

Was Sie erwartet und was Sie nicht erwartet

Dieses Buch sollte sich sowohl als Lehrbuch als auch als Nachschlagewerk eignen. Von daher wendet es sich an den Programmieranfänger genauso wie an den fortgeschrittenen Programmierer. Dem Anfänger oder dem Umsteiger von anderen Programmiersprachen und/oder Programmierumgebungen sei jedoch angeraten, die Buchteile A bis C wie ein Lehrbuch zu behandeln, also von vorne nach hinten durchzuarbeiten. Nach der Lektüre dieser Kapitel werden Sie bereits ein gewisses Grundwissen erlangt haben, sodass Sie sich danach bereits zu den fortgeschrittenen Programmierern rechnen dürfen. Die Kapitel der Teile D, E, F und G bauen nicht aufeinander auf. Hier können Sie ohne Weiteres querlesen, je nachdem, welche Themen Sie besonders interessieren.

Gleich vorweg möchten wir, die Autoren, zur Methodik und zur Zielsetzung insbesondere folgende Punkte erwähnen:

- Auch wenn das Buch, gemessen an der Seitenzahl, sehr umfangreich ist, legt es keinen Anspruch auf Vollständigkeit. Die Abhandlung aller Features von C++/CLI wäre unseres Erachtens ein nahezu aussichtsloses Unterfangen und würde auf Kosten der Verständlichkeit der Ausführungen gehen. Vielmehr wollten wir den Fokus gerade auf diesen Aspekt – gute Verständlichkeit – legen und haben vor allem diejenigen Dinge – wie wir annehmen, in entsprechender Ausführlichkeit – behandelt, welche für einen Visual C++-Programmierer erfahrungsgemäß von besonderer Wichtigkeit sind.
- Wir entwickeln im Buch ausschließlich Anwendungen für .NET Framework, also mit der Sprache C++/CLI. Bei C++/CLI handelt es sich um die an .NET Framework angepasste Spracherweiterung der Programmiersprache C++ und somit um eine der attraktivsten Optionen, um mit C++ moderne Windows-Anwendungen zu erstellen.
- Verweise auf andere Inhalte im Buch haben wir gerade in den ersten Buchteilen sparsam verwendet. In diesem Zusammenhang bitten wir den Leser, darauf zu vertrauen, dass er die notwendigen Informationen an den geeigneten Stellen erfährt. Im Übrigen haben wir uns um einen ausführlichen Index und um eine übersichtliche und nachvollziehbare Aufteilung der behandelten Themen bemüht.
- Der aufmerksame Leser wird im Buch an verschiedenen Stellen Wiederholungen antreffen. In der Regel sind diese durchaus beabsichtigt. Das Thema Visual C++ ist sehr komplex und ein bestimmtes Konzept lässt sich oft nicht ohne Vorgriff auf ein anderes darstellen. Ganz allgemein haben wir uns von der Tatsache leiten lassen, dass sich wiederholende Aussagen im besten Fall für das Verständnis förderlich und im ungünstigsten Fall einfach nur redundant (nichtsdestotrotz schadlos) sind. Dagegen sind unterlassene Informationen dem Verständnis immer abträglich.
- Die Sprache C++ bzw. C++/CLI lässt dem Programmierer viel Freiraum, was dazu führt, dass jeder Entwickler mehr oder weniger seinen eigenen Konventionen folgt. Wir, die Autoren, haben uns bemüht, diese zu vereinheitlichen. Sollte uns das nicht immer gelungen sein, bitten wir den Leser, uns dies nachzusehen.

Beschreibung der einzelnen Buchteile

In Teil A erhalten Sie eine kurze Einführung in die Entwicklungsumgebung Visual Studio 2010 und die Konzepte von .NET Framework. Die Ausführungen zu Visual Studio sind im Wesentlichen auch für die kostenlos erhältliche Visual C++-Entwicklungsumgebung Visual C++ 2010 Express gültig.

Teil B beinhaltet eine Art Crashkurs zum Thema Entwickeln von Windows-Anwendungen. Sie werden außerdem mit den Grundlagen der Programmierung im Allgemeinen und insbesondere mit denen der Programmiersprache C++ vertraut gemacht. In diesem Zusammenhang lernen Sie bereits einige wichtige Konzepte der objektorientierten Programmierung kennen.

In Teil C geht es dann ausschließlich um die objektorientierte Programmierung mit den Themen Klassen und Objekte, Vererbung, Schnittstellen, Methodenüberladung etc. Auch die für die Ereignisprogrammierung so wichtigen Delegaten werden hier behandelt. Bereits mit dem in Teil B erworbenen Wissen dürften Ihrer Kreativität in Bezug auf eigene Projekte nunmehr keine Grenzen gesetzt sein.

