.9	»Live Shaping« von Daten .....	707
12.3.10	Hinzufügen und Löschen von Daten .....	709
12.3.11	Collections auf Worker-Threads bearbeiten .....	710

12.3.12	Mehrere Collections als Datenquelle verwenden .....	712
12.3.13	Binding an ein ADO.NET-DataSet .....	712
<b>12.4</b>	<b>Benutzereingaben validieren .....</b>	<b>716</b>
12.4.1	Validieren mit »ExceptionValidationRule« .....	718
12.4.2	Validieren mit »DataErrorValidationRule« .....	720
12.4.3	Validieren mit »NotifyDataErrorValidationRule« .....	721
12.4.4	Validieren mit eigener ValidationRule .....	725
12.4.5	Die Validation-Klasse .....	726
12.4.6	Validieren mehrerer Bindings mit »BindingGroup« .....	728
12.4.7	Validieren mit der BindingGroup .....	730
12.4.8	Transaktionen mit der BindingGroup .....	735
<b>12.5</b>	<b>Das DataGridView .....</b>	<b>737</b>
12.5.1	Die verwendeten Testdaten .....	738
12.5.2	Autogenerieren von Columns .....	740
12.5.3	Unterschiedliche Column-Typen .....	742
12.5.4	Columns manuell zum DataGridView hinzufügen .....	745
12.5.5	Die Breite einer Column .....	747
12.5.6	Columns mit der Klasse »DataGridTemplateColumn« .....	748
12.5.7	RowDetails anzeigen .....	750
12.5.8	Daten gruppieren .....	752
12.5.9	Die Auswahlmöglichkeiten festlegen .....	754
12.5.10	Auf ausgewählte Daten zugreifen .....	755
12.5.11	Daten bearbeiten .....	757
12.5.12	Daten im DataGridView validieren .....	757
12.5.13	Sonstige Eigenschaften des DataGrids .....	761
<b>12.6</b>	<b>Daten mit DataTemplates visualisieren .....</b>	<b>762</b>
12.6.1	Auswahl mit der Klasse »DataTemplateSelector« .....	762
12.6.2	Hierarchische DataTemplates .....	765
<b>12.7</b>	<b>Drag-&amp;-Drop .....</b>	<b>767</b>
<b>12.8</b>	<b>Daten in FriendStorage .....</b>	<b>771</b>
12.8.1	Die Entitäten »Friend«, »Address« und »FriendCollection« .....	772
12.8.2	Daten im MainWindow .....	772
12.8.3	Der Freunde-Explorer .....	774
12.8.4	Die Detailansicht .....	775
12.8.5	Previous/Next-Buttons zur Navigation .....	776
12.8.6	Data Binding mit Fallback-Mechanismus .....	777
12.8.7	Validierung des Vornamens .....	778
12.8.8	Droppen eines Bildes .....	780
12.8.9	Die StatusBar .....	781



12.8.10	Daten im NewFriendDialog .....	782
12.8.11	Speichern in gezippter .friends-Datei .....	785
<b>12.9</b>	<b>Zusammenfassung</b> .....	<b>787</b>

## TEIL III Reichhaltige Medien und eigene Controls

### 13 2D-Grafik 791

<b>13.1</b>	<b>Shapes</b> .....	<b>792</b>
13.1.1	Das Rectangle .....	793
13.1.2	Die Ellipse .....	794
13.1.3	Linien mit »Line« und »Polyline« .....	795
13.1.4	Spezielle Formen mit »Polygon« .....	797
13.1.5	Ein Außerirdischer aus Shapes .....	799
13.1.6	Die StrokeXXX-Properties der Shape-Klasse .....	801
13.1.7	Komplexe Shapes mit »Path« .....	804
<b>13.2</b>	<b>Geometries</b> .....	<b>804</b>
13.2.1	»RectangleGeometry« und »EllipseGeometry« .....	806
13.2.2	»LineGeometry« .....	807
13.2.3	Mehrere Geometry-Objekte gruppieren .....	807
13.2.4	Geometries kombinieren .....	808
13.2.5	Komplexe Formen mit »PathGeometry« .....	810
13.2.6	Die Klasse »StreamGeometry« .....	813
13.2.7	Die Path-Markup-Syntax .....	814
13.2.8	Clipping mit Geometry-Objekten .....	816
<b>13.3</b>	<b>Drawings</b> .....	<b>816</b>
13.3.1	»GeometryDrawing« und »DrawingGroup« .....	817
13.3.2	»ImageDrawing« und »VideoDrawing« .....	819
13.3.3	Ein Außerirdischer aus Geometries und Drawings .....	820
<b>13.4</b>	<b>Programmierung des Visual Layers</b> .....	<b>823</b>
13.4.1	Die Klasse »DrawingContext« .....	823
13.4.2	DrawingVisual einsetzen .....	825
13.4.3	Visual-Hit-Testing .....	827
<b>13.5</b>	<b>Brushes</b> .....	<b>828</b>
13.5.1	Der »SolidColorBrush« und die Color-Struktur .....	829
13.5.2	Farbverläufe mit GradientBrushes .....	831
13.5.3	TileBrushes .....	836

