

Vorwort

»Zwei Dinge sind zu unserer Arbeit nötig:

Unermüdliche Ausdauer und die Bereitschaft, etwas,

in das man viel Zeit und Arbeit gesteckt hat, wieder wegzuwerfen«.

Albert Einstein

Aufgrund meiner Erfahrungen hat das Wissen in unserer Berufssparte eine Halbwertszeit von ungefähr drei Jahren. Diese sehr große Zerfallsrate bedingt eine unermüdliche Ausdauer um stetes »am Ball zu bleiben«. Für Personen, die wie ich schon lange im Geschäft sind, bedeutet das eine ständige Herausforderung. Für Personen, die gerade in das Geschäft einsteigen, dauert es bis zu drei Jahren, bis sie so viel Wissen aufgebaut haben, dass sie wirklich mitreden können.

Während dieser Jahre muss auch immer wieder etwas fallen gelassen werden, damit Platz für Neues entsteht. So weit so gut, doch bezogen auf den Wissensaufbau steht auch immer die Frage im Raum, wann genau das Wissen um ein Gebiet fallen gelassen werden kann? Diese Frage kann selbstverständlich nur individuell beantwortet werden.

Aber Vorsicht! Nicht alles, was wir wegwerfen, wird wirklich durch Besseres ersetzt, und das Neue ist nicht immer einfacher und schneller. Oft ist das Neue nur aus Sicht des Marketings einfacher. Aufgrund der gesteigerten Möglichkeiten der neueren Technologien, werden die Einsatzgebiete breiter, die möglichen abzudeckenden Anwendungsfälle komplexer. Das stimuliert wiederum die Besteller, für noch weniger Geld noch mehr Funktionalität zu verlangen und fertig ist das Regelwerk, das man Markt nennt.

Seriöse Softwareentwicklung mit alt hergebrachten Werten, wie das Bestreben nach einer möglichst selbsterklärenden Benutzeroberfläche, einer schlagkräftigen Hilfestellung oder der Möglichkeit, zur Laufzeit auch im Hintergrund technische Informationen des Ablaufs der Programme mitzuschneiden, sind selten geworden.

Ich bin der Meinung, dass im Moment ein enormer Umbruch am Softwaremarkt im Gange ist. Der Umbruch ist geprägt von der zunehmenden Vernetzung der Anwendungen, durch die immer größere Verbreitung von einer Anwendung auf verschiedenen Betriebssystemen, und nicht zuletzt durch die veränderte Dynamik von Projekten durch neue Entwicklungsparadigmen wie testgesteuerte Entwicklung und Scrum. Weil nicht alle Betriebssysteme über die gleichen technischen Möglichkeiten verfügen, und nicht alle Softwarebibliotheken für alle Betriebssysteme oder deren Versionen verfügbar sind, ist der Einfluss für Vorhaben, die auf vielen Plattformen funktionieren, enorm. Das führt dazu, dass Bedienkonzepte von Anwendungen verwässert und technische Implementierungen halbherzig angeglichen werden.

Dieses Buch macht im Bereich der Benutzeroberfläche den Spagat von der einfachen, eher konservativ anmutenden Windows Forms Technik bis hin zu Silverlight. Das ist eine geballte Technologierklärung auf ca. 1000 Seiten. Möglich wurde das nur durch das Gesamtkonzept des Handbuchs der .NET 4.0-Programmierung, das in drei Bänden die Grundlagen der .NET Technik, die Techniken der Benutzeroberflächen und die Kommunikationen beleuchtet.

Ich bedanke mich bei allen, die mich in dem Jahr der Entstehung des zweiten Bands meines Werks unterstützt haben. Insbesondere Kollegen in Projekten, die mir immer wieder Gelegenheit geben, Vor- und Nachteile der Technik aus Sicht des realen Lebens zu diskutieren und zu erfahren, aber auch den Studenten, die während der Entstehung mit vielen Hinweisen die Qualität verbessern halfen. Ein besonderer Dank gilt allerdings Frau Sandra Michel und Herrn Florian Helmchen, die mich von Seiten des Verlags betreut haben, und mit unendlicher Geduld alle Verzögerungen hingenommen haben. Ebenfalls danke ich Herrn Uwe Thiemann, der als Fachlektor dabei half, meine sich manchmal überstürzenden Gedanken zu ordnen und mein Schweizerdeutsch in ein verständlicheres Hochdeutsch zu übersetzen.

Der größte Dank allerdings gilt nach wie vor meiner Familie. Alle haben immer wieder Verständnis gezeigt, als ich in dem einen Jahr, in dem ich an diesem Buch schrieb, vor allem durch Abwesenheit glänzte. Es ist unglaublich schön, eine so breite Unterstützung zu genießen, wenn es darum geht, dass ich tagtäglich die Chance erhalte, besser zu werden, damit ich nicht aufhöre gut zu sein.

Rolf Wenger im Juli 2011

Einleitung

Das Handbuch der .NET 4.0-Programmierung gliedert sich in drei Bände.

