

Adrian Vogler
Thomas Wieland

Borland

C++Builder 3.0

für Profis

© Carl Hanser Verlag München Wien
Alle Rechte vorbehalten

Vorwort

Der leidenschaftliche Hacker ist eine Spezies, die immer seltener zu finden ist. Es hat heute kaum noch jemand angesichts des Drucks von Kunden und Management die Zeit, sich über jede Zeile Code Gedanken zu machen und so lange daran zu feilen, bis ein optimales Ergebnis zustande kommt. Oftmals müssen Anwendungen 'von heute auf morgen' erstellt werden, bei deren Entwicklung man sich nicht mit immer wiederkehrenden Standardproblemen herumärgern, sondern sich am liebsten ausschließlich auf die konkrete Aufgabe konzentrieren möchte.

Diese Anforderung erfüllen RAD-Systeme, mit deren Hilfe sich Applikationen schnell entwickeln lassen. Ursprünglich bauten derartige Umgebungen auf „Sprachen der vierten Generation“ (4GL) auf – in den letzten Jahren beherrschten den Markt aber hauptsächlich Microsofts Visual Basic (VB) und Borlands Delphi.

Deren Konzept baut auf objekt- beziehungsweise komponentenorientierte Programmierung, bei der eine Benutzeroberfläche aus vorhandenen Steuerelementen in kurzer Zeit mit der Maus zusammengestellt werden kann. So sind selbst Einsteiger in der Lage, schnell einen Anwendungs-Prototyp zu erstellen. Da beide Systeme auch die problemlose Anbindung an eine Reihe von Datenbanken ermöglichen, war ihnen der Erfolg sicher.

Professionelle Entwickler hingegen belächelten oft VB und Delphi: Das Basic galt in Sprachumfang und Performance als nicht ausreichend für ernsthafte Anwendungen. Delphis Object Pascal hatte diese Probleme zwar weniger, war aber immer noch eine relativ seltene und proprietäre Sprache mit Hobbyprogrammierer-Image, denn der Industriestandard ist inzwischen C++.

Auf diese und andere Bedenken hat Borland, das nun den Namen Inprise trägt, mit dem C++Builder eine Antwort gefunden und den Beweis erbracht, daß auch C++ für RAD durchaus geeignet ist. Mit der Version 3 ist dieses Produkt nun seinen Kinderschuhen entwachsen und hat in Funktionsumfang und Leistungsfähigkeit Delphi eingeholt. Es bietet eine breite Palette an Werkzeugen, die für alle Arten von Entwicklern interessant sind:

- Vom Heimanwender, der auch mal ein Windows-Programm schreiben möchte,
- über den IT-Verantwortlichen und Gelegenheitsprogrammierer, der ab und zu eine Windows-Lösung braucht, sich dafür aber nicht viel Zeit nehmen kann,
- und den Profi-Programmierer, der mit den neuesten Technologien wie ActiveX, Intranet und verteilten Datenbank-Anwendungen arbeiten möchte,
- bis hin zum Entwickler in Großbetrieben und Behörden, der Applikationen erstellen muß, die in einem heterogenen Umfeld – das heißt nicht nur Windows, sondern auch UNIX o.ä. als Betriebssystemplattform – zum Einsatz kommen sollen.

Das vorliegende Buch wendet sich vorwiegend an die zweite bis vierte Gruppe. Es bietet Ihnen den Einstieg in viele fortgeschrittene Themen der 32Bit-Windows-Programmierung mit dem C++Builder.

Borland hat mit der Umfirmierung in Inprise Corp. im Sommer 1998 seine Neuausrichtung auf geschlossene Lösungen für unternehmensweite Softwarestrukturen unterstrichen. In

dieser Hinsicht hat es auch seine Werkzeuge wie den C++Builder konsequent weiterentwickelt. Bereits die aktuelle Version "Borland C++Builder 3.0 Client/Server" verfügt über eine Vielzahl von Fähigkeiten aus diesem Bereich. Mit dem jüngsten Ausbau des Pakets zu "C++Builder 3.0 Enterprise" kam noch weitere Unterstützung für die Entwicklung verteilter Anwendungen hinzu.

