

# Vorwort

Liebe Leserin, lieber Leser,

vor Ihnen liegt die vierte Ausgabe des ASP.NET-Entwicklerbuchs in der C#-Variante. Das wurde aktualisiert auf ASP.NET 4.0 mit C# 2010 und Visual Studio 2010.

## Bitte lesen Sie die FAQ

Normalerweise enthalten die Vorworte meiner Bücher eine Zielgruppendefinition sowie Hinweise zum Aufbau und zur Leseart des Buchs. Wie in der Voraufgabe verwende ich hier jedoch ein anderes Format: Im Anschluss an dieses Vorwort finden Sie eine umfangreiche *Frage-Antwort-Sammlung (FAQ)* über dieses Buch. Vor dem Kauf des Buchs sollten Sie diese Frage-Antwort-Sammlung studieren, damit Ihre Erwartungen nicht enttäuscht werden. Softwareentwickler sind eine sehr heterogene Zielgruppe. Es ist unmöglich, alle Softwareentwickler mit einem einzigen Buch zu erreichen. Ich spreche mit diesem Buch eine Zielgruppe an, die ein bestimmtes Vorwissen besitzt und die eine bestimmte Herangehensweise bevorzugt. Für alle anderen ist dieses Buch nicht oder nur bedingt geeignet. Ich möchte vermeiden, dass Sie dieses Buch mit falschen Annahmen erwerben und dann unzufrieden sind.

## Danksagungen

Meinen Dank für ihre Mitwirkung an diesem Buch möchte ich aussprechen an

- meinen Assistenten *Sebastian Stein*, der für mich einige Bildschirmabbildungen auf die neue Version übertragen hat,
- den Fachlektor *Thomas Irlbeck*, der die Inhalte des Buchs und die Codebeispiele überprüft hat,
- Frau *Silja Brands*, die das Buch gesetzt hat,
- die Lektorin *Sandra Michel*, die dieses Buch von Verlagsseite betreut hat,
- die Korrektorin *Kristin Grauthoff*, die das Buch sprachlich verbessert hat und
- das MVP-Programm bei Microsoft, das mir den direkten Zugang zu Software und Informationen ermöglicht hat.

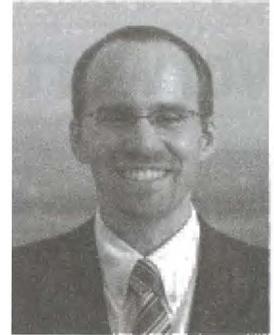
Viel Erfolg beim ASP.NET 4.0-Programmieren mit diesem Buch wünscht Ihnen

*Dr. Holger Schwichtenberg*

Essen, im Februar 2011

# Über den Autor

## Dr. Holger Schwichtenberg



[www.IT-Visions.de](http://www.IT-Visions.de)  
Dr. Holger Schwichtenberg

- Studienabschluss Diplom-Wirtschaftsinformatik an der Universität Essen
- Promotion an der Universität Essen im Gebiet komponentenbasierter Softwareentwicklung
- Seit 1996 selbstständig als unabhängiger Berater, Dozent, Softwarearchitekt und Fachjournalist
- Geschäftsführung und Leitung der Softwareentwicklung der Firma *www.IT-Visions.de*
- Mehr als 40 Fachbücher bei Microsoft Press, Addison-Wesley und dem Carl Hanser Verlag
- Mehr als 500 Beiträge in Fachzeitschriften
- Ständiger Mitarbeiter der Zeitschriften iX und dotnetpro sowie beim Onlineportal heise.de
- Regelmäßiger Sprecher auf nationalen und internationalen Fachkonferenzen (z.B. TechEd, Microsoft IT Forum, BASTA, Advanced Developers Conference, OOP, VS One, Wirtschaftsinformatik, Net.Object Days, Windows Forum, DOTNET-Konferenz, XML-in-Action)
- Zertifikate und Auszeichnungen von Microsoft:
  - Microsoft Most Valuable Professional (MVP)
  - Microsoft Certified Solution Developer (MCSD)
  - Mitglied im weltweiten Codezone Premier Website-Programm
- Thematische Schwerpunkte:
  - Softwarearchitektur, mehrschichtige Softwareentwicklung, Softwarekomponenten
  - Microsoft .NET Framework, Visual Studio, C#, Visual Basic
  - .NET-Strategie /Auswahl von .NET-Technologien
  - Einführung von .NET Framework und Visual Studio/Migration auf .NET
  - Webanwendungsentwicklung mit IIS, ASP.NET, AJAX und Silverlight
  - Enterprise .NET, verteilte Systeme mit .NET, insbesondere Windows Communication Foundation
  - Relationale Datenbanken, XML, Datenzugriffsstrategien
  - Objekt-Relationales Mapping (ORM), insbesondere ADO.NET Entity Framework
  - Windows PowerShell (WPS) und Windows Management Instrumentation (WMI)
- Ehrenamtliche Community-Tätigkeiten:
  - Vorstandsmitglied bei codezone.de
  - Vortragender für die International .NET Association (INETA)  
Betrieb der Community-Websites *www.dotnetframework.de*, *www.windows-scripting.de* und *www.powershell-doktor.de*
- Firmenwebsite: *http://www.IT-Visions.de*
- Weblog: *http://www.dotnet-doktor.de* (bei Heise.de)
- Kontakt: *hs@IT-Visions.de* sowie Telefon 0201/7490-700

