

Wolfgang Beer · Dietrich Birngruber · Hanspeter Mössenböck · Herbert Prähofer · Albrecht WöB

Die .NET-Technologie

Grundlagen und Anwendungsprogrammierung

2., aktualisierte und erweiterte Auflage

 dpunkt.verlag

Dipl.-Ing. Dr. Wolfgang Beer
wolfgang.beer@scch.at

Mag. Dr. Dietrich Birngruber
bdi@ecolog.at

Prof. Dr. Hanspeter Mössenböck
moessenboeck@ssw.uni-linz.ac.at

Dipl.-Ing. Dr. Herbert Prähofer
praehofer@ssw.jku.at

Dipl.-Ing. Albrecht Wöß
AlbrechtWoess@softwaretechnik.at

Lektorat: Christa Preisendanz
Copy-Editing: Ursula Zimpfer, Herrenberg
Satz: Hanspeter Mössenböck, Linz
Herstellung: Peter Eichler, Eberbach
Umschlaggestaltung: Helmut Kraus, www.exclam.de
Druck und Bindung: Koninklijke Wöhrmann B.V., Zutphen, Niederlande

Bibliografische Information Der Deutschen Bibliothek

Die Deutsche Bibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie;
detaillierte bibliografische Daten sind im Internet über <http://dnb.ddb.de> abrufbar.

ISBN 3-89864-421-9

2. Auflage 2006
Copyright dpunkt.verlag GmbH
Ringstraße 19b
69115 Heidelberg

Die vorliegende Publikation ist urheberrechtlich geschützt. Alle Rechte vorbehalten. Die Verwendung der Texte und Abbildungen, auch auszugsweise, ist ohne die schriftliche Zustimmung des Verlags urheberrechtswidrig und daher strafbar. Dies gilt insbesondere für die Vervielfältigung, Übersetzung oder die Verwendung in elektronischen Systemen.

Es wird darauf hingewiesen, dass die im Buch verwendeten Soft- und Hardware-Bezeichnungen sowie Markennamen und Produktbezeichnungen der jeweiligen Firmen im Allgemeinen warenzeichen-, marken- oder patentrechtlichem Schutz unterliegen.

Alle Angaben und Programme in diesem Buch wurden mit größter Sorgfalt kontrolliert. Weder Autoren noch Verlag können jedoch für Schäden haftbar gemacht werden, die in Zusammenhang mit der Verwendung dieses Buches stehen.

5 4 3 2 1 0

Vorwort zur 2. Auflage

Seit der 1. Auflage dieses Buches hat sich die .NET-Technologie rasant weiterentwickelt. In fast allen Bereichen kamen neue Konzepte hinzu, die .NET noch mächtiger machen und seine Benutzung vereinfachen. .NET 2.0 kam Ende 2005 auf den Markt und machte es nötig, dieses Buch grundlegend zu überarbeiten, um den Neuerungen gebührend Rechnung zu tragen. Wie auch in der 1. Auflage war es dabei unser Ziel, uns auf das Wesentliche zu konzentrieren und einen umfassenden Einstieg in die .NET-Technologie zu schaffen, der es dem Leser erlaubt, mit Hilfe der umfangreichen Onlinedokumentation weiter vorzudringen.

Was ist neu in .NET 2.0?

Anstatt die Neuerungen von .NET in einem eigenen Kapitel zu beschreiben, haben wir uns entschlossen, sie in die vorhandenen Kapitel einzubauen, da wir glauben, dass dies zu einem kohärenteren Bild des derzeitigen Stands der .NET-Technologie führt. Die Neuerungen umfassen vor allem folgende Bereiche:

□ *Die Sprache C# 2.0*

In C# 2.0 wurden *generische Typen und Methoden* eingeführt, mit denen man parametrisierbare Bibliotheksbausteine realisieren kann. So lassen sich zum Beispiel allgemeine Listen implementieren, aus denen man Listen von Zahlen oder Listen von Personen erzeugen kann. Ferner wurde ein Sprachkonstrukt eingeführt, mit dem man leistungsstarke *Iteratoren* für Collections und Arrays bauen kann. Als weitere Neuerungen gibt es *partielle Typen*, die es erlauben, den Quelltext einer Klasse oder eines Interface auf mehrere Dateien zu verteilen, sowie *anonyme Methoden*, die das Arbeiten mit Delegates vereinfachen. Mittlerweile arbeitet Microsoft bereits an C# 3.0. Als wesentliche Neuerung werden hier *SQL-artige Abfragen* auf Hauptspeicherdatenstrukturen möglich sein sowie *Lambda-Ausdrücke*, *Erweiterungsmethoden* und *anonyme Typen*. Dieses Buch enthält bereits eine Vorschau auf die neuen Eigenschaften von C# 3.0.