<b>13.6</b>	<b>Cached Compositions</b> .....	<b>841</b>
13.6.1	BitmapCache für ein Element aktivieren .....	842
13.6.2	Nebeneffekte des Cachings .....	843
13.6.3	Elemente mit »BitmapCacheBrush« zeichnen .....	846
<b>13.7</b>	<b>Effekte</b> .....	<b>848</b>
13.7.1	Die Effect-Klassen .....	848
13.7.2	»Blur« und »DropShadow« verwenden .....	849
13.7.3	Properties von »BlurEffect« und »DropShadowEffect« .....	850
13.7.4	Effekte mit eigenen Pixelshadern .....	851
13.7.5	Pixelshader mit weiteren Konstanten .....	860
<b>13.8</b>	<b>Bitmaps</b> .....	<b>863</b>
13.8.1	BitmapSources – Bildquellen .....	863
13.8.2	Bitmap-Operationen .....	865
13.8.3	Bitmap-Operationen in FriendStorage .....	865
<b>13.9</b>	<b>Zusammenfassung</b> .....	<b>866</b>
<b>14</b>	<b>3D-Grafik</b> .....	<b>869</b>
<b>14.1</b>	<b>3D im Überblick</b> .....	<b>870</b>
14.1.1	Inhalte einer 3D-Szene .....	870
14.1.2	2D und 3D im Vergleich .....	871
<b>14.2</b>	<b>Die Objekte einer 3D-Szene im Detail</b> .....	<b>873</b>
14.2.1	Das 3D-Koordinatensystem .....	873
14.2.2	Der Viewport3D als Fernseher .....	874
14.2.3	Die richtige Kamera .....	875
14.2.4	Visual3D-Objekte .....	880
14.2.5	Model3D-Objekte .....	881
14.2.6	»GeometryModel3D« aufbauen .....	882
14.2.7	Licht ins Dunkel bringen .....	888
14.2.8	Transformationen .....	891
14.2.9	Verschiedene Materialien .....	894
14.2.10	Texturen .....	895
14.2.11	Normalen .....	898
<b>14.3</b>	<b>Benutzerinteraktion mit 3D-Objekten</b> .....	<b>903</b>
14.3.1	Interaktivität in »ModelVisual3D« mit Visual-Hit-Testing .....	903
14.3.2	Interaktivität mit »UIElement3D« .....	904
14.3.3	Interaktive 2D-Elemente auf 3D-Objekten .....	906

<b>14.4 Komplexe 3D-Objekte</b> .....	908
14.4.1 Landschaft im Code generieren .....	908
14.4.2 Kugel erstellen .....	910
14.4.3 Komplexe 3D-Objekte mit Third-Party-Tools erstellen .....	912
<b>14.5 Zusammenfassung</b> .....	913
<b>15 Animationen</b> .....	915
<hr/>	
<b>15.1 Animationsgrundlagen</b> .....	916
15.1.1 Voraussetzungen für Animationen .....	916
15.1.2 Übersicht über die Animationsarten und -klassen .....	917
15.1.3 Timelines und Clocks .....	920
15.1.4 Das Interface »IAnimatable« .....	923
<b>15.2 Basis-Animationen in C#</b> .....	924
15.2.1 Start- und Zielwert mit »From«, »To« und »By« .....	925
15.2.2 Animation nur mit »To« .....	926
15.2.3 Animation nur mit »From« .....	927
15.2.4 Animation mit »From« und »By« .....	928
15.2.5 Dauer, Startzeit und Geschwindigkeit .....	929
15.2.6 Rückwärts und Wiederholen .....	931
15.2.7 Die Gesamtlänge einer Timeline .....	933
15.2.8 Wiederholen mit neuen Werten .....	933
15.2.9 Beschleunigen und Abbremsen .....	935
15.2.10 Das Füllverhalten einer Animation .....	936
15.2.11 Eine Animation mit »AnimationClock« steuern .....	938
15.2.12 Animationen in FriendStorage .....	942
<b>15.3 Basis-Animationen in XAML</b> .....	943
15.3.1 Eine einfache Animation in XAML .....	946
15.3.2 Das Storyboard als Timeline-Container .....	950
15.3.3 Animationen mit »ControllableStoryboard« steuern .....	952
<b>15.4 Keyframe-Animationen</b> .....	955
15.4.1 Lineare Keyframe-Animationen .....	956
15.4.2 SplineKeyframe-Animationen .....	958
15.4.3 Animationen mit diskreten Keyframes .....	960
<b>15.5 Pfad-Animationen</b> .....	963
<b>15.6 Easing Functions</b> .....	966
15.6.1 Grundlagen der Easing Functions .....	966

