

Vorwort

In diesem Kapitel:

Für wen wurde das Buch geschrieben?	23
Word-Versionen	24
Wie ist das Buch aufgebaut?	24
Danksagung	26

Sie halten die vierte Auflage von »Word Programmierung – Das Handbuch« in der Hand. Wir sind stolz und zugleich demütig, dass das deutschsprachige Publikum diese Arbeit so sehr schätzt, um eine weitere Fortsetzung zu rechtfertigen.

Kaum zu glauben, dass wir auf gut zehn Jahre zurückblicken können. Die erste Ausgabe für Word 97-2003 befasste sich hauptsächlich mit VBA, gewürzt mit C#-»Übersetzungen« sowie ein wenig VSTO und WordProcessingML.

Seither ruhen die Entwicklung und die Erwartungen der Technologiemarkte nicht. XML und offene Standards sind das Gebot der Stunde; Mobilität wird großgeschrieben. Als Folge erhielten wir in Office 2007 die Office Open XML Datei-Formate und das »Ribbon« (zu Deutsch Multifunktionsleiste und glücklicherweise in Office 2010 in Menüband umbenannt) – und schon wurde der Einflussbereich des Objektmodells bei Anwendungserweiterungen merklich und empfindlich geschmälert.

Dieser Trend setzte sich in Office 2010 mit Einführung der Backstage-Ansicht und Einbindung der Kontextmenüs in das »Ribbon XML« fort. In Office 2013 wurden weitere Befehle in die Backstage-Ansicht voll integriert, was den traditionellen Wirkungsraum von VBA weiter einschränkt.

Damit nicht genug. Wir haben eine komplett neue Technologie für die Erweiterung von Word und Excel erhalten: die Office Apps, die auf Basis von JavaScript und HTML erstellt werden. Nicht nur die Programmiersprache, sondern auch die Fernsteuerungsmöglichkeiten ändern sich damit schlagartig: nur die gegenwärtige Markierung kann beeinflusst werden.

Dafür laufen Office Apps in den Web- sowie Tablet-PC- (Windows RT) Versionen von Office, wo die VBA- und die herkömmlichen API-Unterstützungen fehlen.

Wie sieht die Zukunft aus? Wie lange noch bleiben uns die Office-Anwendungen für den Desktop und VBA erhalten? Wir wissen es nicht und wir schätzen, auch Microsoft betrachtet gespannt die Lage. Die Marktentwicklung der nächsten Jahre wird ausschlaggebend sein. Auf jedem Fall bleiben uns alle bisherigen Technologien bis auf Weiteres erhalten. Microsoft hat im Sommer 2013 klar Stellung genommen und betrachtet die Office Apps als *Ergänzung*, nicht als Ersatz von VBA und VSTO.

Tatsache ist, dass viele der mit Office erledigten Aufgaben auf einem mobilen Gerät nicht machbar sind – dafür sind Bild- und Arbeitsfläche zu eingeschränkt.

Historisch betrachtet glauben wir, dass die herkömmlichen Erweiterungsmöglichkeiten uns noch einige Jahre und Office-Versionen erhalten bleiben. Das einst hoch bejubelte .NET Framework hat sich nicht durchgesetzt – weder in Office noch in Windows, u.a. weil eine riesige Investition in VBA-Code vorhanden ist. Auch wenn die Office Apps soweit entwickelt werden, dass sie die gleichen Aufgaben wie das gegenwärtige Objektmodell erfüllen können, wird es einige (Mann-) Jahre Arbeit und viel Geld brauchen, bis alle Werkzeuge umprogrammiert sind.

Das Objektmodell und VBA bleiben vorläufig aktuell. Gleichzeitig müssen in einem Buch für die Programmierung von Word die neuen Technologien (zunehmend) Platz finden. Da die uns zur Verfügung stehende Anzahl an Seiten bei gut 1.000 gleichbleibt, sehen wir uns auch dieses Mal gezwungen, den Inhalt der vorherigen Auflage in das Downloadarchiv zum Buch auszulagern, um für Neues Platz zu machen. Diese Informationen bleiben gleichwohl gültig und sogar wichtig, weshalb die Vorgängerauflagen des Buchs sich im PDF-Format im Downloadarchiv befinden.