Wir wollen Ihnen mit diesem Buch helfen, alle neuen und modernen Fähigkeiten der Software für Ihre Anwendungen zu nutzen. Dabei konzentrieren wir uns auf weiterführende Themen und können auf die Grundlagen der Bedienung nur am Rande eingehen. Es eignet sich aber nicht für Anfänger; Sie sollten also bereits über einige Erfahrungen in der Bedienung des C++Builder verfügen.

Dafür werden auch Grundlagen von zusätzlichen Technologien besprochen, etwa COM und CORBA für mehrschichtige Applikationen oder MIDAS und Entera. Entsprechend liegen die Schwerpunkte bei Datenbankanwendungen, bei Applikationen in verteilter Architektur mittels COM und CORBA, OLE/ActiveX-Anwendungen sowie Software für Internet und Intranet.

Daneben haben aber auch C++Builder-spezifische Themen wie Windows-Botschaften oder die Komponentenentwicklung ihren Platz. Ein Teil mit Tips & Tricks für die Programmierung rundet das ganze ab, so daß das Buch für jeden interessant ist, der sich professionell mit der Anwendungsentwicklung mit Hilfe des C++Builder beschäftigen will.

Wenn Sie Vorschläge oder Anregungen zum Buch haben, können Sie sich an den Verlag oder direkt an die Autoren unter den Email-Adressen Thomas.Wieland@bigfoot.de bzw. Adrian.Vogler@uumail.de wenden.

Wir danken der Inprise Deutschland für die Überlassung der Rezensions-Software, dem Hanser-Verlag für die gute Betreuung und unseren Familien für ihre Geduld und Nachsicht.

Allen Lesern wünschen wir viel Spaß und Erfolg mit dem C++Builder!

Inhaltsverzeichnis

TEIL I:C++BUILDER INTERN	21
1 Botschaften und ihre Verarbeitung	23
1.1 Botschafter unter Windows.....	23
1.2 WinSight: der Lauschangriff auf Botschaften	24
1.2.1 Die Fensterhierarchie	24
1.2.2 Der Botschaftsausschnitt	25
1.3 Botschaftsverarbeitung in VCL-Programmen	26
1.4 Versenden von Botschaften	28
1.4.1 Botschaftstypen.....	28
1.4.2 Versenden von Botschaften	29
1.4.3 Botschaften an VCL-Komponenten.....	30
1.5 Behandeln von Windows- und eigenen Botschaften.....	30
1.5.1 Botschaftszuordnungstabellen	30
1.5.2 Benutzerdefinierte Botschaften	32
1.6 Beispiel: Wecker mit Taskleisten-Symbol	33
1.6.1 Sinn des Programms.....	33
1.6.2 Aufbau des Formulars	33
1.6.3 Methoden für Timer und Sound.....	34
1.6.4 Botschaften für Taskleistensymbole	37
1.7 Zusammenfassung	43
2 Komponentenentwicklung	45
2.1 Bedeutung eigener Komponenten	45
2.2 Entwurf durch Ableiten aus der VCL.....	46
2.2.1 Die Klassenhierarchie	46
2.2.2 Überlegungen zur Objektorientierung.....	49
2.2.3 Design einer Komponente	52
2.3 Das Gerüst erzeugen.....	55

2.3.1	Der Komponentenexperte	55
2.3.2	Das Testprogramm	56
2.4	Eigenschaften definieren	57
2.4.1	Charakteristika von Eigenschaften	57
2.4.2	Arten von Eigenschaften	62
2.4.3	Implementierung der Laufschrift-Komponente	67
2.4.4	Zusammenfassung	72
2.5	Ereignisse festlegen	72
2.5.1	Was sind eigentlich Ereignisse?	72
2.5.2	Typen für Ereignisse	73
2.5.3	Deklaration als Eigenschaft	75
2.5.4	Eine virtuelle Methode für die Auslösung	75
2.5.5	Das Ereignis auslösen	77
2.5.6	Aufrufe von Ereignissen der Basisklasse ändern	77
2.5.7	Behandlung von Windows-Botschaften	78
2.5.8	Zusammenfassung	80
2.6	Methoden hinzufügen	80
2.7	Testen	81
2.7.1	Das Formular der Testanwendung vervollständigen	82
2.7.2	Die Komponente initialisieren	83
2.7.3	Die Eigenschaften testen	84
2.7.4	Die Ereignisse behandeln	85
2.8	Packages	86
2.8.1	Grundlagen: Entwurfszeit- und Laufzeitpackages	86
2.8.2	Ein neues Package erstellen	87
2.8.3	Ein vorhandenes Package verwenden	89
2.9	Installieren	89
2.9.1	Bitmaps für Komponenten festlegen	89
2.9.2	In ein Package installieren	91
2.10	Einbinden der Online-Hilfe	93
2.10.1	Vorgehensweise	93
2.10.2	Aufbau einer Themen-Datei	95