# Dienstleistungen für Leser nach dem Kauf

Den Lesern dieses Buchs werden vom Autor folgende Serviceleistungen im Rahmen einer zugangsbeschränkten Website angeboten:

- **Downloads** Die Programmcodebeispiele aus diesem Buch werden ständig aktualisiert und erweitert. Laden Sie im Leser-Portal Aktualisierungen des Programmcodes herunter.
- **Diskussionsrunde** Ein webbasiertes Forum bietet die Möglichkeit, Fragen an den Autor zu stellen. Bitte beachten Sie jedoch, dass dies eine freiwillige Leistung des Autors ist und kein Anspruch auf eine kostenlose Betreuung besteht.
- **Newsletter** Alle registrierten Leser erhalten mehrmals jährlich einen Newsletter mit aktuellen Terminen und Publikationshinweisen.
- **Leser-Bewertung** Geben Sie Noten für dieses Buch und lesen Sie nach, was andere Leser von diesem Buch halten.
- **Errata** Trotz eines erprobten Vorgehensmodells und der mehrfachen Qualitätskontrolle ist es möglich, dass sich einzelne Fehler in dieses Buch eingeschlichen haben. Im Webportal können Sie nachlesen, welche Fehler gefunden wurden. Sie können hier auch selbst Fehler melden, die Ihnen auffallen.

## Zugang zum Leser-Portal

Die Internetadresse für den Zugang zum Leser-Portal lautet:

<http://www.IT-Visions.de/leser>

Bei der Anmeldung müssen Sie das Lösungswort **Futurama** angeben. Sie erhalten dann per E-Mail Ihr persönliches Zugangskennwort, mit dem Sie die Leser-Website betreten und verwenden können.

## Fragen und Antworten zum Buch (FAQ)

Dieser Abschnitt liefert Informationen über dieses Buch in Frage-Antwort-Form. Er beantwortet im Vorfeld viele Fragen, die dem Autor zu anderen Buchprojekten gestellt wurden, und enthält Informationen, die sonst üblicherweise im eigentlichen Vorwort enthalten sind.

## Fragen zum Inhalt des Buchs

### Wer ist die Zielgruppe dieses Buchs?

Zielgruppen dieses Buchs sind

- Professionelle Webanwendungsentwickler
- Architekten und Projektleiter im Bereich der Webanwendungsentwicklung
- Dozenten für ASP.NET
- Fortgeschrittene Hobby-Entwickler von Websites

## Welche Kenntnisse werden in diesem Buch vorausgesetzt?

Das Buch kann nicht alle Grundlagen behandeln. Der Autor setzt daher folgende Kenntnisse bei den Lesern voraus:

- Gute Kenntnisse in HTTP, HTML, XML und CSS
- Gute Kenntnisse in einer objektorientierten Programmiersprache (z. B. Java, C++, Delphi, Visual Basic 6.0)
- Gute Kenntnisse in relationalen Datenbanken und SQL
- Gute Kenntnisse des Betriebssystems Windows einschließlich Microsoft Internet Information Services (IIS) und Internet Explorer
- Basiskenntnisse in DHTML und JavaScript
- Basiskenntnisse in Cascading Style Sheets (CSS)
- Basiskenntnisse in der Entwicklung serverseitiger Webanwendungen
- Basiskenntnisse bezüglich Microsoft .NET Framework (\*)
- Basiskenntnisse in C# oder Visual Basic (\*)
- Basiskenntnisse in Bezug auf Visual Studio (\*)
- Basiskenntnisse bezüglich der .NET-Klassen für ADO.NET, XML und LINQ (\*)

Vorkenntnisse in ASP.NET werden ausdrücklich *nicht* vorausgesetzt.

---

**HINWEIS**

Sofern Sie die mit (\*) gekennzeichneten Vorkenntnisse nicht besitzen, empfiehlt sich die Lektüre des Buchs:

*Holger Schwichtenberg, .NET 4.0 Crashkurs, Microsoft Press 2010*

Auszüge dieses Buchs finden Sie als PDF-Dokument im Leser-Webportal zu diesem Buch.

---

## Werden in diesem Buch auch die Grundlagen von .NET behandelt?

Nein. Da ASP.NET inzwischen so umfangreich ist, dass es alleine mehr als ein Buch füllen kann, decken alle gedruckten Seiten nur das Kernthema ab. Sofern keine Vorkenntnisse in .NET, den .NET-Programmiersprachen sowie den .NET-Klassenbibliotheken vorhanden sind, empfiehlt sich die Lektüre des im vorherigen Eintrag genannten Buchs.

## Welche Version von ASP.NET wird beschrieben?

Das Buch basiert auf .NET Framework Version 4.0 (Version 4.0.30319) sowie Visual Studio 2010 mit (Version 10.0.30319.1), beides erschienen am 12.4.2010.

## Für wen ist dieses Buch nicht geeignet?

Das Buch ist nicht geeignet für diejenigen, die

- eine Schritt-für-Schritt-Anleitung (»Klick-Anleitung«) benötigen und keinerlei Vorkenntnisse besitzen.
- bereits über sehr viel Erfahrung mit ASP.NET verfügen und nur kleinere Details und Spezialwissen suchen.