Mit Teil D beginnen diejenigen Buchteile, die vor allem auch zum Nachschlagen geeignet sind. Teil D (»Utility-Klassen«) behandelt, nach Themengebieten geordnet, spezielle Klassen, die besonders wichtig sind. Hervorgehoben seien hier vor allem die String- und die Auflistungsklassen.

Teil E widmet sich in aller Ausführlichkeit den Windows Forms, insbesondere den zur Verfügung stehenden Steuerelementen. Wenn Sie also Informationen zu einem bestimmten Steuerelement benötigen, sehen Sie am besten hier nach. Außerdem lernen Sie unter anderem, wie Sie in Ihren Programmen Grafikfunktionen einsetzen, MDI-Fenster aufrufen, Drag & Drop unterstützen und eigene Steuerelemente entwickeln.

In Teil F behandeln wir als weiterführende Techniken die Datenbankprogrammierung und die Umsetzung der parallelen Programmierung. Ein weiteres Kapitel beschäftigt sich mit der Windows Registry. Der abgesehen vom Anhang letzte Buchteil G, »Visual Studio für Fortgeschrittene«, beleuchtet spezielle Themen zur Entwicklungsumgebung.

Noch ein Wort zur Programmiersprache C++

Die Programmiersprache C++ ist aus der Programmiersprache C hervorgegangen. Sie gehört von jeher zu den Schwergewichten unter den Programmiersprachen, was seinerzeit – bis Ende der 1980er-Jahre – auch von der Programmiersprache C gesagt werden konnte. Der Erfinder von C++ ist Bjarne Stroustrup von der amerikanischen Firma AT&T.

C++ war von Anfang an gewissermaßen als erweitertes C gedacht, erweitert vor allem um objektorientierte Features. Diese und auch die allgemeine Syntax wurden anschließend von vielen anderen Programmiersprachen übernommen, in den 1990er-Jahren z.B. von Java, JavaScript, Perl, PHP – um nur einige zu nennen – und nicht zuletzt in unserem Jahrtausend von C#. Das hat für Sie als Kenner der Programmiersprache C++ unter anderem den Vorteil, dass der Aufwand eher gering ist, wenn Sie sich einmal in eine der genannten Programmiersprachen einarbeiten wollen.

Was die objektorientierten Features angeht, so lässt sich sagen, dass C++ eine gewisse Obermenge darstellt. Das heißt, viele objektorientierte Features, denen Sie bei C++ begegnen, werden Sie in der einen oder anderen Form auch in anderen objektorientierten Programmiersprachen wiederfinden.

Betrachtet man nun die originale Programmiersprache C++ als solche, so vermisst man trotz der genannten Vorteile jedoch Einiges. Als Stichworte seien genannt: visuelle Programmierung (gemeint sind Features zum Entwickeln von grafischen Benutzeroberflächen, sprich: Windows-Anwendungen), Datenbankprogrammierung, Webprogrammierung und die Unterstützung von verschiedenen neuen Technologien wie Webdiensten oder XML.

Nun, die Lösung heißt Visual C++ – das *Visual* im Namen bezieht sich, wie schon erwähnt, auf die visuelle Programmierung. Eigentlich nichts Neues, denn C++ hat seit den 1990er-Jahren schon immer von zusätzlicher Unterstützung in Form von mehr oder weniger umfangreichen Klassenbibliotheken profitiert. Das war auch in Visual Studio der 1990er-Jahre mit der MFC der Fall.

Wirklich nichts Neues? Doch! Weil die Klassenbibliothek von .NET Framework – die aktuelle Version von .NET Framework ist 4.0 – jeden Rahmen sprengt, von dem Programmierer noch vor einigen Jahren nicht einmal zu träumen wagten. Mit der Funktionalität, die Ihnen in .NET Framework zur Verfügung steht, können Sie als Entwickler praktisch jede Aufgabe lösen. Sogar der Programmierneuling kann mit dieser Unterstützung in verhältnismäßig kurzer Zeit ansehnliche Ergebnisse erzielen.

Der einzige Nachteil, der sich in Visual Studio .NET des Jahres 2003 mit der CLI-Erweiterung eingeschlichen hatte, nämlich eine etwas umständliche bzw. gewöhnungsbedürftige Syntax, wurde bereits in Visual Studio 2008 restlos beseitigt. Grund genug also, um »mit viel Spaß an der Freude« an die Sache heranzugehen. Genau das und vor allem viel Erfolg wünschen wir Ihnen bei der Lektüre dieses Buchs.