

# Functional C

Martin Weissenböck

Pieter Hartel, Henk Muller, *Functional C*, Addison Wesley, Harlow, England, 1997, ISBN 0-201-41950-4, 432 Seiten, englisch.

Der Begriff „Functional Programming“ bezeichnet ein Programm-Paradigma, bei dem nicht die technische Realisierung von Programmkonstrukten die Sprache und die Notation bestimmen, sondern der logische (funktionelle) Ablauf eines Programms.

Ein Beispiel, zum besseren Verständnis in Pascal formuliert:

Zahlen werden bis zum Ende der Datei eingelesen und dabei addiert. Eine funktionelle Darstellung wäre:

- Setze die Summe auf Null.
- Solange das Datenende nicht erreicht ist,
- lies eine Zahl und addiere sie zur Summe

Nicht-funktionell sah das in alten Pascal-Versionen so aus:

```
sum := 0;
read (z);
while ioresult=0 do
begin
  sum := sum + z;
  read (zahl)
end
```

und erst in neueren Version wurde daraus

```
sum := 0;
while true do
begin
  read (z);
  if ioresult=0 then break;
  sum := sum + z
end
```

Manche Programmiersprachen kommen der verbalen Programmbeschreibung sehr nahe, sind dafür aber nicht so sehr verbreitet. Das Buch geht deshalb davon aus, daß an den Informatik-Abteilungen der Universitäten üblicherweise eine funktionale Sprache als erste Programmiersprache unterrichtet wird, zum Beispiel SML. (Im Anhang ist eine kurze Einführung in SML enthalten.)

Anschließend sollen die Studenten weitere Paradigmen kennenlernen. Ziel des Buches ist der Weg von der funktionalen Programmierung in SML

hin zur imperativen Programmierung in C. Somit wird hier weder ein Programmierkurs noch eine Anleitung für das Lösen von Problem angeboten, wohl aber werden die Prinzipien der Programmierung und die Methodik der Problemlösung verfeinert. Darüber hinaus wird auch der Unterschied zwischen imperativer und funktionaler Programmierung vermittelt und außerdem Syntax und Semantik von ISO-C. (So sind im Anhang auch alle Syntax-Diagramme von ISO-C enthalten.) Da außerdem die für die Entwicklung von Programmen notwendige Abstraktion immer wieder betont wird, zeigt das Buch den Weg zu gutem *software engineering*.

Wer sich mit den Beispielen auseinandersetzen möchte, findet sie unter <http://www.awl-he.com/computing/titles/0-201-41950-5.html>.

Zu jedem Kapitel gibt es eine Reihe von Übungsaufgaben samt Lösungsvorschlägen im Anhang.

## Programmieren unter Windows NT 4.0

# Wo geht's hier zu Windows?

Robert P. Michelic

Michael Rößmann; *Applikationen entwickeln unter Windows NT 4.0*; 672 Seiten, viele Abbildungen und Programmbeispiele (in C); Addison-Wesley BackOffice-Bibliothek; ISBN 3-8273-1080-6; AT\$ 656,-

Was interessiert einen eingefleischten Pascal-Programmierer, wenn er ein Buch über Programmierung unter Windows in die Hand bekommt? In etwa: "Ist es lesbar?"

Die Antwort ist einfach: "Es ist!" Und zwar trotz der zwangsläufigen C-Diktion in den vielen Beispielen, an die sich auch Pascal-Programmierer gewöhnen müssen. Erfreulich an diesem Buch ist der Text zwischen den Beispielen, der - für Bücher dieser Art eher unüblich - wirklich erklärt. Freilich wird beim Leser eine gewisse Erfahrung mit Programmieren allgemein und speziell

unter Windows vorausgesetzt, insgesamt aber bietet das Buch auch weniger erfahrenen Windows-Programmierern viel Information, verständlich aufbereitet.

Die Abschnitte über die grafische Benutzerschnittstelle enthalten in übersichtlicher Form alle Begriffe, Typen und Funktionen erläutert, die einem Programmierer bei der Gestaltung eines Windows-Programmes unterkommen. Da in diesem Zusammenhang sehr viel von früheren Windows-Versionen (Windows 3.XX, Windows 95) Ererbtes vorkommt, wäre es manchmal hilfreich, die Unterschiede bzw. Neuerungen besonders hervorzuheben.

Besonders interessant für diejenigen, die zwar Erfahrung mit der Windows-Programmierung, nicht aber mit Windows NT 4.0 haben, sind die Abschnitte über Prozesse und Threads. Selbst wenn man nicht gerade vorhat, ein Programm mit mehreren Threads zu entwickeln, ist es doch aufschlussreich, das dahinterstehende Konzept und die Schlüsselmechanismen einmal präsentiert zu bekommen.

Die Abschnitte über Remote Procedure Calls habe ich mehr aus theoretischem Interesse gelesen, betrachte sie aber als verständliche Einführung in die Technologie verteilter Anwendungen.

Weitere Abschnitte zu Schlüsselthe-



men in Windows NT (Dienste, Registrierdatenbank, DLLs) runden das Buch ab. Insgesamt erhält man also viel Information zum Arbeiten unter Windows NT, sodass ich das Buch sowohl als Nachschlagewerk wie auch als Einführung empfehlen kann.