2.10.3	Schreiben der Übersichtsseite.....	96
2.10.4	Schreiben von Seiten für alle Klassenelemente.....	97
2.10.5	Schlüsselwörter für die kontextsensitive Hilfe	98
2.10.6	Übersetzen des Textes in eine Help-Datei	99
2.10.7	Einbinden in das C++Builder-Hilfesystem	100
2.11	Ein Systeminformationsdialog als Komponente	101
2.11.1	Dialoge in Komponenten.....	101
2.11.2	Vorgehensweise	102
2.11.3	Systeminformationen abfragen	103
2.11.4	Die Komponente erstellen	107
2.11.5	Zusammenfassung	108
2.12	Fazit	108

TEIL II: OBJEKTMODELLE, COM UND ACTIVEX 111

3	Grundlagen des Component Object Model COM	113
3.1	Einführung	113
3.2	Überblick über COM.....	114
3.3	Warum gerade COM?.....	115
3.3.1	Argumente gegen COM	115
3.3.2	Pro COM.....	116
3.3.3	Überwindung von Problemen mit DLLs	117
3.4	Details einer COM-Schnittstelle	119
3.4.1	Aufbau der Schnittstelle	119
3.4.2	Server, Co-Klassen und CLSIDs.....	121
3.4.3	Umgang mit einem COM-Objekt	124
3.5	Kommunikation zwischen Objekten	125
3.5.1	Marshaling	125
3.5.2	IDL und ODL.....	126
3.6	Anwendung von COM-Konzepten.....	127
3.6.1	OLE-Dokumente.....	127
3.6.2	OLE-Automatisierung	128

3.6.3	ActiveX-Steuerelemente	128
3.7	Typbibliotheken (type libraries)	128
3.7.1	Compiler für Typbibliotheken.....	129
3.7.2	OLEView	129
3.8	Schnittstellen zur Automatisierung.....	130
3.8.1	Die Schnittstelle IDispatch	131
3.8.2	Wozu IDispatch?	133
3.8.3	Duale Schnittstellen.....	134
3.9	Zusammenfassung.....	134
4	OLE-Automatisierungsserver	137
4.1	Einstieg in den Server	137
4.2	Die Typbibliothek erstellen.....	139
4.2.1	Eigenschaften definieren.....	139
4.2.2	Methoden hinzufügen	143
4.2.3	Weitere Objekte in der Typbibliothek.....	145
4.2.4	Aktualisieren und Registrieren.....	147
4.3	Den Zugriffscode auf die Typbibliothek verstehen	148
4.3.1	Der Header	148
4.3.2	Die C++-Datei.....	152
4.4	Den Server implementieren.....	152
4.4.1	Das Gerüst begutachten	153
4.4.2	Die Methoden schreiben	154
4.4.3	Ein Formular für den Server	156
4.5	Zusammenfassung.....	158
5	OLE-Automatisierungscontroller (Clients)	159
5.1	Überblick.....	159
5.2	Oberfläche des ListServ-Controllers.....	159
5.3	Steuerung über eine duale Schnittstelle	161
5.3.1	Vorgehensweise.....	161
5.3.2	Import einer Typbibliothek	162