## Wo liegt der Schwerpunkt dieses Buchs?

ASP.NET ist ein sehr umfangreiches Thema. Auf der einen Seite gibt es sehr viele technische Konzepte, die es zu beherrschen gilt, auf der anderen Seite einen umfangreichen Katalog von Steuerelementen mit vielen Tausend Optionen. Ich habe mich ganz bewusst dazu entschlossen, den Schwerpunkt dieses Buchs auf die technischen Konzepte zu legen. Ganz abgesehen davon, dass es auf ca. 1000 Seiten nicht möglich wäre, die Steuerelemente vollständig zu beschreiben, sehe ich auch nur wenig Mehrwert in einer detaillierten Diskussion aller Steuerelemente. In der MSDN-Bibliothek und anderen Büchern gibt es bereits eine hinreichende Beschreibung der Steuerelementoptionen.

Ich möchte die mir zur Verfügung stehenden Seiten mehr darauf verwenden, die Themen zu diskutieren,

- von denen ich tagtäglich erlebe, dass sie bei meinen Kunden eine Herausforderung darstellen und/oder
- die in der MSDN-Bibliothek nicht, nur unvollständig oder zu komplex behandelt werden.

Dies sind insbesondere:

- Grundprinzipien von ASP.NET Page Framework (z.B. Serversteuerelemente, Ereignismodell und Seitenübergänge)
- Dienste von Page Framework (z.B. Zustandsverwaltung, Zwischenspeicherung und Sicherheit)
- Architekturfragen zur Webentwicklung mit ASP.NET (z.B. Datenbindung und Mehrschichtenanwendungen)
- Werkzeugunterstützung (z.B. Kompilierungsoptionen)

Um ein vollständiges Bild zu ergeben, werden selbstverständlich auch Steuerelemente thematisiert. Hierbei geht es aber immer nur um ausgewählte Optionen, niemals darum, die MSDN-Bibliothek nacherzählen zu wollen.

## Welche Themen werden in diesem Buch nicht behandelt?

Einige der Themen, die in diesem Buch nicht abgedeckt werden konnten, möchte ich explizit benennen:

- Grundlagenwissen zu allen allgemeinen Standards wie z.B. HTTP, HTML, CSS, XML, JavaScript sowie SOAP und WSDL
- Gestaltungsthemen (Diese werden nur am Rande behandelt. Details zu den gestalterischen Optionen der Steuerelemente könnten eigene Bücher füllen. Dies ist aber ein Buch für Softwareentwickler!)
- Das Konfigurations-API von ASP.NET (Klassen im Namensraum `System.Web.Configuration`) zur programmgesteuerten Konfiguration (In diesem Buch werden nur die deklarativen Möglichkeiten und die Werkzeuge beschrieben.)
- Erstellung eigener Konfigurationsdateiabchnitte durch Implementierung von `System.Configuration.IConfigurationSectionHandler`
- Asynchrone Seitenaufrufe
- Die Erstellung eigener AJAX-Steuerelemente und Extender-Steuerelemente
- ASP.NET in Verbindung mit anderen Programmiersprachen
- Verteilung von Webprojekten mit MSI-Paketen
- ASP.NET Model-View-Controller Framework (MVC)
- ASP.NET Razor-Syntax und WebMatrix

- Die inoffiziellen Erweiterungen der Web Client Software Factory (<http://www.codeplex.com/websf>)
- Steuerelemente von Drittanbietern (z. B. von Telerik, Infragistics und Developer Express)
- Alle Vorabversionen potenzieller zukünftiger ASP.NET-Funktionen, die Microsoft unter <http://aspnet.codeplex.com> präsentiert
- Microsoft Silverlight (siehe dazu [HSJF01])

### Warum lassen Sie manche Details aus?

ASP.NET ist ein komplexes und sehr umfangreiches Thema. Ich schätze, für eine komplette Darstellung des Themas ASP.NET müsste man 10.000–12.000 Seiten schreiben. Folglich musste ich auswählen, welche Themen in das Buch kommen und welche nicht. Die Seitenzahl eines Buchs lässt sich leider aus betriebswirtschaftlichen und produktionstechnischen Gründen nicht unbegrenzt nach oben erweitern.

### Finde ich in diesem Buch Informationen, die es nicht in der MSDN-Bibliothek gibt?

Kaufen Sie dieses Buch nicht, wenn Sie der Meinung sind, dass die MSDN Library eine Informationsquelle ist, in der Sie *schnell* alles finden, was Sie brauchen. In diesem Buch stehen nur wenige Dinge, die Sie nicht auch irgendwo in der MSDN Library finden können. Tatsächlich gibt es fast keine Bücher zu Microsoft-Technologien, die Themen behandeln, die nicht auch durch die MSDN-Bibliothek abgedeckt sind, denn ein Buch kann natürlich auf rund 1000 Seiten nicht mehrere Hunderttausend Seiten der MSDN-Bibliothek inhaltlich übertreffen. Dieses Buch bietet Informationen in einer *anderen Form*: komprimiert, praxisnäher und selektiver. Sie brauchen dieses Buch, wenn Sie nicht die Zeit haben, lange in der MSDN-Bibliothek zu suchen und sich durch lange Texte zu arbeiten, aber komprimierte und praxisnahe Informationen aus einer Hand zu schätzen wissen.

### Warum schreiben Sie nicht ein Buch für »echte Profi-Entwickler«?

Diese Frage wird mir hin und wieder einmal in dieser provokanten Form gestellt. Die grundsätzliche Schwierigkeit besteht darin, zu definieren, was ein »echter Profi-Entwickler« ist. Die Definitionen darüber gehen ebenso weit auseinander wie das Vorwissen der in Deutschland als Softwareentwickler tätigen Personen.

