

Zusammenfassung

Nachdem vor einigen Jahren für Personal Computer kaum Software, insbesondere Compiler, auf dem Markt waren, bieten heute viele Softwarehäuser vielfältige Programmierwerkzeuge an. Es handelt sich dabei um Softwaretools, z.B. zur automatischen Erstellung von Bildschirmmasken, zum interaktiven Testen von Programmen, zur Zugriffsoptimierung von Datenbeständen etc. Bei zwei Produkten, dem Display-Manager und dem Access-Manager von Digital-Research, sind dabei in Verbindung mit dem PL/1 Compiler der gleichen Firma umfangreiche Erfahrungen in der Programmentwicklung gewonnen worden.

Abstract

Some years ago hardly any software, especially softwaretools were offered for personal computers. Nowadays many software-rehouses are selling them. Softwaretools can be used for automatical creating of screenforms, for interactive testing of programs or access-optimization of datasets. So far we have gained a lot of experiences in program development with two products, the Display-Manager and the Access-Manager by Digital-Research, in connection with the PL/1 compiler.

Ein Großteil der Arbeit bei der Erstellung von Anwendungsprogrammen ist das Codieren von Bildschirmmasken und der Datenaustausch auf Datenträger. Lange bevor es Personal Computer gab, erstellten Programmierer Bibliotheken, in denen immer wieder benötigte Module wie zum Beispiel zur:

- Erstellung von Bildschirmmasken,
- Steuerung des Cursors im Bildschirm,
- Farbunterstützung am Bildschirm,
- Speicherung von Datensätzen in Index-Files

gesammelt wurden.

Diese Sammlungen und Überlegungen mündeten in eine Reihe von Datenbanksystemen und Softwarewerkzeugen für Großrechner. So gibt es zum Beispiel für das Datenbanksystem IMS von IBM das Hilfsprogramm MFS (Message Format Service) zur leichteren Erstellung von Bildschirmmasken und zum Datenaustausch zwischen Anwendungsprogramm und Terminal.

Mit der Einführung leistungsfähigerer Personal Computer wurden Softwarewerkzeuge angeboten, die ähnliche Aufgaben auch für den PC durchführen.

## 20.2 Softwarewerkzeuge für Personal Computer

Der IBM-PC kam in der BR Deutschland 1983 auf den Markt. In der ersten Zeit war die Programmiersoftware für die PC's sehr dürftig. So gab es für die gängigen Programmiersprachen wie FORTRAN oder COBOL nur wenige und in ihrem Leistungsumfang sehr eingeschränkte Compiler. Für PL/1 war lange Zeit kein Compiler verfügbar. Diesen brachte Digital Research erst sehr spät auf den Markt.

In der letzten Zeit wird das Angebot an Compilern und Softwarewerkzeugen zunehmend unübersehbar. Für alle gängigen Programmiersprachen wie

- COBOL,
- FORTRAN,
- PASCAL,
- C,
- BASIC,
- PROLOG,
- LISP,

sind heute Compiler verfügbar.

Mit folgenden Softwarewerkzeugen ist in der Programmentwicklung bei uns gearbeitet worden:

- - Display-Manager und Access-Manager von Digital Research für die Programmiersprachen PL/1, Fortran, BASIC und C von Digital Research.
- - Turbo Machine von Lauer & Wallwitz für Turbopascal von Borland International.
- - FABS-PLUS und AUTOSORT/96 von Computer Control System für - PASCAL, MS-FORTRAN, MS-COBOL und MS-BASIC.
- Pascal Routinen für MS-Pascal von Dieter Ludwigs.

Die angebotenen Werkzeuge sind dabei sehr unterschiedlich. Dies äußert sich besonders in ihrer speziellen Leistungsfähigkeit. Während Display- und Access-Manager von Digital Research in der Handhabung noch sehr an die Verhältnisse des Großrechners orientiert sind, unterstützt zum Beispiel die Turbo-Machine von Lauer & Wallwitz Befehle, die in das Betriebssystem DOS des Personal Computer eingreifen.

### 20.3 Erfahrungen mit Display- und Access-Manager

Von der Idee sind die beiden Softwarewerkzeuge Display- und Access-Manager von Digital Research richtungsweisend. Die beiden Werkzeuge erlauben den Datenaustausch und die Bildschirmformatierung unabhängig vom verwendeten Compiler.

In der Anwendung erinnert der Display-Manager an die Programmierung unter IMS am Großrechner. Der Anwender entwirft unabhängig vom Programm seine Bildschirmmasken mit Hilfe eines Dialogprogrammes und speichert die Masken in einem sogenannten Formatfile ab. Das Anwendungsprogramm greift auf den Formatfile zu, liest die Maske und gibt diese auf den Bildschirm aus. Den Aufbau der Masken erledigt ein Run-Time Modul, das beim Linkvorgang an das Anwendungsprogramm gebunden wird.

