

---

# MikroComputer-Praxis

---

Herausgegeben von

Dr. L. H. Klingen, Bonn, Prof. Dr. K. Menzel, Schwäbisch Gmünd

Prof. Dr. W. Stucky, Karlsruhe

## **FRAMEWORK-Praxis für kaufmännische Berufe**

### **Band 2: Modelle auf Programmebene**

Von Dipl.-Wirtsch.-Ing. Claus Kühlewein, Karlsruhe

und Dipl.-Handelslehrer Karl Nüßle, Saulgau



Springer Fachmedien Wiesbaden GmbH 1990

FRAMEWORK ist ein eingetragenes Warenzeichen der Firma Ashton Tate, Frankfurt/Main.

CIP-Titelaufnahme der Deutschen Bibliothek

**Kühlewein, Claus:**

FRAMEWORK-Praxis für kaufmännische Berufe / von Claus

Kühlewein u. Karl Nüssle.

(MikroComputer-Praxis)

NE: Nüssle, Karl:

Bd. 2. Modelle auf Programmebene. – 1990

ISBN 978-3-519-09337-4

ISBN 978-3-663-12203-6 (eBook)

DOI 10.1007/978-3-663-12203-6

Das Werk einschließlich aller seiner Teile ist urheberrechtlich geschützt. Jede Verwertung außerhalb der engen Grenzen des Urheberrechtsgesetzes ist ohne Zustimmung des Verlages unzulässig und strafbar. Das gilt besonders für Vervielfältigungen, Übersetzungen, Mikroverfilmungen und die Einspeicherung und Verarbeitung in elektronischen Systemen.

© Springer Fachmedien Wiesbaden 1990

Ursprünglich erschienen bei B.G. Teubner Stuttgart 1990

Gesamtherstellung: Druckhaus Beltz, Hemsbach/Bergstraße

Einband: P.P.K, S-Konzepte T. Koch, Ostfildern/Stuttgart

## Vorwort

Kaufmännische Anwendungen auf einem Personalcomputer mit Hilfe von Standardsoftware? Ist dieser Ansatz für die heutigen Anforderungen akzeptabel? Die beiden Autoren haben bereits im ersten Band den Nachweis geliefert, daß viele der routinemäßigen Berechnungen, die in kaufmännischen Berufen notwendig sind, mit relativ geringem Aufwand auf der Ebene eines Personalcomputers und einer integrierten Standardsoftware wie Framework erledigt werden können.

Der Personalcomputer ist aus den kaufmännischen Anwendungen heute nicht mehr wegzudenken. Er stellt immer mehr die kostengünstige Basis jeder elektronischen Datenverarbeitung dar. Insbesondere macht er den ausgebildeten EDV-Programmierer entbehrlich und liefert den Zugang zur schnellen und effektiven Informationsverarbeitung für jedermann/frau. Nur so ist es überhaupt möglich, die notwendige Breite der Anwendungen bei fehlenden EDV-Spezialisten zu gewinnen.

Während der erste Band bis auf einfache Makrostrukturen ganz ohne einen eigenen Programmieraufwand auskommen konnte, erweitert der hier vorgelegte Band nun den Horizont in den Bereich der direkten Anwendung in einer integrierten Programmierumgebung. Was hat man darunter zu verstehen? Framework ist zunächst das Beispiel für die wirkungsvolle Verbindung einer Textverarbeitung, Tabellenkalkulation, Business-Grafik und Dateiverwaltung in einer gemeinsamen Daten- und Konzeptumgebung.

Framework enthält als integriertes Standardpaket aber zusätzlich noch eine Programmiersprache FRED (Framework-Editor), mit dem sich die genannten Standardbausteine für sich allein und untereinander äußerst wirkungsvoll ausbauen lassen. Diese Möglichkeit wird nicht generell erforderlich sein. Das bedeutet, daß nicht jeder Anwender von Framework individuell durch die Hintertür wieder zum Programmierer werden muß.

Für den routinemäßigen, d.h. ständigen Einsatz eines integrierten Systems wie Framework bietet der zusätzliche Programmierbaustein jedoch die Möglichkeit, für Benutzergruppen, wie sie etwa im Bürobereich auftreten, einheitliche und automatisierte Arbeitsumgebungen zu schaffen. Ständig wiederkehrende Arbeitsvorgänge können so für unterschiedliche Anwender auf einheitliche Weise organisiert werden.

Die Programmierung unter Framework gestattet u. a. den einfachen und fehlerarmen Übergang zu anderen Standardsystemen wie dBASE oder MULTIPLAN. Auch der Datenex- und-import in Spezialsoftware wie Finanzbuchhaltung oder Personalverwaltung läßt sich damit elegant bewerkstelligen. Nicht zuletzt lassen sich unter FRED eigene Menüs entwickeln, um standardisierte Arbeitsumgebungen für die eigenen Anwendungen einsetzen zu können.

Nach den Benutzerhinweisen in Kapitel 1 werden dem Programmierneuling die gängigen Programmierstrukturen anhand einfacher Anwendungen dargelegt, um sich in die Grundlagen einer strukturierten Arbeitsweise hineinzufinden. Dem Einsteiger ist anzuraten, die angebotenen Beispiele konsequent durchzuarbeiten, um zu einem klaren Programmierstil zu kommen. Sorgfalt lohnt sich hier ganz besonders.

