

# Vorwort

*»Wer aufhört besser zu werden,  
hat aufgehört gut zu sein.«*

Dieses Sprichwort hat bereits mein Erstlingswerk *Handbuch der .NET Programmierung* im Vorwort übertitelt. Ich habe es erneut hier abgedruckt, weil es für mich in keiner meiner bisherigen Lebensphasen so stark allgegenwärtig war, wie im Verlauf der letzten beiden Jahre. Die .NET-Technik hat sich seit 2005 im Markt etabliert und große industrielle Projekte verwenden die Technik im Alltag. Mit diesen Projekten haben wir gelernt, die Theorie und die Gedanken von Microsoft in die Praxis umzusetzen. Im gleichen Zeitraum hat Microsoft erworbene Erkenntnisse in neuen, bedeutenden Versionen von .NET auf den Markt gebracht. Damit hält der Hersteller die Spirale von Anwendung, Erkenntnis und Innovation am Drehen.

Innovation in der Informatik ist der Motor des Markts. Durch die technischen Innovationen werden neue Möglichkeiten geschaffen. Diese wiederum generieren neue Bedürfnisse der Kunden und zwingen uns Informatiker, uns weiterzubilden. Diese beiden Faktoren sind die Grundlage neuer Projekte und Produkte, mit denen die wirtschaftliche Entwicklung am Leben gehalten wird.

Selbstverständlich kann meine Trilogie »Handbuch der .NET 4.0 Programmierung« nicht alle Details dessen abdecken, was Sie in einem Projekt benötigen. Aber aus Sicht der Programmierung ist das hier überreichte Wissen eine geballte Ladung. Wenn Sie die vielen Hinweise und Kommentare zur Technik alle beherzigen, dann sind Sie einer richtigen Lösung schon ein gutes Stück näher.

Das vorliegende Werk habe ich mit aller Vorsicht und viel Enthusiasmus in unzähligen Stunden neben meiner täglichen Arbeit hergestellt. Zu dem Buch gehören zahlreiche Beispiele, die die beschriebenen Techniken implementieren. Obschon die Qualitätssicherung bei mir einen hohen Stellenwert genießt, wird es einige Fehler im Werk geben. Ich gehe nicht davon aus, dass meine Werke von den generellen statistischen Untersuchungen des Softwaremarktes ausgenommen sind.

Ich bedanke mich bei allen, die mich im Jahr des Entstehens des ersten Bandes 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 als Verlegerin. Sie hat mich mit einer unendlichen Geduld bei den vielen Verzögerungen stets unterstützt und ermuntert. Ebenfalls danke ich Herrn Uwe Thiemann, der als Fachlektor meine sich manchmal überstürzenden Gedanken geholfen hat zu ordnen und mein Schweizerdeutsch in ein verständlicheres Hochdeutsch übersetzt hat.

Der größte Dank allerdings gehört meiner Familie. Alle haben immer wieder Verständnis gezeigt, als ich während einem Jahr vor allem durch Abwesenheit glänzte. Es ist unglaublich schön, eine so breite Unterstützung zu genießen, wenn es darum geht, dass ich tagtäglich die Chance kriege besser zu werden, damit ich nicht aufhöre gut zu sein.

# 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 in der einen oder anderen Art in einem konkreten Projekt anwenden. 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 ist.
- Band 2 beschäftigt sich mit den Techniken und der Gestaltung von Benutzerschnittstellen unserer Anwendungen. Dabei werden die verschiedenen Techniken im Vergleich diskutiert und viele Details vorgestellt.
- Band 3 widmet sich der Herstellung von verteilten Anwendungen. Entsprechend ist 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 von .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 Handgriffe. Insbesondere auch die Handgriffe, die eben nicht alltäglich sind. 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 Benutzung von Forms-Authentifizierung?
- Welches ist die schnellste Client/Server-Technik?
- In welcher Genauigkeit arbeitet die Klasse `Timer`?
- Wie programmiere ich einen eigenen Editor zu einem Steuerelement `PropertyGrid`?
- 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 mit Beschreibung, über 700 Abbildungen und zahlreichen Tabellen und Codebeispielen.