Ich möchte die Softwareentwickler in drei Hauptgruppen unterteilen:

1. Entwickler, die eine detaillierte Einführung und Schritt-für-Schritt-Anleitung (»Klick-Anleitung«) benötigen.
2. Entwickler, die anhand von Beispielen und kurzen Erklärungen in der Lage sind, sich in eine neue Technologie selbst einzuarbeiten.
3. Entwickler, die bereits über viel Vorwissen verfügen und nur kleinere Details und Spezialwissen suchen.

Die meisten meiner Bücher richten sich an Zielgruppe 2. Für Zielgruppe 1 gibt es viele Bücher anderer Autoren. Für die Zielgruppe 3 habe ich in der Vergangenheit Bücher angeboten, derzeit plane ich hier aber keine neuen Projekte mehr, da der Markt in Deutschland nicht groß genug zu sein scheint.

## Woher beziehen Sie Informationen, die Sie in diesem Buch niederschreiben?

Erste Informationen zu neuen Microsoft-Produkten erhalte ich meist auf Veranstaltungen von Microsoft (sowohl öffentliche Entwicklerveranstaltungen wie die Professional Developer Conference als auch »interne« Veranstaltungen) oder durch Dokumente, die mir Microsoft im Rahmen des MVP-Programms zur Verfügung stellt. Weitere Informationsquellen sind darüber hinaus die MSDN-Library, Artikel auf Websites, Weblogs und Newsgroups.

Meine wichtigste Informationsquelle sind aber eigene Erfahrungen, die ich mit dem Produkt sammle. In früheren Produktstadien sind es meist nur kleinere Testanwendungen. In der Betaphase entwickle ich oft bereits Lösungen für »mutige« Kunden, die dann bald nach dem endgültigen Veröffentlichungstermin des Produkts in auslieferungsfertige Lösungen münden. Bücher anderer Autoren stehen mir meist nicht zur Verfügung, weil ich viele Bücher schon zu einem frühen Produktstadium beginne.

## Wie treffen Sie die Entscheidung, welche Themen in das Buch kommen und welche nicht?

Die Entscheidungen, welche Themen ich in das Buch aufnehmen und welche nicht, basieren auf vier Punkten:

- meiner eigenen Projekterfahrung, welche Funktionen man braucht und welche nicht
- meiner Erfahrung, welche Funktionen Kunden einsetzen, die ich bei der Nutzung von .NET-Technologien berate
- Nachfragen von Teilnehmern an Schulungen und Vorträgen
- Auswertung des Feedbacks zu vorherigen Büchern auf der Leser-Website

## Wie werden die Informationen in diesem Buch dargestellt?

Wann immer sinnvoll und möglich, erfolgt die Darstellung in diesem Buch in Gestalt von Grafiken, Auflistungen und Tabellen. Tendenziell wird auf umfangreichen Fließtext zugunsten dieser Darstellungsformen verzichtet.

## Wie haben Sie die Unterscheidung zwischen Basiswissen, Aufbauwissen und Profiwissen getroffen?

Das Buch ist grundsätzlich nach technischen Konzepten gegliedert, also z.B. Datenbindung, Konfiguration, Zwischenspeicherung und Sicherheit. In jedem dieser Kapitel gibt es Grundlagen und fortgeschrittene Themen. Ich habe diese technischen Konzepte nicht auseinander gerissen, sondern aus meiner Erfahrung als Architekt und Berater entschieden. Das Kriterium war, welche Techniken üblicherweise immer, weniger häufig oder eher seltener bzw. welche Techniken von Einsteigern, Fortgeschrittenen oder Profis verwendet werden.

## Wie halten Sie es mit Anglizismen?

Ich versuche, Anglizismen zu vermeiden. Dies bedeutet z.B., dass ich *Zwischenspeicher* statt *Cache*, *entfernter Computer* statt *Remote Computer*, *Performanz* oder *Leistung* statt *Performance* und *Protokollierung* statt *Logging* verwende. Nur in Fällen, bei denen eine Übersetzung unverständlich erscheinen würde (z.B. *Client*, *Server*, *DataSet*, *DataReader*), bleibe ich bei den englischen Begriffen.

Begriffe habe ich nicht immer wörtlich, sondern zum Teil sinngemäß übersetzt (z.B. *Seitenzustand* statt *ViewState*).

**HINWEIS** Die Tatsache, dass einige Klassen in providerspezifischen Ausprägungen (mit zum Teil sehr unterschiedlichen Namen) vorliegen, macht die schriftliche Darstellung einiger Sachverhalte schwierig. Dieses Buch folgt dem üblichen Entwickler-sprachgebrauch, die Begriffe *Datareader*, *Datenadapter*, *Command* und *Connection* als Technologieoberbegriffe für die jeweiligen providerspezifischen Klassen zu nutzen – auch wenn diese Klassennamen in der Klassenbibliothek gar nicht existieren. Um den Text zu straffen, wird daher an verschiedenen Stellen synonym zu »Das DataReader-Objekt...« (mit Auszeichnung des Objektnamens in der Schriftart Minion Pro) einfach »Der Datareader...« (in der gewöhnlichen Fließschriftart) verwendet.

## Fragen zu Grafiken und Bildschirmabbildungen

### Sind die Bildschirmabbildungen in diesem Buch auf Deutsch oder Englisch?

Dieses Buch beschreibt die englische Version von Visual Studio, weil inzwischen viele deutsche Entwickler (einschließlich meiner eigenen Person) die englische Version der Software bevorzugen, da die Übersetzungen ins Deutsche oft holprig sind und die Fehlermeldungen nur schwerer verständlich machen. Als Kompromiss zwischen dem Kundenkreis mit englisch- und deutschsprachiger Entwicklungsumgebung wird dieses Buch – so oft es möglich ist, ohne den Lesefluss zu stören – sowohl die deutschen als auch die englischen Namen der Menüpunkte nennen. Die Bildschirmabbildungen sind aus Platzgründen allerdings immer nur in einer Sprache abgedruckt.

