

# Programming Microsoft .NET

Martin Schönhacker

Martin Schönhacker

Zunächst sei gleich einmal deutlich vor diesem Buch gewarnt, denn es kann Ihre Gesundheit ernsthaft gefährden. Mit beachtlichen 1,75kg ist es nicht nur ein eindrucksvoller Wälzer, sondern disqualifiziert sich auch als Bettlektüre. Wer darunter einschläft und das Buch fallen lässt, riskiert einen Nasenbeinbruch. Andererseits hat es sicher einen positiven Einfluss auf den Bizeps, dieses Werk mehrmals pro Tag aus dem Bücherregal zu heben...

Aber nicht nur in diesem Sinn handelt es sich um schwere Lektüre. Weil der Umfang des Microsoft .NET Framework so groß ist, deckt auch das Buch eine ungewöhnlich große Anzahl teils durchaus komplexer Themenbereiche ab. Man darf sich keinesfalls erwarten, es ohne Vorkenntnisse verstehen zu können. Erfahrung in objektorientierter Programmierung (idealerweise in C#, C++ oder Java) wird ebenso vorausgesetzt wie Grundkenntnisse über Windows-Programmierung, den Einsatz von HTML, den Betrieb eines Webservers, Datenbanken und mehr.

Das mag nun je nach individuellem Wissensstand etwas demotivierend klingen, aber so schlimm ist es auch wieder nicht. Der Autor versteht es nämlich, im netten Plauderton und mit gut ausgewählten Beispielen auch schwierige Zusammenhänge durchaus verständlich darzustellen. Die Beispiele sind so kompakt wie möglich, aber auch so umfangreich wie nötig, um weder Langeweile noch Verwirrung aufkommen zu lassen.

Der erste von drei großen Abschnitten besteht aus den **Kapiteln 1-4** und beschäftigt sich mit einer klassischen „Hello, World“-Anwendung unter .NET, grundlegenden Datentypen und dem Arbeiten mit Exceptions, einem Überblick über die Klassenbibliothek des .NET Framework, sowie der Entwicklung von Programmen mit grafischer Benutzeroberfläche für Microsoft Windows (Windows Forms).

Das Microsoft .NET Framework umfasst mehr als 7.000 (!!!) Klassen, Datenstrukturen, Interfaces, Aufzählungstypen und Delegates (das sind Wrapper-Klassen für Callback-Funktionen, durch die eine starke Typprüfung ermöglicht wird). Daher können viele Dinge nur im Überblick erwähnt werden, und noch mehr fallen einfach durch den Rost. Aber es geht ja auch nicht darum, jede einzelne Komponente des .NET Framework genau vorzustellen. Für einen ersten Eindruck von der Mächtigkeit dieser riesigen Bibliothek reichen die Beispiele allemal.

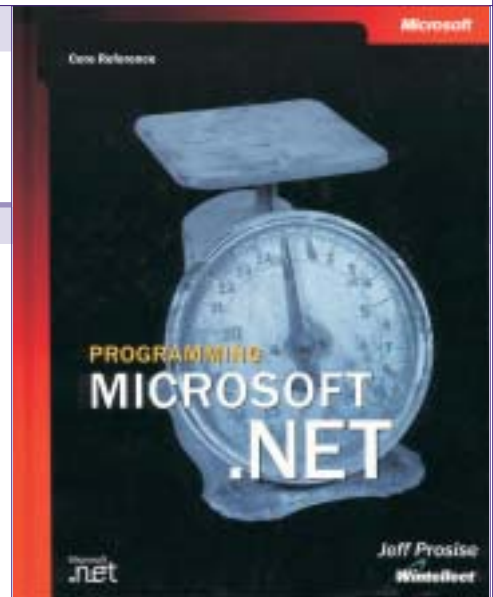
In den **Kapiteln 5-11** folgt eine Einführung in ASP.NET. Zunächst werden Web-Formulare, Web Controls, User Controls und Custom Controls erläutert und anhand praktischer Beispiele de-

monstriert. Um hier praktisch mitarbeiten und ausprobieren zu können, empfiehlt sich nach Möglichkeit die Installation des Internet Information Servers IIS. Weiter geht es mit Web-Applikationen, Sicherheitsaspekten von ASP.NET und sowohl Implementierung als auch Verwendung von XML Web Services.

Der dritte Abschnitt schließlich behandelt noch einige Spezialthemen: die Datenbankbindung mittels ADO.NET, das allgemeine Datenformat XML und seine Verwendung mit den Hilfsmitteln des .NET Framework, parallele Prozesse und ihre Synchronisation (*Multithreading*), sowie die Verteilung einer Anwendung auf mehrere Computer (*Remoting*).

Auf der beigepackten CD-ROM ist neben dem Microsoft .NET SDK und allen Beispieldateien erfreulicherweise auch das komplette Buch selbst als eBook in Form einer Windows-Hilfedatei mit ca. 5,4 MB enthalten. Dadurch ist der Inhalt nicht nur leicht zu durchsuchen, sondern z.B. auch in der Entwicklungsumgebung von Microsoft Visual Studio immer direkt verfügbar.

Wer sich durch die englische Sprache nicht abschrecken lässt, dem sei das hier besprochene Original ans Herz gelegt.



Jeff Proise; Microsoft Press; ISBN-0-7356-1376 1; 773 Seiten + CD-ROM (ca. 220 MB); ca. Euro 57,90

Aber es gibt auch eine deutsche Übersetzung unter dem Titel „Microsoft .NET Entwicklerbuch“ und mit einem kleinen Vorteil: Auf der CD-ROM befindet sich nicht nur die deutsche, sondern auch die englische Version des eBooks.

Im Textsatz haben sich an manchen Stellen unfreiwillige Knobelaufgaben in Form fehlender Leerzeichen eingeschlichen (siehe Textausschnitt), aber das kann ja auch ganz lustig sein. Abgesehen davon ist dieses Werk für einen ersten groben Einblick in das .NET Framework jedenfalls sehr zu empfehlen.

The Microsoft .NET Framework is chiefly a platform for writing Web applications and Webservices, but it supports other programming models as well. Chapter 3 spotlighted console applications and even threw in a Web application for good measure. This chapter is about Windows Forms—the programming model used to write GUI applications for the .NET Framework.

**The System.Web.Mail Namespace**

Want to send e-mail from a .NET Framework application? You could do it the hard way by using sockets to establish a connection to a mail server and then transmit a mail message using Simple Mail Transfer Protocol (SMTP). Or you could do it the easy way and rely on classes in the System.Web.Mail namespace. System.Web.Mail provides a simple managed interface to SMTP. The core classes are MailMessage, which represents e-mail messages; MailAttachment, which represents attachments; and SmtpMail, which places a friendly wrapper around the host system's SMTP mail service.

Here's how easy it is to send e-mail from a managed application:

```
using System.Web.Mail;

MailMessage message = new MailMessage ();
message.From = "webmaster@intelllect.com";
message.To = "everyone@intelllect.com";
message.Subject = "Scheduled Power Outage";
message.Body = "Our servers will be down tonight.";
SmtpMail.SmtpServer = "localhost";
SmtpMail.Send (message);
```

Having the capability to send e-mail programmatically can come in handy in countless ways. For example, you might want to send e-mail confirmations to customers who purchase merchandise from your Web site, or you might write your own software to transmit electronic newsletters to clients. Whatever your motivation, it doesn't get much easier than this.