- Band 1 beschäftigt sich mit den grundlegenden Techniken von .NET. Die in diesem Band angesprochenen Elemente werden wohl die meisten auf die eine oder andere Weise in einem konkreten Projekt angewendet. Ein Hauptbestandteil des Handbuchs bildet die Programmiersprache C# und LINQ (Language Integrated Query). Die beiden Elemente bilden den Leim zwischen den einzelnen Bausteinen der BCL (Base Class Library), die in diesem Band vorgestellt wird.
- Band 2 beschäftigt sich mit den Techniken und der Gestaltung von Benutzeroberflächen unserer Anwendungen. Dabei werden die Techniken Windows Forms, Windows Presentation Foundation, ASP.NET und Silverlight erklärt und gegenübergestellt. Viele Details in den Erklärungen machen auch diesen Band zum Nachschlagewerk.
- Band 3 widmet sich der Herstellung von verteilten Anwendungen. Entsprechend befindet sich die Windows Communication Foundation (WCF) im Zentrum des Bands. Ergänzt wird der Band mit Hinweisen zur Installation von Anwendungen.

Zielpublikum dieses Buchs

Als Generalist werden Sie über .NET und seine Anwendung wissen wollen, wo und mit welchem Aufwand diese neuen Technologien einsetzbar sind. Dieser Band gibt Ihnen einen breiten Einblick in die aktuellen grundlegenden Techniken von .NET.

Als Architekt von Lösungen wird Sie interessieren, welche Techniken für welche Lösungsansätze .NET zu bieten hat. Fragen zu Persistenz, Aufbau von Clientsoftware und technische Implementierungen von Software beantwortet dieses Buch genauso, wie die Fragen um die Wahl der richtigen Servertechnik.

Als Programmierer sind Sie interessiert zu erfahren, wie Sie eine konkrete Technik anwenden oder umsetzen. Sie finden im vorliegenden Werk zahlreiche, reich kommentierte Codebeispiele, Ideen für Lösungsansätze der Programmierung und ein sehr kompaktes Nachschlagewerk zur Programmierung mit .NET.

Der Nutzen dieses Buchs

Der Nutzen des Buchs liegt in der Kompaktheit und zugleich der Tiefe des Stoffes. Wieso finde ich drei Bände und über 2000 Seiten kompakt? Das Buch deckt eine bisher noch selten da gewesene Breite der .NET-Technologie in einem Band ab. Die Beschreibungen beschränken sich nicht nur auf die theoretischen Betrachtungen, sondern liefern viele Praxishinweise und konkrete Umsetzungen mit Beispielcode für die Programmierung bis hin zur Systemkonfiguration.

Wenn Sie das Handbuch an Ihrem Arbeitsplatz zur Hand haben, werden Sie es bald als Nachschlagewerk in der Praxis benutzen, denn das Handbuch ist themenorientiert aufgebaut und definiert dazu die tagtäglichen und viele auch nicht alltägliche Handgriffe, Tipps und Tricks. Prüfen Sie sich selbst:

- Welches ist die ideale Klasse für die Verarbeitung einer Auflistung von Objekten?
- Wie setze ich eine Vererbung in eine relationale Datenbank um?

- Wie konfiguriere ich einen Webserver und die Anwendung für die Nutzung der formularbasierten Authentifizierung?
- Welches ist die schnellste Client/Server-Technik?
- In welcher Genauigkeit arbeitet die Klasse `Timer`?
- Wie programmiere ich einen eigenen Editor zu einem `PropertyGrid`-Steuerelement?
- Wie erstelle ich eine serverseitige asynchrone Verarbeitung?
- Wie programmiere ich eine Ablaufverfolgung?
- Kann eine generische Klasse von einer anderen generischen Klasse erben?
- Wie erstelle ich einen Webdienst nach der Contract-First-Methode?

Diese Fragen und noch viele mehr beantwortet das Handbuch der .NET 4.0-Programmierung in drei Bänden. Dies geschieht mit Hilfe von Beschreibungen, über 1000 Abbildungen und zahlreichen Tabellen und Codebeispielen.

Der Benutzer, die Benutzer, die Benutzerin

Da Band 2 der Buchreihe ausschließlich die Techniken für die Benutzeroberfläche diskutiert, ist auch in diesem Band besonders häufig von den Benutzern die Rede. Ich betone, dass für mich der Begriff »Benutzer« in diesem Buch im Sinne von »Benutzer und Benutzerinnen« gemeint ist. Ich habe mich für die Kurzform der beiden Nennungen entschieden, weil die Texte so einfacher und flüssiger zu lesen sind.

Die Codebeispiele

Das Handbuch der .NET 4.0-Programmierung umfasst viele Codebeispiele. Die Beispiele sind nach Kapitel und Themengebiet geordnet. Die Organisation der Beispiele ist der Kapitelstruktur nachempfunden, und somit sicher für alle einfach verständlich. Um Ihnen die Referenzierung zwischen dem Buch und dem Beispielcode zusätzlich zu erleichtern, ist beim abgedruckten Code in der Listingunterschrift das Beispiel direkt als Textreferenz definiert.