Ich verwende auf meinen Systemen Windows 7 (englisch mit optionaler deutscher Oberfläche) sowie Windows Server 2008 R2 (in einer englischen Version). Visual Studio setze ich ebenso fast ausschließlich in englischer Sprache ein.

### Warum sind in meinem Visual Studio Menüs und Symbolleisten zum Teil anders aufgebaut?

Die Anordnung der Menüs und auch einige Tastaturkürzel hängen von zwei Faktoren ab:

- von der Visual Studio-Edition, die Sie verwenden, und
- den gewählten Einstellungen in Visual Studio/Visual Web Developer

Alle Ausführungen in diesem Buch beziehen sich auf

- Visual Studio 2010 Ultimate
- die Umgebungseinstellung *Allgemeine Entwicklungseinstellungen (General Development Settings)*, die bei der Installation des Produkts ausgewählt werden kann

An einigen Stellen im Buch kommen zusätzliche Add-Ins (z.B. Web Deployment-Add-In und Web Application-Add-In) zum Einsatz.

## Fragen zu den Programmcodebeispielen

### Die Buch-DVD

Die Codebeispiele zum Buch finden Sie auf der diesem Buch beiliegenden DVD. Alternativ erhalten Sie alle Codebeispiele auch auf der Leser-Website <http://www.IT-Visions.de/leser> (Registrierung notwendig mit Kennwort *Futura*).

## Wo finde ich die Beispiele?

Zu diesem Buch gibt es drei Download-Pakete:

- Das Fallbeispiel »World Wide Wings« enthält den überwiegenden Teil des Programmcodes aus diesem Buch (siehe gleichnamiges Kapitel 3). Im Falle eines Downloads wählen Sie bitte Version 0.6 oder höher. Die Benutzeroberfläche der Webanwendung selbst führt Sie zu den Beispielen (insbesondere das Menü).
- *ASPNET40\_Demos.zip* enthält isolierte Beispiele zu den neuen Funktionen in ASP.NET 4.0. Auch hier wird die Datenbank von World Wide Wings verwendet. Im Falle eines Downloads führt Sie die Benutzeroberfläche der Webanwendung zu den Beispielen (insbesondere das Menü).
- *ASPNET\_ErsteAnwendungen.zip* enthält die Einsteigerbeispiele aus Kapitel 2 »Drei erste schnelle ASP.NET-Lösungen«. Auch hier wird die Datenbank von World Wide Wings verwendet.

## Gibt es verschiedene Sprachversionen der Beispiele?

Für viele, aber nicht alle Beispiele gibt jeweils eine Version in C# und in Visual Basic. Es war aber nicht mit vertretbarem Aufwand möglich, alle Beispiele in beiden Sprachen zu entwickeln. Außer bei Webprojekten gibt es immer zwei verschiedene Projekte pro Sprache. In dem World Wide Wings-Webprojekt sind beide Sprachen gemischt. Oft sind daher dort ASP-Seiten doppelt vorhanden. Im Allgemeinen enden die Dateinamen dann zur Unterscheidung der verwendeten Programmiersprache auf *\_CS.aspx* und *\_VB.aspx*, anhand der Dateierweiterung der Hintergrundcodedatei (*.cs* bzw. *.vb*) lässt sich ebenso auf die jeweils eingesetzte Programmiersprache schließen.

Die meisten Codebeispiele wurden in C# erstellt und mithilfe der Software *Instant VB* der Firma Tangible Software Solutions [TAN01] automatisch nach Visual Basic übersetzt.

## Nach welchen Richtlinien erstellen Sie Codebeispiele?

Ich verwende auf Basis meiner Erfahrung und auf Basis von Leserfeedback folgende Richtlinien zu den Programmcodebeispielen in diesem Buch:

- Die Beispiele sollen möglichst autonom sein, das heißt, ein abgedrucktes Listing soll alle notwendigen Elemente enthalten, auch wenn dies auf Kosten der Wiederverwendung von Programmcode geht.
- Alle notwendigen Konfigurationsinformationen (z.B. Computernamen, Verbindungszeichenfolgen) sollen Teil des Listings sein, auch wenn dies dazu führt, dass diese Daten etwas wenig elegant im Quelltext statt wie üblich in einer Konfigurationsdatei abgelegt werden.
- Beispiele sollen dann nicht autonom sein, wenn es explizit darum geht, die Wiederverwendbarkeit von Programmcode, die Nutzung von Konfigurationsdateien oder eine gute Softwarearchitektur darzustellen.
- Die Variablennamen in den Codebeispielen sollen einfach sein. Kurznamen, die nur aus einem oder zwei Buchstaben bestehen, sind entgegen den üblichen Gepflogenheiten erlaubt, um Platz zu sparen und Zeilenumbrüche in abgedruckten Listings zu vermeiden (dies erhöht die Lesbarkeit von Listings in einem Buch!).
- Bei allen Codebeispielen steht die Funktionalität im Vordergrund, nicht die Gestaltung der Ein- und Ausgabe.