Wo liegt nun der Schwerpunkt dieser Auflage?

- Die Weiterentwicklung von Inhaltssteuerelementen
- Aufgrund der zunehmenden Rolle von XML eine vertiefte Einführung in diese Technologie im Zusammenhang mit Word

- Eine Einführung in die Bearbeitung des Office Open XML-Dateiformats mit dem Open XML SDK
- Eine Einführung in die neue Office Apps-Technologie

Managed Code (.NET) Add-ins & VSTO

Word unterstützt für die Desktopausgaben von Office weiterhin auf der IDTExtensibility2-Schnittstelle basierende COM-Add-Ins. (Sie laufen nicht in den Web Apps oder auf Windows RT.) Das VSTO-Add-In übernimmt in diesem Zusammenhang eine wichtigere Rolle, weil Microsoft die Visual Studio »Shared Add-In«-Vorlage aus Visual Studio 2012/2013 entfernt hat. Wer nicht mit VSTO arbeiten will, muss das »Shared Add-In« und das dazu benötigte »Shim« von Grund auf selber schreiben oder ein Produkt eines Drittherstellers erwerben.

Word Open XML

Das Office Open XML-Dateiformat hat nochmals an Bedeutung gewonnen. Wer Dokumente für Benutzer der Web- und Windows RT-Ausgaben von Office erstellen oder bearbeiten will, kann die Aufgabe nur auf diesen Weg erfüllen.

Für wen wurde das Buch geschrieben?

Obwohl sich die Ausrichtung von Office und Word verändert, sind unsere Beweggründe noch die gleichen. Unser gemeinsames Wissen zum Thema Word-Steuerung soll in schriftlicher Form festgehalten und weitergegeben werden, um allen zu helfen, die Word ihren Bedürfnissen entsprechend anpassen wollen. Die Möglichkeiten hierzu sind weiterhin vorhanden, es gilt nur, sie zu kennen und korrekt einzusetzen.

Nach wie vor richtet sich dieses Buch an ein breites Spektrum von Lesern – so breit wie der Funktionalitätsumfang von Word. Der eine freut sich über eine »Super-Schreibmaschine«, der kaufmännische Sachbearbeiter interessiert sich für Berichte und der Redakteur will damit umfangreiche Handbücher verwalten. Anderswo in einer stillen Ecke sitzt der Autor, der darin das Werkzeug sieht, um seinen nächsten Bestseller zu verfassen. Daneben steht der professionelle Entwickler, der mit wenigen bis keinen Word-Kenntnissen diese Funktionalität den Bedürfnissen einer Großfirma anpassen muss. Dass eine Anwendung alle diese Erwartungen erfüllen kann, ist bemerkenswert. Zugegeben, einige Aufgaben sind leichter zu realisieren als andere. Die entsprechenden Werkzeuge sind jedoch vorhanden, und als Entwickler ist es unsere Aufgabe, die Vorgänge für den Benutzer zu entflechten, zu vereinfachen und so zu erklären, dass er effizienter ans Ziel gelangt, ohne sich mit den internen Einzelheiten von Word auseinandersetzen zu müssen.

Aus diesen Überlegungen heraus richtet sich das vorliegende Buch in erster Linie an folgende Anwendergruppen, die mit Word arbeiten und dessen Möglichkeiten nicht nur oberflächlich ausreizen möchten:

- Anwender, die Makros nicht nur mit dem integrierten Makrorekorder aufzeichnen, sondern selbstständig erstellen und bereits angeeignete Kenntnisse vertiefen möchten
- VBA-Programmierer, die professionelle Lösungen entwickeln und neben dem grundlegenden Wissen vertiefende Informationen erhalten möchten
- Alle Programmierer (einschließlich Microsoft .NET), die aus einer eigenen Applikation heraus auf Word zugreifen und dieses Programm automatisieren möchten oder müssen

- Programmierer, die Word-Dokumente erstellen oder bearbeiten müssen, ohne Beziehen der Word-Anwendung

Das Buch richtet sich jedoch nicht an die Anwender, die noch keine Erfahrung mit Word oder VBA (oder einer anderen Programmiersprache) haben. Denn das vorliegende Buch beinhaltet keine Anleitung zum Einstieg in VBA; für diesen Bereich sind bereits verschiedene Bücher erschienen.