15.6.2 Easing Functions in Basis-Animationen .....	970
15.6.3 Easing Functions in Keyframe-Animationen .....	972
15.6.4 Eigene Easing Functions erstellen .....	974
<b>15.7 Low-Level-Animationen</b> .....	976
<b>15.8 Zusammenfassung</b> .....	980
<b>16 Audio und Video</b> .....	981
<hr/>	
<b>16.1 Audio (.wav) mit »SoundPlayerAction« und »SoundPlayer«</b> .....	981
16.1.1 Audio mit »SoundPlayerAction« (XAML) .....	982
16.1.2 Audio mit »SoundPlayer« (C#) .....	983
<b>16.2 Audio und Video mit »MediaPlayer« (C#)</b> .....	985
16.2.1 Einfaches Abspielen .....	985
16.2.2 Audio mit dem MediaPlayer .....	985
16.2.3 Videos abspielen .....	986
16.2.4 Steuerung mit »MediaClock« und »MediaTimeline« .....	988
<b>16.3 Audio und Video mit »MediaElement« (XAML)</b> .....	991
16.3.1 Einfaches Abspielen .....	992
16.3.2 Steuerung mit Methoden (unabhängiger Modus) .....	994
16.3.3 Steuerung mit »MediaTimeline« (Clock-Modus) .....	994
16.3.4 Storyboard mit »MediaTimeline« und »AnimationTimeline« .....	997
16.3.5 Snapshots von Videos .....	999
<b>16.4 Zusammenfassung</b> .....	1001
<b>17 Eigene Controls</b> .....	1003
<hr/>	
<b>17.1 Custom Controls</b> .....	1004
17.1.1 Die Struktur eines Custom Controls .....	1005
17.1.2 Der zu erstellende VideoPlayer .....	1008
17.1.3 Klassenname anpassen .....	1008
17.1.4 Template-Parts definieren .....	1010
17.1.5 Dependency Properties erstellen .....	1013
17.1.6 Routed Events implementieren .....	1017
17.1.7 Commands unterstützen .....	1018
17.1.8 Das Aussehen des lookless Controls festlegen .....	1022
17.1.9 Das Control testen .....	1026

17.1.10	Optional weitere Theme-Styles anlegen .....	1028
17.1.11	Templates auf Windows-Ebene definieren .....	1031
<b>17.2</b>	<b>Custom Control mit Visual States</b> .....	1035
17.2.1	Visual States im Code implementieren .....	1035
17.2.2	States für andere sichtbar machen .....	1038
17.2.3	States im Default-ControlTemplate unterstützen .....	1039
17.2.4	Den VideoPlayer mit Visual States testen .....	1041
<b>17.3</b>	<b>User Control</b> .....	1042
17.3.1	Die Struktur eines User Controls .....	1042
17.3.2	Das selbst erstellte PrintableFriend-Control .....	1044
17.3.3	Das UI des Controls definieren .....	1045
17.3.4	Properties in der Codebehind-Datei erstellen .....	1046
17.3.5	Die Content-Property festlegen .....	1048
<b>17.4</b>	<b>Alternativen zu Custom Control und User Control</b> .....	1049
17.4.1	Wann sollte man die OnRender-Methode überschreiben? .....	1049
17.4.2	Adorner erstellen und Elemente damit ausschmücken .....	1050
<b>17.5</b>	<b>Zusammenfassung</b> .....	1056

## **18 Text und Dokumente** 1059

<b>18.1</b>	<b>Text</b> .....	1060
18.1.1	»FrameworkContentElement« als Basis für Text .....	1060
18.1.2	Formatierung mit Spans .....	1062
18.1.3	Formatierung mit den Properties aus »TextElement« .....	1064
18.1.4	Elemente im Text mit »InlineUIContainer« .....	1066
18.1.5	Fonts und Typefaces .....	1067
18.1.6	Typografie .....	1068
18.1.7	Die FormattedText-Klasse .....	1069
18.1.8	Texteffekte .....	1071
18.1.9	Nützliche Eigenschaften der TextBlock-Klasse .....	1073
<b>18.2</b>	<b>Das Text-Rendering beeinflussen</b> .....	1075
18.2.1	Text klar zeichnen .....	1075
18.2.2	Text für Animationen optimieren .....	1076
18.2.3	Den Algorithmus für das Anti-Aliasing festlegen .....	1077
18.2.4	Den ClearType-Algorithmus aktivieren .....	1078
<b>18.3</b>	<b>Flow-Dokumente</b> .....	1080
18.3.1	Die Klasse »FlowDocument« .....	1080
18.3.2	Die fünf Block-Arten .....	1082