## Was bedeutet die Methode Demo.Print(), die sich in einigen Listings findet?

Dieses Buch orientiert sich an einem übergreifenden Fallbeispiel, der Fluggesellschaft »World Wide Wings«. Während viele Programmcodebeispiele direkt in die entsprechenden Anwendungsteile des Gesamtbeispiels integriert sind, liegen aus didaktischen Gründen einige in autonomen, besser überschaubaren Codebeispielen vor. Diese Codebeispiele verwenden zwar oft auch Daten aus dem Fallbeispiel, sind jedoch einzeln startbar. Zum Start dieser autonomen Beispiele wird eine vom Autor dieses Buchs entwickelte Rahmenanwendung, der *Demo Viewer*, eingesetzt. Der DemoViewer erlaubt die übersichtliche Anzeige und den einfachen Start aller Beispiele, die sich in einer definierbaren Assembly befinden. Der DemoViewer existiert in zwei Varianten: in Form einer Windows-Anwendung und in Form einer ASP.NET-Webanwendung (Abbildung V.1).

Die Klasse Demo stellt Methoden bereit, damit ein Programmcodebeispiel Ausgaben in dem Fenster des DemoViewers erzeugen kann.

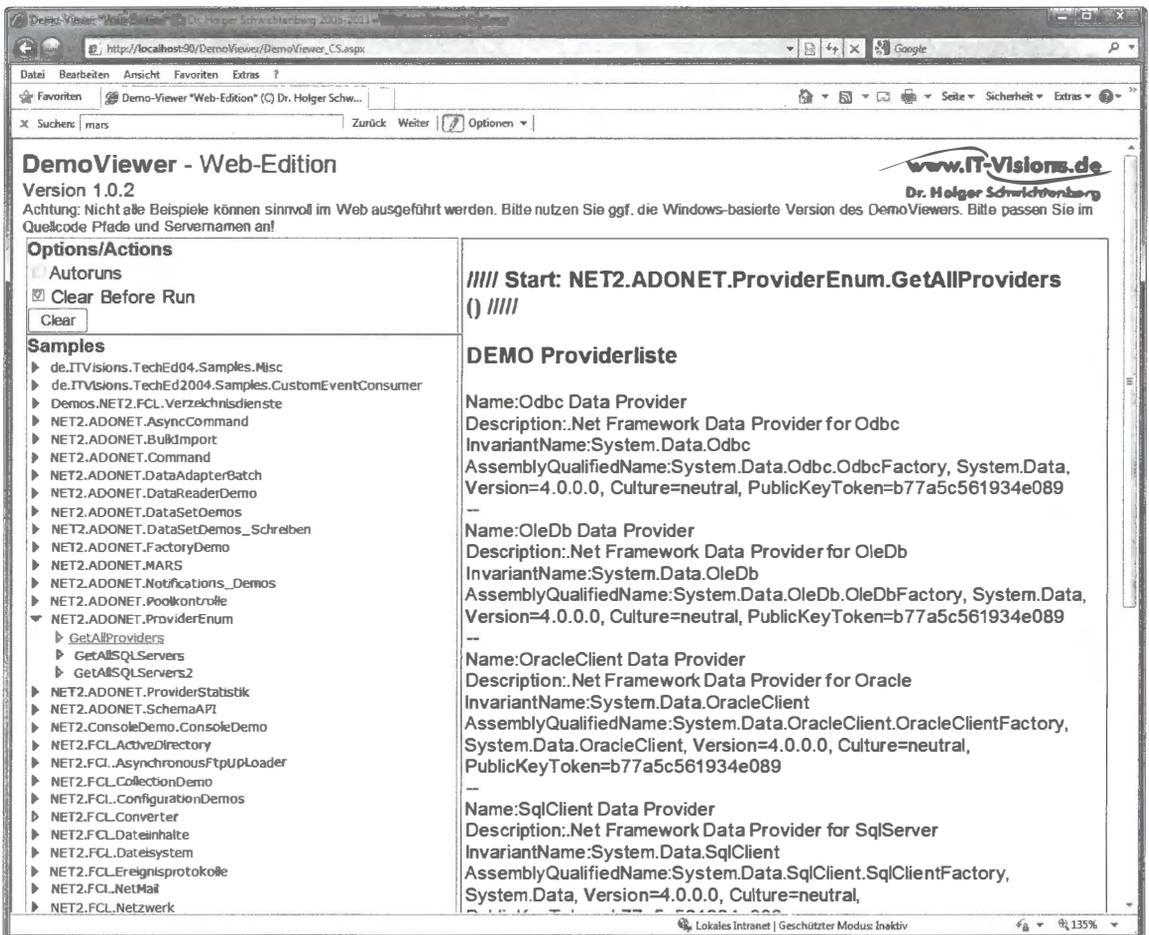


Abbildung V.1 Webbasierter DemoViewer

Sie werden in den herunterladbaren Beispielen zu diesem Buch bzw. den im Buch abgedruckten Listings folgende Vorkehrungen zur Nutzung des DemoViewers finden:

- Projekte, die Beispiele enthalten, referenzieren folgende Softwarekomponente: *ITV\_DemoViewer.dll*
- Code Dateien, die Beispiele aufweisen, importieren folgenden Namensraum: `de.ITVisions.DemoViewer`
- Klassen, die Beispiele enthalten, sind mit `[Demo]` bzw. `<Demo>` annotiert
- Ausführbare Beispiele sind Instanzmethoden und besitzen keine Parameter
- Ausgaben in den Beispielen erfolgen mit `Demo.Print()` – für normale Ausgaben – und `Demo.PrintHeader()` – für hervorgehobene Ausgaben
- Die Anwendung startet den DemoViewer für C# mit:

```
new de.ITVisions.DemoViewer.Viewer(System.Reflection.Assembly.GetExecutingAssembly());
```

bzw. für Visual Basic mit:

```
Dim d As New Viewer(System.Reflection.Assembly.GetExecutingAssembly())
```

Wenn Sie Beispiele aus diesem Buch in eigene Anwendungen übernehmen wollen, müssen Sie lediglich die oben genannten Verbindungen zu dem DemoViewer entfernen, insbesondere die Ausgaben mit `Demo.Print()` und `Demo.PrintHeader()` ersetzen (siehe Tabelle).