5.3.3	Der ListServ-Client	163
5.4	Steuerung über eine Dispatch-Schnittstelle	166
5.4.1	Vorgehensweise	166
5.4.2	Der ListServ-Client	168
5.5	Steuerung über ein Variant-Objekt	170
5.5.1	Vorgehensweise	171
5.5.2	Der ListServ-Client	173
5.6	Fernsteuerung von Microsoft Word	176
5.6.1	Warum Word als Automatisierungsserver?.....	177
5.6.2	Die Benutzerschnittstelle von OLEWord.....	177
5.6.3	Der Aufruf	178
5.7	Zusammenfassung	184
6	ActiveX-Controls	185
6.1	Einführung und Überblick	185
6.2	ActiveX-Controls aus VCL-Komponenten.....	186
6.3	Import vorhandener ActiveX-Controls.....	188
6.4	FlashButton, ein Beispiel-Control.....	190
6.4.1	Zusätzliche Elemente in der Klassendeklaration.....	191
6.4.2	Anpassung der generierten Methoden.....	191
6.4.3	Lese- und Schreib-Methoden für Active, Zugriff auf die Komponente	191
6.4.4	Konstruktor und Destruktor	193
6.4.5	OnTimer	193
6.4.6	Registrieren eines ActiveX-Steuerelements	194
6.4.7	Testen und debuggen des ActiveX-Steuerelements.....	194
6.5	Eigenschaftenseiten	194
6.5.1	Vorgehensweise	195
6.5.2	Erstellen einer Eigenschaftenseite	195
6.6	Persistenz der Eigenschaften.....	199
6.6.1	Hintergrund.....	200
6.6.2	Vorgehensweise	201
6.7	Eigene Ereignisse definieren	202

6.8	Zusammenfassung.....	204
7	ActiveX-Formulare	205
7.1	ActiveX-Formulare in Internet und Intranet	205
7.2	Der Experte für ActiveX-Formulare.....	206
7.3	Ein Formular mit einem Chart	207
7.4	Ereignisbehandlung der Formularekomponenten	209
7.5	Distribution über das Web.....	211
7.5.1	Vorgehensweise.....	211
7.5.2	Optionen.....	212
7.5.3	Testen der Web-Seite.....	214
7.6	Zusammenfassung.....	216
8	OLE-Container	217
8.1	Die C++Builder-Komponente	217
8.1.1	Eigenschaften	217
8.1.2	Methoden	220
8.2	Oberfläche der Beispiel-Anwendung.....	222
8.2.1	Aufbau.....	222
8.2.2	Die Toolbar	225
8.3	Programmierung des OLE-Containers.....	228
8.3.1	Aktivierung des OLE-Objekts.....	228
8.3.2	Arbeit mit Dateien	228
8.3.3	Arbeit mit der Zwischenablage	233
8.3.4	Operationen am Objekt.....	236
8.3.5	Verben im Kontextmenü.....	237
8.4	Zusammenfassung.....	238
TEIL III: PROGRAMMIERUNG VON DATENBANK- UND MULTI-TIER-ANWENDUNGEN		241
9	Datenbankgrundlagen	243

9.1	Grundlagen & Tools	243
9.1.1	Datenzugriff und Datensteuerung	243
9.2	Die Borland Database Engine	250
9.2.1	Die Registerkarte KONFIGURATION	251
9.2.2	Die Registerkarte DATENBANKEN	252
9.3	Eine eigene Datenbank erstellen	255
9.3.1	Datenbankbeschreibung	255
9.4	Der Datenbankformular-Experte	259
9.4.1	Auf die Adreß-Datenbank zugreifen	259
9.4.2	Die Anwendung vervollständigen	263
9.5	Eine Einführung in SQL	274
9.5.1	Arbeiten mit dem SQL-Explorer	275
9.5.2	Aufgabe 1: Alle Felder und alle Datensätze einer Tabelle anzeigen	277
9.5.3	Aufgabe 2: Nur bestimmte Felder aus einer Tabelle anzeigen	278
9.5.4	Aufgabe 3: Bestimmte Felder zu bestimmten Datensätzen anzeigen	279
9.5.5	Aufgabe 4: Mehrere Bedingungen eingeben	280
9.5.6	Aufgabe 5: Der logische Operator OR	281
9.5.7	Aufgabe 7: Abfragen über mehrere Tabellen	282
9.6	Der Einsatz der TQuery-Komponente	284
9.6.1	Umgang mit TQuery	284
9.6.2	TQuery im Vergleich zu TTable	286
9.7	Adreßetiketten drucken	288
9.7.1	Umgestaltung des Hauptformulars	288
9.7.2	Die auszudruckenden Datensätze per TQuery selektieren	289
9.7.3	Die QuickReport-Etiketten verwenden	290
9.8	Zusammenfassung	292
10	Client/Server-Programmierung	295
10.1	Einführung	295
10.2	Arbeiten mit Abfragen	295
10.2.1	Einfache Abfragen erstellen	296
10.2.2	Dynamische Abfragen erstellen	298