□ *Klassenbibliothek*

Die .NET-Klassenbibliothek wurde ebenfalls um generische Typen und Methoden erweitert. Es gibt nun einen eigenen Namensraum für generische Collections. Der Namensraum für Reflection wurde an generische Typen angepasst. Große Teile der Klassenbibliothek wurden überarbeitet sowie in ihrer Performanz und Benutzbarkeit verbessert. Es würde allerdings den Rahmen dieses Buches sprengen, alle neuen Namensräume zu beschreiben. Hier verweisen wir wie in der 1. Auflage auf die Onlinedokumentation von .NET.

□ *ADO.NET 2.0*

Mit der Version 2.0 wurden in ADO.NET zahlreiche Neuerungen und Verbesserungen vorgenommen. Das Grundprinzip des verbindungsorientierten und verbindungslosen Datenzugriffs blieb aber unverändert. Viele der Neuerungen wurden gezielt für den Microsoft SQL Server eingeführt und sind somit nur in Verbindung mit diesem verfügbar. Die wichtigsten Punkte, auf die wir in Kapitel 5 näher eingehen werden, sind: eine Unterstützung für die Realisierung von Programmen, die vom jeweiligen Datenanbieter weitgehend unabhängig sind, einen Verbindungszeichenfolgen-Generator, mit dem die syntaktische Korrektheit und Vollständigkeit einer Verbindungszeichenfolge bereits beim Aufbau geprüft werden kann, die asynchrone Ausführung von Datenbankoperationen, die gleichzeitige Verwendung von mehreren Datenbankcursoren für eine Verbindung und in Zusammenarbeit mit dem System.Transactions-Namensraum die Unterstützung verteilter Transaktionen.

□ *ASP.NET 2.0*

Im Bereich der Webseitenprogrammierung wurden so genannte *Master-Seiten* eingeführt, mit denen man allen Webseiten einer Applikation ein einheitliches Layout geben kann. Mit Hilfe von *Benutzerprofilen*, *Themen* und *Skins* können Benutzer das Erscheinungsbild von Webseiten ihren individuellen Wünschen anpassen. Die Authentifizierung von Benutzern beim Zugriff auf geschützte Webseiten wurde mit Hilfe von *Membership-Klassen* vereinfacht. Schließlich gibt es noch zahlreiche neue Steuerelemente, etwa zur Navigation durch Webseiten, zur Implementierung von Login-Dialogen oder zur bequemer Darstellung von Daten in Tabellen.

□ *Web-Services*

Bei Web-Services hat sich für Entwickler kaum etwas von Version 1.0 auf Version 2.0 geändert. Microsoft arbeitet jedoch an einer Programmierschnittstelle namens *Windows Communication Foundation*, um serviceorientierte Softwareentwicklung unter Windows noch besser als mit Web-Services zu unterstützen. *Windows Communication Foundation* vereinheitlicht verschiedene Technologien, wie *Web-Services*, *Enterprise Services* und *.NET-Remoting* für das Entwickeln von verteilten, serviceorientierten Systemen in einer einheitli-

chen Programmierschnittstelle. *Windows Communication Foundation* wird vermutlich Teil der nächsten .NET-Version sein. In Kapitel 7 werden wir dieses Framework kurz vorstellen.

Webseite zu diesem Buch

Die Webseite <http://dotnet.jku.at> enthält ergänzende Materialien zu diesem Buch, zum Beispiel:

- Umfangreiche Powerpoint-Foliensätze zu diversen von den Autoren gehaltenen Lehrveranstaltungen im Bereich .NET (Einführung in C# und .NET; Compilerbau mittels .NET).
- Übungsaufgaben und Musterlösungen zu den einzelnen Kapiteln dieses Buches.
- Hilfsklassen und Beispielanwendungen.
- Nützliche Verweise auf Werkzeuge, Tutorials und Informationsquellen im Bereich .NET.

Da das Internet immer stärkere Verbreitung findet, haben wir uns entschlossen, der 2. Auflage dieses Buches keine CD mehr beizulegen. Die aktuelle Version von .NET kann von <http://msdn.microsoft.com/netframework> bezogen werden.