Anwendungsform	Einfache sequenzielle Ausgabe	Ausgabe in bestimmtes Element
Konsolenanwendung	<code>Console.WriteLine(Ausgabe)</code>	Nicht möglich
ASP.NET-Web Forms-Anwendungen	<code>Response.Write(Ausgabe)</code>	<code>TextBox1.Text += Ausgabe</code>
Windows Forms-Anwendung	Nicht möglich	<code>RichTextBox1.AppendText(Ausgabe)</code>

**Tabelle V.1** Alternative Ausgabemöglichkeiten

## Warum verwenden Sie an einigen Stellen im Buch das Ein-Datei-Modell für ASPX-Seiten, obwohl man doch besser Programmcode und Gestaltung trennen sollte?

Ich greife an einigen Stellen aus Gründen der Übersichtlichkeit – aus der Sicht des Buchlesers – auf das Ein-Datei-Modell anstelle des Zwei-Dateien-Modells (Hintergrundcodemodell) zurück. Codebeispiele sind besser zu erfassen, wenn man nicht zwischen zwei Listings mit dem Auge hin- und herspringen muss.

## Ist der komplette Programmcode der Beispiele im Buch abgedruckt?

Das ist natürlich nicht möglich und wäre auch nicht sinnvoll. Der Programmcode der Fallbeispiele alleine würde – ohne eine Zeile Text – die Seitengrenze dieses Buchs sprengen. Im Buch sind meistens nur einige relevante Ausschnitte abgedruckt. Unter den Listings finden Sie meistens Hinweise auf die Stelle in den Begleitprojekten, in denen Sie den Rest lesen können.

Ich weiß, dass sich einige Leser wünschen, dass der Text in gut kommentierter Form im Buch selbst abgedruckt ist. Wenn man dies in Anbetracht der Komplexität heutiger Software tun möchte, müsste sich ein Buchautor auf triviale Beispiele beschränken. Dies wollte ich jedoch nicht. Für diejenigen, welche die Listings im Buch sehen möchten, habe ich an einigen Stellen gut kommentierte, vollständige Listings abgedruckt und mir an den übrigen Stellen sehr viel Mühe mit der Auswahl der relevanten Ausschnitte gegeben.

## Fragen zur Qualitätssicherung

### Gibt es eine Qualitätssicherung für die Inhalte in diesem Buch?

Ja. Informationen zu Verfahren und Funktionen werden grundsätzlich nicht übernommen, ohne dass sie durch mich ausprobiert wurden. Codebeispiele werden selbst erstellt und getestet. Zitierte Aussagen, die nicht mit vertretbarem Aufwand verifiziert werden konnten, werden als solche gekennzeichnet (»Microsoft sagt...« etc.).

Code wird grundsätzlich in einem geeigneten Editor erstellt, kompiliert und ausgeführt. Erst dann wird der Code per *Kopieren/Einfügen* in das Word-Dokument übernommen, aus dem das Buch entsteht. Die Eingabe von Code direkt in das Word-Dokument kommt nur bei einzelnen Bezeichnern (z.B. Klassennamen) vor. So ist sichergestellt, dass der im Buch abgedruckte Code kompilierfähig ist.

Das vorliegende Buch wurde von einem Fachlektor gelesen, bevor es in Druck ging.

### Auf welcher Plattform testen Sie die Funktionen und Codebeispiele?

Aus wirtschaftlichen Gründen ist es leider nicht möglich, die Funktionen und Codebeispiele auf *allen* Plattformen und Systemkonfigurationen zu testen. Die folgende Tabelle nennt die beiden Testplattformen. Auf der primären Testplattform wurden die Codebeispiele entwickelt und getestet sowie alle Verfahren ausprobiert. Die Codebeispiele wurden zudem auf der sekundären Testplattform verifiziert.

	Primäre Testplattform	Sekundäre Testplattform
Betriebssystem	Windows Server 7	Windows Server 2008 R2
Domänenmitglied	Ja	Nein
.NET Framework	4.0.30319	4.0.303191
Visual Studio	Ultimate 2010 Englisch mit Service Pack 1 Version 10.0.30319.1	Ultimate 2010 Englisch mit Service Pack 1 Version 10.0.30319.1
Visual Studio Extensions	ADO.NET Entity Framework Code Generation Power Pack ADO.NET POCO Entity Generator Visual Studio 2010 Productivity Power Tools Tangible T4-Editor Visualization and Modeling Feature Pack	--
Zusatzwerkzeuge	Fiddler HTTP Debugging Proxy 2.3.1 Browser Firefox, Chrome, Opera	--

**Tabelle V.2** Übersicht über die Testplattformen

### Wie kann es sein, dass trotz der Qualitätssicherungsmaßnahmen Programmbeispiele aus Ihrem Buch auf meinem Computer nicht funktionieren?