10.2.3	Abfragen verknüpfen	303
10.3	Arbeiten mit Stored Procedures	309
10.3.1	Was sind Stored Procedures?	309
10.3.2	Stored Procedures anzeigen	309
10.3.3	Stored Procedures erfassen	309
10.3.4	Stored Procedures einsetzen.....	310
10.4	Arbeiten mit der Database-Komponente.....	313
10.5	Zusammenfassung.....	314
11	Applikationen erstellen mit MIDAS	315
11.1	Mehrschichtige Applikationen.....	315
11.1.1	Single-tier-Applikationen	315
11.1.2	Two-tier-Applikationen	316
11.1.3	Multi-tier-Applikationen.....	317
11.2	Einführung in MIDAS.....	319
11.2.1	Vorteile von MIDAS	320
11.2.2	Bestandteile von MIDAS	322
11.3	MIDAS-Applikationen erstellen.....	327
11.3.1	Ein MIDAS-Beispiel ausführen	328
11.3.2	Eine Two-tier-Anwendungen konvertieren	335
11.3.3	Eine neue MIDAS-Anwendung erstellen	338
11.3.4	MIDAS-Sockets-Verbindung	345
11.3.5	MIDAS-DCOM-Verbindung.....	348
11.3.6	MIDAS-OLEnterprise-Verbindung	349
11.4	Intranet- und Internet-Applikationen	353
11.4.1	ActiveForm-Client.....	353
11.4.2	Web-Server-Erweiterungen.....	361
11.5	Zusammenfassung und Ausblick	365
12	Erstellung von DCE-basierten Multi-tier- Applikationen	367
12.1	Einführung	367

12.2	DCE QuickStart.....	367
12.2.1	DCE-Komponenten und -Dienste	368
12.3	DCE-Anwendungen und Entera	371
12.3.1	Was ist Entera?	371
12.3.2	DCE made easy?	372
12.3.3	Tools im Bereich der Programmentwicklung.....	372
12.3.4	Tools zur Programmlaufzeit	375
12.4	Die C++Builder-Enterprise-Version.....	376
12.4.1	Die Zielgruppe	376
12.4.2	Die Programm-Architektur.....	376
12.5	Eine Entera-Applikation erstellen	378
12.5.1	Entera konfigurieren.....	378
12.5.2	Die Schnittstelle festlegen	382
12.5.3	Den C++Builder-Client erstellen.....	385
12.5.4	Den Anwendungsserver erstellen.....	393
12.5.5	Die Anwendung nach Entera 4 portieren	398
12.6	Eine Datenbank-Applikation erstellen.....	404
12.6.1	Den Anwendungsserver erstellen.....	404
12.6.2	Die ODBC-Schnittstelle konfigurieren	406
12.6.3	Den Applikationsserver erstellen.....	407
12.6.4	Den C++Builder-Client erstellen.....	412
13	Erstellung von C ORBA-basierten Multi-tier- Applikationen	415
13.1	Einführung.....	415
13.2	CORBA QuickStart	415
13.2.1	Die Schnittstelle der Objekte.....	418
13.2.2	IIOB – Internet Inter-ORB Protocol.....	418
13.2.3	Applikationserstellung in der Theorie.....	419
13.3	DCOM und CORBA im Vergleich.....	419
13.3.1	Historie von CORBA und DCOM	420
13.3.2	Aktion vs. Reaktion.....	422

13.4	CORBA und der VisiBroker.....	423
13.4.1	VisiBroker ORB	423
13.4.2	Der VisiBroker Naming Service	423
13.4.3	Der VisiBroker Event Service.....	423
13.4.4	Der VisiBroker Integrated Transaction Service.....	424
13.5	Eine einfache CORBA-Applikation.....	424
13.5.1	Der Smart Agent des VisiBroker.....	425
13.5.2	Die Beispiel-Applikation	427
13.5.3	Die IDL-Schnittstelle.....	427
13.5.4	Der CORBA-Server.....	430
13.5.5	Der CORBA-Client	432
13.5.6	Der VisiBroker Object Activation Demon	434
13.6	Ein Ausblick – CORBA Integration in C++Builder 4	436
13.6.1	Eine Datenbank-Applikation erstellen.....	436
13.7	Zusammenfassung.....	446