*Wolfgang Beer, Dietrich Birngruber, Hanspeter Mössenböck,
Herbert Prähofer, Albrecht Wöß*

Linz, Juli 2006

Vorwort zur 1. Auflage

Schon wieder eine neue Softwaretechnologie? Gestern hat man uns noch erzählt, Java (oder COM, CORBA, STL, ATL, XML etc.) sei der entscheidende Durchbruch in der Softwareentwicklung. Nun kommt Microsoft und will uns .NET verkaufen? .NET revolutioniere die Entwicklung von Windows- und Web-Anwendungen. Alles werde einfacher, sicherer und bequemer als mit den »alten« Technologien (die übrigens letztes Jahr noch neu waren). Was soll man davon halten?

Das Misstrauen mancher Softwareentwickler ist verständlich, aber dieses Mal unbegründet. .NET ist tatsächlich revolutionär – zumindest in der Windows-Welt, wo es den größten Architekturwechsel seit 10 Jahren darstellt, vergleichbar etwa mit dem Wechsel von DOS zu Windows. .NET ist eine Plattform, auf der Programme, die in unterschiedlichen Sprachen geschrieben wurden, nahtlos zusammenarbeiten können. Das Internet wurde an zentraler Stelle in .NET integriert. Die Programmierung dynamischer Webseiten oder der Zugriff auf entfernte Softwaredienste (so genannte Web-Services) wird ebenso einfach wie die Entwicklung lokaler Anwendungen.

Andererseits ist .NET auch wieder *nicht* revolutionär, denn es basiert auf Ideen, die anderswo schon seit Jahren im Einsatz sind und sich bewährt haben. In gewisser Hinsicht ist .NET eine Weiterentwicklung von Java (böse Zungen nennen es sogar eine Kopie). Viele Dinge wurden aus der Java-Welt übernommen, etwa die Idee der Laufzeitumgebung mit Garbage Collection, Code-Verifikation und anderen Sicherheitsmechanismen. Auch die neue .NET-Sprache C# ist im Kern ein Java-Dialekt. Bei dynamischen Webseiten und Web-Services geht .NET jedoch weiter als Java und bietet ausgereifere Konzepte und Werkzeuge an.

Im Gegensatz zu Java, dessen Ziel es ist, *eine einheitliche* Programmiersprache (nämlich Java) unter möglichst vielen Betriebssystemen anzubieten, hat .NET zum Ziel, *viele verschiedene* Sprachen auf einem einzigen System (nämlich Windows) zu unterstützen. Den letzten Satz hört Microsoft nicht gerne. .NET kann natürlich auch unter anderen Betriebssystemen als Windows implementiert werden und es gibt auch bereits .NET-Implementierungen unter Linux und FreeBSD-Unix. Die beste Unterstützung wird .NET aber wahrscheinlich trotzdem immer unter Windows haben.

Inhalt und Ziele

Das vorliegende Buch gibt einen Überblick über die gesamte .NET-Technologie. Es führt in seine Architektur ein, beschreibt die neue Sprache C#, gibt einen Überblick über die Klassenbibliothek und behandelt dann ausführlich die Anwendung von .NET (insbesondere für Web-Applikationen) in Form von ASP.NET, ADO.NET und Web-Services.

Die .NET-Technologie ist äußerst umfangreich. Über jeden der genannten Aspekte könnte man ein eigenes Buch schreiben. Daher gehen wir nicht auf jedes Detail ein, sondern konzentrieren uns darauf, die zugrunde liegenden Konzepte so weit zu beschreiben, dass man sie versteht und in der Lage ist, durchschnittlich komplexe .NET-Programme zu entwickeln. Mit Hilfe der ausgezeichneten Onlinedokumentation (siehe beiliegende CD) sollte der Leser oder die Leserin in der Lage sein, auf eigene Faust in alle Nischen von .NET vorzustößeln.

Wer soll dieses Buch lesen?

Unser Buch spricht mehrere Leserkreise an, unter anderem:

- Studenten, die das Buch begleitend zu Lehrveranstaltungen über C#, objektorientierte Programmierung, Komponententechnologie oder Web Engineering einsetzen möchten;
- Praktiker, die sich einen Überblick über .NET verschaffen wollen, um abzuschätzen, ob diese Technologie in ihrem Arbeitsbereich Nutzen bringt;
- Java-Programmierer, die den Unterschied zwischen Java und .NET kennen lernen wollen;
- C++-Entwickler, die nach mehr Sicherheit und Komfort streben;
- COM-Entwickler, denen der Umgang mit der IDL, den GUIDs und der Registry verbesserungswürdig erscheint;
- CORBA-Entwickler, die nach einfacheren Alternativen für verteilte objektorientierte Systeme suchen;
- Web-Programmierer, denen Technologien wie ASP oder Java-Servlets und JSP zu umständlich erscheinen;
- Entwickler, die ihre Software z.B. in Sprachen wie C++, Visual Basic, Fortran, Cobol oder Eiffel geschrieben haben und sie nun mit Programmen in anderen Sprachen kombinieren wollen.