Der Zugriff auf den Display-File ist von mehreren Sprachen von Digital Research möglich. Dadurch können einmal erstellte Bildschirmmasken sehr schnell von einem Programm zum anderen übertragen werden.

In der Anwendung ist der Display-Manager sehr langsam. Dies äußert sich durch eine geruhsame Bildschirmausgabe. Das System ist nicht in der Lage, mehr als eine Maske im Speicher zu halten. Dadurch erfolgt bei jedem Seitenwechsel ein erneuter Zugriff auf die Diskette. Schnelle Programme, wie sie sicherlich in Zukunft erwartet werden, sind somit mit dem Display-Manager kaum zu realisieren.

Neuere Anwendungsprogramme für Personal Computer arbeiten immer häufiger mit sogenannten Windows oder Fenstern. Mit ihrer Hilfe läßt sich der Bildschirm in unterschiedliche Teilbildschirme zerlegen, die dann zum Einblenden von Hilfstexten oder Listen verwendet werden können. Diese Technik erhöht dabei gerade dort die Bedienungsfreundlichkeit, wo häufig wie in der Landwirtschaft Abkürzungen oder Codes in Eingabefelder der Masken einzugeben sind. Im einem Window lassen sie sich bequem einblenden, ohne daß die eigentliche Bildschirmmaske verschwinden muß. Mit dem Display-Manager ist diese Technik nicht möglich.

Weiterhin reagiert das System beim Editieren von Bildschirmmasken sehr empfindlich auf einen Systemzusammenbruch. Bei einem erneuten Start kann es dabei zu Programmproblemen sehr verschiedener Art kommen.

Die Routinen zur Reorganisation des Formatfiles arbeiten nicht einwandfrei. Nach einem Absturz sind die zuletzt bearbeiteten Formate nicht mehr vorhanden. Dies kann besonders bei Eingabemasken mit vielen Datenfeldern, wie z.B. der Schlagkartei sehr ärgerlich sein.

Ähnlich wie der Display-Manager ist der Access-Manager von seiner anfänglichen Idee positiv zu sehen. Der Aufbau von Datenfiles, die von jeder beliebigen Programmiersprache her anzusprechen sind, ist eine alte Forderung in der Datenverarbeitung.

Der Vorteil des Access-Managers von Digital Research ist es, daß ein Zugriff auf die Datenfiles sowohl von PL/1 als auch von Fortran und C möglich ist.

Die Datenverwaltung beim Access-Manager beruht auf dem B-Tree Konzept. Der eigentliche Zugriff auf die Daten und den Indexfile erfolgt von Run-Time Modulen aus, die beim Linkprozeß an das Ausführungsprogramm gebunden werden.

Call-Befehle greifen auf die Daten zu, die vom Anwendungsprogramm abgeschickt werden. Für jeden Vorgang, den der Access-Manager ausführen soll, steht ein Modul zur Verfügung. So gibt es zum Beispiel ein Modul zum Lesen des nächsten Satzes oder zum Einfügen eines Satzes usw..

Bei der Arbeit mit diesem Werkzeug traten folgende, schwerwiegende Mängel auf, die letztendlich zu einer Abkehr von diesem Programm führten.

1. Die Funktionen des Access-Manger arbeiten nicht in der Weise wie im Handbuch beschrieben. So werden Übergabeparameter bei CALL-Befehlen nicht mit den Werten besetzt, die angegeben sind.
2. Die Datenfiles sind nach einem Systemzusammenbruch, der sowohl durch Stromausfall, Betriebssystemfehler oder Anwendungsprogramm bedingt sein kann, häufig nicht mehr oder nur noch teilweise ansprechbar.
3. Die Hilfsprogramme zur Reorganisation der Datenfiles arbeiten sehr unsauber. Diese Routinen löschen in einer nicht nachvollziehbaren Weise Daten aus den Files.
4. Abstürze des Access-Manager, die beim Aufruf des Programmes auftreten, führen in der Regel zu einem vollkommenen Absturz des Anwendungsprogrammes und des Betriebssystems.
5. Der Zugriff von verschiedenen Programmiersprachen auf die gleichen Datenfiles ist nicht, bzw. sehr umständlich, möglich.

6. Der Zugriff auf die Daten über die Indexfiles wird nach längerem Einsatz immer unsicherer. So zeigen die Pointer im Indexfile bei längerer Anwendung auf die falschen Datensätze. Dieser Mißstand ist mit Hilfe der mitgelieferten Reorganisationsroutinen nicht zu reparieren.
7. Der Zugriff auf die Daten ist wesentlich umständlicher als bei der indexse-quentiellen Dateiverwaltung. So kann ein Datensatz nur über die Satznummer angesprochen werden. Die Satznummer wiederum kann der Anwender nur über einen Zugriff auf den Indexfile bekommen.