Word-Versionen

Da die Word-Programmierschnittstelle weiterhin auf dem in Word 97 eingeführten Objektmodell basiert, gelten die Kernaussagen für alle seither veröffentlichten Versionen. Diese Auflage behandelt aber primär die »Ribbon«-Versionen von Word 2007 bis 2013. Alle Abbildungen wurden in Word 2013 unter Windows 7 erstellt. Im Allgemeinen hat das Entwicklerteam auf die Rückwärtskompatibilität geachtet und falls die Funktionalität in einer älteren Version von Word vorhanden ist, sollen die Codebeispiele auch in Word 97, 2000, 2002 sowie 2003 laufen.



Word
2013

Alle Codebeispiele wurden in Word 2013 getestet, falls notwendig korrigiert, aber weitgehend für gut befunden und belassen. Um gezielt Word 2013-spezifische Informationen zu finden, halten Sie bitte nach dem nebenstehend dargestellten Symbol Ausschau.



Die C#-Beispiele stehen im Visual Studio 2008-Format bereit und funktionieren wie der VBA-Code weiterhin. Die Referenzen zeigen auf die PIAs für Word 2013. Falls Sie mit 2007 oder früher arbeiten, werden Sie die Referenzen anpassen müssen (siehe Kapitel 9).

HINWEIS

C#-Entwickler, die bereits Visual Studio 2010 oder 2012 mit .NET Framework 4.0 oder höher benutzen, benötigen diese Beispiele nicht unbedingt, da .NET Framework 4.0 eine Unterstützung für optionale und benannte Parameter einführte. Wir weisen jedoch darauf hin, dass solcher Code langsamer laufen könnte.

Wie ist das Buch aufgebaut?

Dieses Buch besteht aus sechs Teilen, von denen jeder einem bestimmten Schwerpunkt gewidmet ist. Die einzelnen Teile bauen zwar aufeinander auf, stehen aber in keiner direkten Abhängigkeit zueinander. Somit ist gewährleistet, dass nicht alle Seiten gelesen werden müssen, um sich in ein spezifisches Thema zu vertiefen.

Der Inhalt der einzelnen Teile kurz zusammengefasst:

- Teil A Eine Einführung in die Welt der Makros und VBA. Er vermittelt das grundlegende Wissen zu den Möglichkeiten von Makros und stellt die Werkzeuge der eigentlichen Programmierumgebung – den Visual Basic-Editor – vor. Eine Zusammenfassung zu den VBA-Grundlagen, erste allgemeine Beispiele, eine Diskussion des Office-Sicherheitskonzepts und das Einbinden von Windows-API runden diesen Teil ab. Das Kapitel zum Thema XML vermittelt die Grundlagen einschließlich Konzepten, die für die Arbeit mit Word relevant sind, wie Schemas und die Programmierschnittstelle: das XML-Dokument-Objektmodell (DOM).
- Teil B Gilt als Nachschlagewerk zum äußerst komplexen Objektmodell von Word. Es werden jeweils die wichtigsten Eigenschaften und Methoden zu den einzelnen Objekten detailliert aufgezeigt. Anhand

von passenden Beispielen werden diese dem Leser nähergebracht. Dieser Teil enthält auch Beispiele in C#, welche die Schnittstellen zu den Word-APIs veranschaulichen. Anhand dieser Kenntnisse sollte sich der .NET-Programmierer Zugang zum gesamten Objektmodell verschaffen können.