Wie soll dieses Buch gelesen werden?

Die einzelnen Kapitel des Buches sind dazu gedacht, in sequenzieller Reihenfolge gelesen zu werden, obwohl auch selektives Lesen möglich ist. Auf jeden Fall sollte man mit Kapitel 1 beginnen, das einen Überblick über .NET gibt. Auch Ka-

pitel 2 über C# ist gewissermaßen Voraussetzung, da C# in allen Beispielen der restlichen Kapitel verwendet wird. Kapitel 3 wirft einen Blick hinter die Kulissen von .NET und kann beim ersten Lesen überflogen werden. Viele der darin enthaltenen Details über die Architektur von .NET können später gezielt nachgelesen werden. Die restlichen Kapitel dieses Buches sind weitgehend voneinander unabhängig und können in beliebiger Reihenfolge studiert werden.

Wir setzen voraus, dass der Leser bereits programmieren kann, am besten in Java oder C++. Eine gewisse Vertrautheit mit HTML und der Entwicklung dynamischer Webseiten ist für Kapitel 6 nützlich, aber nicht notwendig.

Systemanforderungen

Alle Beispiele dieses Buches wurden mit der englischen Version des .NET-SDK 1.0 unter Windows 2000 und Windows XP (Professional) getestet. Dabei wurde der Microsoft Internet Information Server 5.0 und der Microsoft Internet Explorer 6.0 verwendet.

CD zu diesem Buch¹

Die beiliegende CD enthält folgende Materialien:

- ❑ *Microsoft .NET-SDK 1.0*. Umfasst das .NET-Framework mit der Common Language Runtime, der .NET-Klassenbibliothek, dem C#-Compiler, diversen Werkzeugen sowie der gesamten API-Dokumentation. Visual Studio .NET ist – weil kostenpflichtig – nicht enthalten.
- ❑ *Microsoft ASP.NET Web Matrix*. Bietet annähernd die Funktionalität von Visual Studio .NET für die ASP.NET-Entwicklung. Enthält einen WYSIWYG-Webseiten-Designer sowie Mechanismen zur Datenbankbindung.
- ❑ *.NET Webservice Studio*. Ein Werkzeug, um Methoden von Web-Services interaktiv aufzurufen und zu testen.
- ❑ *Dokumentationen und Links*. Diverse technische Dokumentationen rund um .NET sowie nützliche Links zu .NET-Sprachen und Werkzeugen.

Webseite zu diesem Buch

.NET ist neu und wird sich in den nächsten Jahren sicher noch weiterentwickeln. Daher gibt es als Ergänzung zu diesem Buch unter

<http://dotnet.jku.at>

eine Webseite, die nicht nur den Quellcode aller Beispiele aus diesem Buch enthält, sondern auch nützliche Links und Hinweise auf neue Entwicklungen.

1. Eine CD lag nur der 1. Auflage dieses Buches bei.

Danksagung

Ein Buch wie dieses entsteht mit der Hilfe vieler Personen. Wir möchten an dieser Stelle den Leuten von Microsoft in Redmond, Cambridge und Wien danken, die immer für Fragen zur Verfügung standen, allen voran *Jim Miller*, *Eric Gunner-son*, *David Stutz*, *Van Eden* und *Ralph Zeller*. Neben den anonymen vom Verlag eingesetzten Begutachtern gilt unser Dank *Christoph Ertl*, *Christian Nagel*, *Christoph Steindl*, *Rainer Stütz*, *Josef Templ*, *Thomas Thalhammer* und *Sandra Wöß*, die das Manuskript sorgfältig lasen und wertvolle Verbesserungsvorschläge beisteuerten. Ohne die Kenntnis diverser Grafikprogramme und die Geduld von *Karin Haudum* wären wohl auch manche Abbildungen in diesem Buch nicht in der gebotenen Form vorhanden. *Kurt Prünner* und *Hermann Lacheiner* halfen bei der Gestaltung der Webseite zu diesem Buch. Die Zusammenarbeit mit dem dpunkt.verlag verlief wie immer äußerst angenehm und unbürokratisch. Ein großes Dankeschön an *Christa Preisendanz*, *Ursula Zimpfer* und *Josef Hegele*.

Wolfgang Beer
Dietrich Birngruber
Hanspeter Mössenböck
Albrecht Wöß

Linz, August 2002