TEIL IV: TIPS & TRICKS 449

14	Benutzeroberfläche	451
14.1	Arbeiten mit der Zwischenablage.....	451
14.1.1	Kopieren und Einfügen von Text	451
14.1.2	Abfrage von Formaten	452
14.1.3	Ein Dialog für „Inhalte einfügen“	453
14.2	Mehrzeilige Eingabefelder	454
14.2.1	Zugriff auf den Text	454
14.2.2	Positionieren des Textcursors.....	455
14.2.3	Rückgängig (Undo)	456
14.2.4	Ausschneiden, Kopieren und Einfügen	456
14.2.5	Suchen und Ersetzen über Dialoge.....	457
14.3	Die Multidokument-Schnittstelle (MDI).....	461
14.3.1	Tochterfenster anordnen	462
14.3.2	Tochterfenster endgültig schließen	462

14.3.3	Menüs fusionieren	462
14.3.4	Liste der MDI-Fenster im Menü	463
14.3.5	Ein Bitmap als Hintergrund im Rahmenfenster anzeigen.....	465
14.4	Vermischtes	467
14.4.1	Den Leerlauf für Aktualisierung der Anzeige nutzen.....	467
14.4.2	Ein Fenster ohne Titelleiste oder in beliebiger Form	469
14.4.3	Ein Info-Fenster beim Start kurz einblenden.....	471
14.4.4	Ein Werkzeugfenster.....	472
14.4.5	Metafiles	474
15	Systemroutinen	477
15.1	Datum und Uhrzeit	477
15.1.1	Bestimmung der Systemzeit	477
15.1.2	Bestimmung der Zeit über Netzwerk	479
15.1.3	Arbeiten mit der TDateTime-Klasse.....	480
15.1.4	Test auf Schaltjahr	482
15.1.5	Die DateTimePicker-Komponente.....	482
15.2	Dateien und Verzeichnisse	484
15.2.1	Laufwerkstyp bestimmen	484
15.2.2	Gesamten und freien Platz auf einem Laufwerk bestimmen.....	485
15.2.3	Einen Verzeichnisast anlegen.....	485
15.2.4	Das aktuelle Verzeichnis wechseln.....	486
15.2.5	Einen Pfadnamen zurechtstutzen	486
15.2.6	Einen Verzeichnisbaum anzeigen.....	487
15.3	Start anderer Prozesse.....	489
15.3.1	Asynchrones Starten mit ShellExecute	490
15.3.2	Ein Link ins Internet	491
15.3.3	Synchrones Starten mit CreateProcess	492
15.3.4	Warten auf den anderen Prozeß.....	494
15.4	Vermischtes	495
15.4.1	Kommandozeilenargumente auswerten	495
15.4.2	Windows herunterfahren	496

16	Internet-Clients	499
16.1	Ein WebBrowser	499
16.1.1	Die Komponente THTML	500
16.1.2	Der Microsoft Internet Explorer als Komponente.....	505
16.2	FTP-Client mit Verzeichnisbaum und Listenfenster	510
16.2.1	Kurzfassung der Bedienung der FTP-Komponente.....	511
16.2.2	Die Oberfläche	513
16.2.3	Verbindungsaufbau und Datenübertragung	515
16.2.4	Auftrennung der Verzeichniseinträge.....	521
16.2.5	Verzeichnisbaum und Dateiliste.....	522
16.2.6	Download und Öffnen per Doppelklick.....	533
16.2.7	Zusammenfassung	535
16.3	Weitere Internet-Komponenten	535
16.3.1	TNMPOP3 für den Email-Empfang	536
16.3.2	TNMSMTP für den Email-Versand	539
16.3.3	TNMNNTP für den Umgang mit Online-Foren.....	542
16.4	Zusammenfassung.....	549
17	Algorithmen	551
17.1	Suchen in Texten.....	551
17.2	Lösen von gewöhnlichen Differentialgleichungen	553
17.2.1	Mathematischer Hintergrund	553
17.2.2	Implementation des Runge-Kutta-Verfahrens	555
17.2.3	Das Wachstum-Nährstoff-Modell als Beispiel-Programm.....	558
17.2.4	Zusammenfassung	562
Index	563