- Teil C Zeigt die Grundlagen des Zusammenspiels mit anderen Applikationen auf. Dies ist unabhängig davon, ob Word aus anderen Applikationen heraus gesteuert wird oder ob Word selbst andere Applikationen steuert bzw. ob eingebundene Objekte in Word gesteuert werden. In diesem Teil befinden sich C#-Beispiele für die Automatisierung eingebetteter Objekte. Es werden auch die Möglichkeiten von VSTO vorgestellt.
- Teil D Eine Zusammenfassung, die aufzeigt, auf welche unterschiedlichen Arten die Benutzerschnittstelle von Word angepasst werden kann. Dazu gehört unter anderem die Verwendung der internen Dialogfelder, das Erstellen von benutzerdefinierten Dialogfeldern, das Anpassen des Menübands (in Word 2007 Multifunktionsleiste genannt) und Tastaturkombinationen.
- Teil E Eine Übersicht der neuen Office Apps-Technologie. Die Grundlagen werden vorgestellt, gefolgt von einigen einfachen Beispielen.
- Teil F Widmet sich dem Open XML-Dateiformat und zeigt dessen Einsatzgebiet und Möglichkeiten im Zusammenhang mit Word auf. Nach einer Vorstellung der Grundlagen wird der Aufbau eines Word-Dokuments erläutert. Es wird ferner aufgezeigt, wie Word-Dokumente mithilfe des Open XML SDK bearbeitet und erstellt werden.

Stichwortverzeichnisse

Am Ende des Buchs befinden sich zwei Stichwortverzeichnisse mit unterschiedlichen Schwerpunkten:

- Das eigentliche **Stichwortverzeichnis** fasst den Inhalt nach einzelnen Themen und Inhalten zusammen. Es bietet so einen schnellen Zugriff auf die inhaltlichen Stellen im Text.
- Das **Verzeichnis zum Objektmodell** fasst alle Stellen im Dokument zusammen, die einen direkten Bezug auf ein Objekt sowie dessen Eigenschaften und Methoden haben. Somit ist ein schneller Zugriff auf die einzelnen Objekte innerhalb des Texts gewährleistet.

Die Begleitdateien zum Buch

Zum Buch gehört nun ein Downloadarchiv. Alle aufgeführten Programmsequenzen werden in Form von Beispieldateien mitgeliefert. Die Dateinamen werden jeweils am Ende eines Abschnitts oder Kapitels erwähnt. Die entsprechenden Dateien befinden sich im Downloadarchiv im Ordner *\Beispiele\KapXX* (wobei *XX* für die Nummer des entsprechenden Kapitels steht).

Neben diesen Beispielen sind weitere wertvolle Informationen im Downloadarchiv enthalten. Einen entsprechenden Hinweis finden Sie in den jeweiligen Kapiteln. Die Dateien befinden sich im Downloadarchiv im Ordner *\Beilagen*.

Im Buch wird an verschiedenen Stellen auf Informationsquellen im Internet verwiesen. Die entsprechende Internetadresse (URL) ist jeweils direkt im Text mit angegeben. Damit diese zum Teil recht langen und kryptischen Zeichenfolgen nicht manuell abgetippt werden müssen, finden Sie auf der Supportseite zu diesem Buch eine ausführliche Linkliste (Dateiname *Linkliste Ms5-472.zip*), die Sie auf Ihren Rechner herunterladen können. In dieser Liste sind sämtliche Links aufgeführt, die in die-

sem Buch angegeben sind, und Sie können so einfach per Klick die Software bzw. jeweiligen Informationen herunterladen.

All diese Begleitdateien finden Sie unter

www.microsoft-press.de/support/9783866454729

und

msp.oreilly.de/support/2401/851

Weitere Informationen zu den Begleitdateien zum Buch sind im Anhang zusammengefasst.

Danksagung

Wir danken recht herzlich unserem Kollegen Markus Hahner für das Durchlesen des neuen Materials. Seine Korrekturen und Bemerkungen haben wesentlich zum Wert des Inhalts beigetragen.

Ferner danken wir unserem ehemaligen Mitautor Peter Jamieson für seine langjährige Mitarbeit bei den früheren Auflagen dieses Buchs.

Das *Autorenteam*, im Dezember 2013.