## 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:

```
(Wiedergabe gekürzt)
// Hilfsmethode für die Konvertierung
static void ConvertToDecimal(string strValue, IFormatProvider objFormat) {
    try {
        Console.WriteLine(Convert.ToDecimal(strValue, objFormat));
    }
    catch (Exception exc) { ... }
}
```

**Listing 0.1** Band\_1\Kapitel\_02\CSharpSpecials\Specials\Program.cs#region Konvertierungen

Die Unterschrift in Beispiellisting 0.1 definiert, dass Sie den Originalcode in der Datei `Program.cs` im Verzeichnis `Band_1\Kapitel_02\CSharpSpecials\Specials` finden. In einigen Fällen ist zudem eine Coderegion angegeben. Suchen Sie in der bezeichneten Datei mithilfe der Bezeichnung der Coderegion (hier `#region Konvertierungen`) oder wenn keine Coderegion angegeben ist, mit dem Namen der abgedruckten Methode. Im Beispiel ist das `ConvertToDecimal()`.

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 auf der einleitenden Zeile geöffnet, und nicht wie im Editor üblich, in einer separaten Zeile (Zeichen {}).
- Die XML-Kommentierung von Code ist nicht abgedruckt. Speziell im Band 1 kommt dazu, dass die normalen Kommentierungen oft durch normale Ausgaben auf die Konsole ersetzt wurden. Das sieht für geübte Leser von Code zunächst ein wenig fremd aus, bot mir aber den Vorteil, dass ich nicht kommentieren musste, und zusätzlich auch noch eine Ausgabe für die normale Benutzung programmieren musste.

Der Code liegt dem Buch nicht als CD bei. Sie können den Code allerdings bei Microsoft auf der offiziellen Buchreferenz oder auf meiner eigenen Website (<http://www.weroSoft.net>) herunterladen. Auf meiner Website finden Sie neben dem Beispielcode 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 5 Minuten Zeit, schauen Sie auf meiner Korrigenda zum Buch auf der Website 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.ch>

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

## Support für dieses Buch

Dieses Buch wurde mit großer Sorgfalt erstellt. Korrekturen zum Buch bietet Microsoft Press unter folgender Webadresse:

<http://go.microsoft.com/fwlink/?LinkID=194575&clid=0x407>

Wenn Sie Kommentare, Fragen oder Anregungen zum Buch haben, die sich nicht durch eine Abfrage der Knowledge Base beantworten lassen, wenden Sie sich an Microsoft Press:

Per E-Mail:

[presscd@microsoft.com](mailto:presscd@microsoft.com)

Per Post:

Microsoft Press

Betrifft: *Windows 7 für Entwickler*

Konrad-Zuse-Straße 1

85716 Unterschleißheim

Den Support von Microsoft Press erreichen Sie unter <http://www.microsoft-press.de/support.asp>.

Beachten Sie bitte, dass unter den genannten Adressen kein Produktsupport geleistet wird. Informationen über den Produktsupport finden Sie auf der Microsoft-Produktsupportwebsite unter der Adresse <http://support.microsoft.com>.

# Rolf Wenger – zur Person

Rolf Wenger beschäftigt sich seit seinem Studienabschluss 1987 mit der Entwicklung von Software für Windows-Systeme. Nach Projekten auf der Basis von C entwickelte er bis zur Einführung von .NET mit C++ und der Microsoft Foundation Class Library (MFC) viele Projekte, darunter zum Teil umfangreiche Client/Server-Anwendungen. Der Umstieg und somit der Beginn des Know-how-Aufbaus für .NET fand mit den ersten Beta-Versionen statt.

Sein Wissen hat er vielen Firmenkunden und in ständigen Ausbildungen an der Fachhochschule Bern, Abteilung Software Schule Schweiz, seit 1997 weitergegeben. Aktuell unterrichtet er an diesem Institut die .NET-Technologien in einem ganzheitlichen Blockkurs.

Privat lebt Rolf Wenger mit seiner Familie in Ostermundigen bei Bern in der Schweiz. Wenn er gerade nicht seiner Passion, der Entwicklung von Software, nachkommt, versucht er seine Gedanken in Form von Büchern zu ordnen und sich für die nächsten Ausbildungen vorzubereiten. Nebenbei sucht er den Ausgleich zum Alltag im Sport und zusammen mit der Familie beim Reisen und Fotografieren.

Sie erreichen Rolf Wenger unter:

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

Mail: *[info@weroSoft.net](mailto:info@weroSoft.net)*