Dazu ein Beispiel:

```
<Label Grid.Row="1" Content="Einzeilige Eingabe:" Margin="10,0"/>
<TextBox Grid.Row="1" Grid.Column="1" Margin="5,2" AutoWordSelection="True"
    SelectionBrush="Red" SelectionOpacity="0.2" CaretBrush="Red"/>
<Label Grid.Row="2" Content="Mehrzeilige Eingabe:" Margin="10,0" />
<TextBox Grid.Row="2" Grid.Column="1" Margin="5,2" TextWrapping="Wrap" AcceptsReturn="True"
    VerticalScrollBarVisibility="Visible" />
```

Listing E.1 Original Nachrichtenstruktur von Windows (C-Stil)

Die Unterschrift im Beispiellisting 8.6 definiert, dass Sie den Originalcode in der Datei `CpnlTextBox.xaml` im Verzeichnis `Band_2\Kapitel_08\LearningWpf_04\LearningWpf_04\10_SimpleControls` finden. In einigen Fällen ist zudem eine Coderegion angegeben. Suchen Sie in der bezeichneten Datei mit Hilfe der Bezeichnung der Coderegion oder wenn keine Coderegion angegeben ist, mit dem Namen der abgedruckten Methode oder einem sonst auffälligen Namen.

Aus Platzgründen haben wir viele Codebeispiele gekürzt abgedruckt. Dazu gehören:

- Direkte Weglassungen von Code. Diese sind jeweils an der stellvertretenden Codezeile (Wiedergabe gekürzt) oder an der Zeichenfolge ... erkennbar.
- Löschung von leeren Ausgabezeilen. Diese Kürzungen sind im abgedruckten Code nicht ersichtlich.
- Geänderte Anordnung von Codeblöcken. Codeblöcke werden in den abgedruckten Beispielen meistens in der einleitenden Zeile geöffnet, und nicht wie im Editor üblich, in einer separaten Zeile (Zeichen {}).
- Die XML-Kommentierung von Code ist nicht abgedruckt

Der Code liegt dem Buch nicht als CD bei. Sie können den Code allerdings bei Microsoft auf der offiziellen Webseite zum Buch auf <http://www.microsoft-press.de/support.aspx?s110=439> oder <http://msp.oreilly.de/support/9783866454392/663> oder von meiner eigenen Webseite (<http://www.weroSoft.net>) herunterladen. Auf meiner Webseite finden Sie neben dem Beispielcode falls nötig auch Dokumentationen und weiterführende Angaben.

Ich mache Sie darauf aufmerksam, dass einige Beispiele und Messresultate oder Ausgaben, die im Buch abgedruckt sind, bei der Ausführung auf Ihrem Rechner anders aussehen können. Das wird immer dann der Fall sein, wenn Angaben von der Rechnerphysik oder -logik abhängen.

WICHTIG Die Codebeispiele enthalten meistens keine spezielle Ausnahmebehandlung oder Variantenprüfung. Ferner existieren viele Coderedundanzen zwischen den einzelnen Beispielen. Es ist mir bewusst, dass der Code somit nicht immer praxisorientiert aufgebaut ist. Das ist aber in vollem Bewusstsein geschehen, da der Fokus eines Beispiels sich möglichst klar herauschälen soll. Zu viele Plausibilisierungen und sonstige in der Praxis notwendigen Codierungen würden zu stark von der Darstellung des effektiven Beispiels ablenken.

Ein Wort zu den Bezeichnern

Auch wenn Microsoft die Verwendung von Präfixen nicht mehr empfiehlt, beharre ich persönlich insbesondere bei Code zu Ausbildungszwecken darauf. Gerade der oft herausgestrichene Vorteil der Codetransparenz (alles sieht gleich aus) ist aus meiner Erfahrung vor allem ein Nachteil, denn ich erkenne nicht, was von mir und was von anderen Herstellern in den Code eingebracht wurde. Nun, je nach persönlicher Neigung sind Sie froh oder stören sich daran, dass ich an den Typpräfixen festhalte. Auf alle Fälle sollte es Ihnen leicht fallen, meinen Code von .NET-Bibliothekscodes zu unterscheiden, und gerade das ist in einem Nachschlagewerk meiner Meinung nach wertvoll.

Feedback

Ich will mit diesem Buch ein wertvolles Element für die Ausbildung und die tägliche Arbeit schaffen. Das ist die gute Absicht. Die Realität wird sein, dass ich es fertig bringe, trotz mehrfachen Lesens von mehreren Personen Fehler im Ganzen zu hinterlassen. Sollten Sie einen solchen entdecken und haben Sie gerade fünf Minuten Zeit, schauen Sie auf meiner Korrigenda zum Buch auf der Webseite nach, ob bereits eine Korrektur angemeldet ist. Wenn nicht können Sie direkt von dort aus den Fehler an mich melden.

URL Korrigenda: <http://www.korrigenda.literatur.weroSoft.net>

URL Feedback: <http://www.feedback.literatur.weroSoft.net>

URL Buchseite: <http://www.literatur.weroSoft.net>

URL Webseite: <http://www.weroSoft.net>