Im Kapitel 3 werden dann die Besonderheiten von FRED vorgestellt. Auch der erfahrene Programmierer muß sich auf die recht eleganten Aspekte der FRED-Programmierung und das Zusammenspiel mit den Standardoptionen von Framework erst einstellen, um davon optimal profitieren zu können. Kapitel 4 erläutert die externen Datenschnittstellen insbesondere zum Datenbanksystem dBASE. Auch zu

anderen gängigen Standardsystemen bietet Framework wesentlich mehr als den bloßen Transfer über einen reinen ASCII-Code.

Im Kapitel KFZ-Verwaltung geht es nun mit dem Übergang von der Kommandoebene zur Programmebene in einem großen Beispiel richtig in die Vollen. In einer kompletten Fuhrparkverwaltung wird das automatische Zusammenspiel von Berichtstexten, Auswertungstabellen, grafischen Übersichten und Datenbanken der Fahrzeuge und Fahrer entwickelt. Wie auch bei den anderen Beispielen ist das komplette Modell auf der zugehörigen Diskette verfügbar.

Der Leser hat die Möglichkeit, den Modellaufbau schrittweise nachzuvollziehen und auch eigene Modifikationen und Ergänzungen vorzunehmen. Die Programmumgebung läßt sich später dann auch auf andere Anwendungen übertragen. Die in diesem Projekt entwickelten Programmstandards sind vom speziellen Inhalt der Datenbanken unabhängig, d. h. auch auf andere kaufmännische Aufgaben übertragbar.

Das sechste Kapitel Finanzplanung bietet dem Leser wiederum nicht nur eine komfortable Abwicklung für die Standardmethoden der Abschreibung und Darlehenstilgung an, sondern entwickelt ganz allgemeine Programmmethoden zum automatischen Laden von Programmen über eine eigene Menütechnik. Damit gewinnt der Leser einen universellen Ansatz, um eigene Anwendungen unabhängig von deren speziellem Inhalt in eine automatisierte Arbeitsumgebung umzusetzen.

Was gewinnt der Anwender mit den in diesem Band dargelegten Programmiermodellen unter einem integrierten System wie Framework? Gegenüber den sonstigen Programmierstandards kann er/sie auf die mächtigen Grundbausteine von Framework aufbauen. Fragen der oft mühsamen Eingaberoutinen oder der Darstellung in Tabellen- und Datenbankschemata kann er/sie von vornherein völlig vergessen, sie sind im Grundsystem fehlerfrei verfügbar.

Aus der Sicht der reinen Kommandoebene von Framework bieten sich deutliche Verbesserungen für den routinemäßigen Einsatz an. Es geht dabei nicht nur darum, ständig wiederkehrende Arbeits- d. h. Kommandovorgänge in einem übergeordneten FRED-Programm automatisieren zu können, sondern aus der statischen Kommandoebene in die dynamische Programmebene aufzusteigen. Damit wird nicht nur kostbare Zeit eingespart, sondern die Fehleranfälligkeit langer Kommandoabläufe vermieden.

Es sei nochmals unterstrichen, daß der Programmierhintergrund nicht Sache des einzelnen Benutzers sein kann. Im Prinzip soll der direkte Anwender bei seiner laufenden Arbeit von den übergeordneten und übergreifenden Programmteilen gar nichts bemerken. Die hier vorgestellte FRED-Programmierung kann vielmehr die Arbeit der einzelnen Benutzer in einen komfortablen und automatisierten Rahmen einbetten.

Zum Abschluß sei noch angemerkt, daß die hier unter Framework entwickelten Methoden für die beruflichen Schulen weitgehende Möglichkeiten anbieten. Für die Grundausbildung aller Schüler lassen die Standardbausteine eines integrierten Systems die Konzentration auf die kaufmännischen Inhalte ohne einen Zwang zum ständigen Wechsel zwischen verschiedenen Benutzeroberflächen und Kommandoebenen zu. Für die weitergehende strukturelle Ausbildung im Rahmen einer modernen elektronischen Datenverarbeitung lassen sich dann in der gleichen Softwareumgebung Programmiermethoden zwanglos einführen.

Schwäbisch Gmünd, im Sommer 1990

Dr. Klaus Menzel für die Herausgeber der MCP-Reihe

# Inhaltsverzeichnis

<b>1. Benutzerhinweise</b> .....	7
1.1 Inhaltliche Konzeption .....	8
1.2 Anwendungsdisketten .....	8
1.3 Programmübersicht .....	9
1.4 Sonstiges .....	9
<b>2. Programmstrukturen</b> .....	10
2.1 Lineare Struktur .....	11
2.2 Zyklische Struktur .....	32
2.3 Alternativstruktur .....	50
2.4 Unterablaufstruktur .....	66
<b>3. FRED's Besonderheiten</b> .....	80
3.1 Dynamische Formeln .....	81
3.2 Makros.....	100
3.3 Tastaturfilter.....	134
<b>4. Datenaustausch</b> .....	147
4.1 Allgemeines .....	148
4.2 Multiplan .....	151
4.3 dBASE .....	167
<b>5. KFZ-Verwaltung</b> .....	197
5.1 Allgemeines .....	198
5.2 Fuhrpark-Verwaltung .....	199
5.3 KFZ-Kosten-Abrechnung .....	210
<b>6. Finanzierung</b> .....	217
6.1 Automatisches Laden von Programmen mit Menütechnik .....	218
6.2 Abschreibung .....	224
6.3 Darlehenstilgung ohne Tastaturfilter .....	232
6.4 Darlehenstilgung mit Tastaturfilter .....	236
6.5 Investition .....	240
Stichwortverzeichnis .....	245