18.3.3	Die AnchoredBlocks »Figure« und »Floater« .....	1086
18.3.4	Controls zum Betrachten .....	1089
<b>18.4</b>	<b>Annotationen</b> .....	1091
<b>18.5</b>	<b>XPS-Dokumente (Fixed-Dokumente)</b> .....	1096
18.5.1	FlowDocument als XPS speichern .....	1096
18.5.2	Ein XPS-Dokument laden und anzeigen .....	1100
18.5.3	Die Inhalte eines XPS-Dokuments .....	1101
18.5.4	XPS in C# mit FixedDocument & Co. erstellen .....	1105
<b>18.6</b>	<b>Drucken</b> .....	1107
18.6.1	Einfaches Ausdrucken .....	1108
18.6.2	Drucken mit »PrintQueue« .....	1109
18.6.3	Festlegen von Druckeigenschaften mit »PrintTicket« .....	1110
18.6.4	Drucken mit »PrintDialog« .....	1111
<b>18.7</b>	<b>Dokumente in FriendStorage</b> .....	1112
18.7.1	Hilfe mit Flow-Dokument .....	1112
18.7.2	Export der Freundesliste als XPS .....	1113
18.7.3	Drucken der Freundesliste .....	1118
<b>18.8</b>	<b>Zusammenfassung</b> .....	1119

## **TEIL IV Interoperabilität und Apps**

### **19 Standard-Dialoge, Windows Taskbar und mehr** 1123

<b>19.1</b>	<b>Standard-Dialoge</b> .....	1123
19.1.1	Der SaveFileDialog .....	1124
19.1.2	Der OpenFileDialog .....	1124
<b>19.2</b>	<b>Integration in die Windows Taskbar</b> .....	1125
19.2.1	Übersicht der Möglichkeiten .....	1126
19.2.2	Thumb-Buttons im Vorschaufenster .....	1128
19.2.3	Ein Overlay-Bild auf dem Taskbar-Button .....	1131
19.2.4	Eine Fortschrittsanzeige auf dem Taskbar-Button .....	1132
19.2.5	Den Ausschnitt im Thumbnail festlegen .....	1134
19.2.6	Eine JumpList mit JumpTasks .....	1135
19.2.7	JumpList mit JumpTasks und JumpPaths .....	1136
19.2.8	JumpList mit letzten und häufigen Elementen .....	1139
<b>19.3</b>	<b>Deployment</b> .....	1140
<b>19.4</b>	<b>Zusammenfassung</b> .....	1141

<b>20 Interoperabilität</b>	1143
<b>20.1 Unterstützte Szenarien und Grenzen</b>	1144
20.1.1 Mögliche Interoperabilitätsszenarien	1144
20.1.2 Grenzen und Einschränkungen	1145
<b>20.2 Windows Forms</b>	1146
20.2.1 Windows Forms in WPF	1147
20.2.2 WPF in Windows Forms	1155
20.2.3 Dialoge	1157
<b>20.3 ActiveX in WPF</b>	1159
<b>20.4 Win32</b>	1162
20.4.1 Win32 in WPF	1162
20.4.2 Win32-Nachrichten in WPF abfangen	1173
<b>20.5 Zusammenfassung</b>	1176
<b>21 Universal Windows Apps und WinUI</b>	1177
<b>21.1 Einführung</b>	1177
21.1.1 Windows Store Apps in Windows 8	1178
21.1.2 Universal Windows Apps in Windows 10	1179
21.1.3 Das WinUI-Framework	1180
21.1.4 Universal Windows Apps/WinUI vs. WPF	1181
<b>21.2 Die FriendViewer-App erstellen</b>	1182
21.2.1 Die Tools installieren	1182
21.2.2 Das Projekt anlegen	1183
21.2.3 Projektstruktur und Dateien	1184
21.2.4 Die MainPage befüllen	1185
21.2.5 Visuelle Zustände für adaptives Layout	1190
21.2.6 Wie geht es weiter?	1194
<b>21.3 Zusammenfassung</b>	1194
Index	1195