Trotz der Qualitätssicherung hat es immer wieder Fälle gegeben und wird es wohl auch immer wieder Fälle geben, in denen vorher getesteter Code auf dem Zielsystem nicht läuft (Fehler erzeugt oder nicht das gewünschte Verhalten zeigt). Dies gilt nicht nur für EDV-Bücher, sondern auch für echte Anwendungen in der Praxis.

Diese technischen Schwierigkeiten können verschiedene Ursachen haben:

- Die Ursache liegt in der Systemkonfiguration (z.B. nicht vorhandene Softwarekomponenten, Rechtebeschränkungen)
- Das Beispiel benötigt systemspezifische Angaben (z.B. Pfade, Computernamen), die nicht auf das Zielsystem angepasst wurden
- Sie verwenden eine Betriebssystemversion oder -variante, welche die Funktionalität nicht unterstützt (aus wirtschaftlichen Gründen können leider nicht alle Beispiele auf allen Plattformen getestet werden)
- Sie verwenden eine andere Version einer Software, die eine (leicht) andere Funktionalität besitzt
- Auf Ihrem System sind andere Anwendungen vorhanden, welche die Funktionalität beeinflussen (Seiteneffekte)

Außerdem ist es grundsätzlich natürlich möglich, dass Autor, Fachlektor und Testleser etwas übersehen haben.

### **Warum gibt es keine Unit Tests zu den Codebeispielen?**

Leider kann man aus wirtschaftlichen Gründen den Beispielcode in einem Buch nicht so umfangreich testen wie den Programmcode für eine echte Anwendung. Wenn man mit den Maßstäben des Software-engineerings an Beispielcode für ein Buch herangehen würde, dann müsste das Buch um ein Vielfaches teurer werden.

Dies soll und darf aber keine Entschuldigung für eine Häufung von Fehlern sein. Aus diesem Grund verwende ich verschiedene Qualitätssicherungsverfahren, die über die Anforderungen des Verlags hinausgehen und damit die Maßnahmen vieler anderer Buchautoren übertreffen.

## **Fragen zur Unterstützung nach dem Kauf**

### **Gibt es Unterstützung nach dem Kauf eines Buchs?**

Hier ist grundsätzlich zu unterscheiden – zwischen der technischen Unterstützung des Verlags und der inhaltlichen Unterstützung.

Microsoft Press bietet Ihnen Hilfe bei Fragen zur Handhabung des Buchs. Für inhaltliche Fragen stelle ich als Autor eine Website (<http://www.IT-Visions.de/Leser>) und Diskussionsforen (<http://www.IT-Visions.de/Foren>) bereit.

Bitte beachten Sie zwei Dinge:

- Die Leser-Website ist eine völlig freiwillige Leistung, für die keine Rechtspflicht besteht und die ich auch in keinsten Weise entlohnt bekomme. Bitte sehen Sie es mir nach, falls es mal etwas länger dauert.
- Ich halte es für fairer gegenüber anderen Lesern, wenn Sie Ihre Fragen in den öffentlichen Foren stellen. Bitte nutzen Sie den direkten Kontakt nur, wenn Sie Fragen haben, die nicht an die Öffentlichkeit dringen dürfen.

Ich biete auch einen kommerziellen Support für Entwicklerthemen an:

<http://www.IT-Visions.de/Produkte/Support.aspx>

**Antworten Sie auf E-Mails, in denen ein Leser eine inhaltliche Frage stellt?**

Ich antworte mit einer Standardnachricht, die besagt, dass man die Frage bitte in meinem öffentlichen Webforum (<http://www.IT-Visions.de/Foren>) stellen möge, wenn es möglich ist. Dort beantworte ich gerne inhaltliche Fragen. Der Grund meiner Abneigung gegen E-Mails als Medium für inhaltliche Fragen liegt in der Fairness, die ich von den Fragestellern gegenüber anderen Lesern erwarte. Wer eine Antwort möchte, sollte dazu bereit sein, dass andere auch einen Nutzen von der Antwort haben können. Dieses Fairnessprinzip wende ich auch bei meinen Hochschulstudenten an, was im Allgemeinen akzeptiert wird.

Leider kann ich – auch wenn Sie das Forum verwenden – nicht garantieren, dass ich alle Fragen beantworten kann. Damit ich auch noch Zeit für andere Tätigkeiten habe (Projekte und Bücher), kann ich nur einen begrenzten Zeitraum pro Woche für das Forum verwenden. In manchen Wochen bin ich aufgrund von Geschäftsreisen leider gar nicht in der Lage, das Forum zu betrachten (wenn Sie mich über ein Projekt gebucht haben, würden Sie auch nicht wollen, dass ich zwischendurch anderen Leuten innerhalb Ihres Zeitbudgets helfe).

**Wenn ich einen Fehler in dem Buch finde, möchten Sie dann, dass ich Sie darauf hinweise?**

Ja, bitte. Unbedingt! Auf der Leser-Website gibt es dazu ein Formular, das in strukturierter Form Daten abfragt. Bitte wählen Sie diesen Weg, da es mir dann einfacher fällt, die Informationen zu katalogisieren und weiterzuverarbeiten.

**Bieten Sie auch kommerzielle Unterstützung für .NET /ASP.NET?**

Ja, das ist eigentlich mein Beruf (die Tätigkeit als Buchautor ist nur ein Hobby!). Ich biete Beratung, individuelle Schulungen, Workshops sowie die Entwicklung von Prototypen und Lösungen an:

<http://www.IT-Visions.de/Produkte>