Will ein Anwender einen Satz direkt ansprechen, muß er zuerst die Satznummer über einen Zugriff auf den Indexfile herausfinden. Im Anschluß daran kann der Datensatz mit Hilfe der Satznummer über einen CALL-Aufruf gelesen werden.

Von der Idee ist die Entwicklung von Digital Research, Programme anzubieten, die eine Erstellung von Bildschirmmasken und die Datenverwaltung unabhängig von der Programmiersprache erlauben, sehr interessant. In der Realisierung des Vorhabens sind jedoch noch sehr große Mängel vorhanden. Es ist zu bedauern, daß das Projekt von Digital Research nicht mehr weiter gepflegt und weiterentwickelt wird. Dies führte schließlich dazu, sich nach anderen Lösungsmöglichkeiten umzusehen. In Betracht kamen hier komplette Entwicklungssysteme. Sie bieten einen Compiler in einer speziellen Sprache an. Darüberhinaus stellen sie einen dazu passenden Maskengenerator, eine interaktive Programmtestmöglichkeit und spezielle Editoren zum schnellen Erstellen von Quellencodes zur Verfügung. Der Vorteil bei diesen Systemen ist, daß ein Quellencode sowohl für das Datenhandling als auch für den Maskenaufbau erzeugt wird, in den der Anwender jederzeit eingreifen kann. Über ein solches System wird ausführlicher im Beitrag, Pohlmann, 1986, berichtet.

## 20.4 Literatur

- 1) **BYTE, THE SMALL SYSTEMS JOURNAL, 1986:**  
August 1986 VOL.11, NO 8  
McGraw-Hill Inc. Peterborough, New Hampshire
- 2) **COMPUTER CONTROL SYSTEMS, 1985:**  
FABS PLUS  
Computer Control Systems, Lake City, 1984, 1985
- 3) **COMPUTER CONTROL SYSTEMS, 1985:**  
Autosort/86M  
Computer Control Systems, Lake City, 1984, 1985
- 4) **DIGITAL RESEARCH, 1983:**  
Display-Manager, Productivity Tool  
User's Guide  
Digital Research, Pacific Grove, 1983
- 5) **DIGITAL RESEARCH, 1983:**  
Access-Manager, Productivity Tool  
User's Guide  
Digital Research, Pacific Grove, 1983
- 6) **HENK, M., 1983:**  
Der IBM-Personal Computer  
Markt & Technik München, 1983
- 7) **LAUER & WALLWITZ, 1985:**  
TURBO-MACHINE 1.0  
Lauer & Wallwitz, Wiesbaden, 1985, 1986
- 8) **LUDWIGS D., 1985:**  
Professionell arbeiten mit dem IBM-PC  
Software-Werkzeuge in Microsoft-Pascal  
Vogel-Buchverlag Würzburg, 1985
- 9) **POHLMANN, J., 1986:**  
Cobol-Compiler als effiziente Programmierwerkzeuge mit der  
Möglichkeit des Programmtransfers für Großrechner - Personal Computer  
Agrarinformatik, Band 10, Verlag E. Ulmer

Zusammenfassung

Der vorliegende Text informiert über das Konzept einer computerergänzten Datenmanagement in der Züchtung von Weiden. In diesem Zusammenhang werden drei Computer-Typen: ein hand-held Computer zur mobilen Datenerfassung; ein Personal Computer zur Zwischen- und Übertragung von Programmen und Daten; eine Großrechenanlage, auf der das Datenbanksystem SIR/DMS und die Statistik- bzw. Graphikprogrammpakete SPSS und PLOTIT benutzt werden. In dem Text wird der Schwerpunkt auf die mobile Datenerfassung gelegt. Der verwendete hand-held Computer - der CP/M Kompatible TRALER PC 410 - sowie das Programm, das zur mobilen Datenerfassung eingesetzt wird, werden besprochen. Ferner werden die mit dem Gerät und dem Programm in der Praxis gemachten Erfahrungen mitgeteilt.

Abstract

The presented text contains information on a concept of using computer aided data management in breeding weaning thereby hybrid crossbreeds. The concept is based on three types of computers: a hand-held computer for collecting data at the experimental site; a personal computer used for temporary data storage and transfer of data and programs; a mainframe where the data are stored in a data base or analysed using the packages SPSS or PLOTIT. Special attention is paid to the collection of data with the portable computer TRALER PC 410 (compatible to CP/M systems) and the program employed for sampling data at the experimental site. Additionally some experiences about the practical application of the program and the hand-held computer are described.