

## Liebe LeserInnen!

Je größer die Zahl der installierten Einheiten, desto zäher ihre Umstellung auf eine neue Technik, ganz zu schweigen von der Umstellung der Benutzer. Die Schule ist ein gutes Beispiel. Während jeder neu einsteigende Schüler oder Lehrer ganz selbstverständlich mit Windows arbeitet und irgendwo auch das Symbol für die DOS-Box gesehen hat, gibt es noch ganze Lehrsäle, bei denen an den Betrieb von Windows noch einige Zeit nicht zu denken sein wird. Auch Lehrer, die ein großes Arbeitsvolumen in Vorbereitungen investiert und unter rauen Unterrichtsbedingungen getestet haben, stellen sich nur allmählich auf eine neue Ara ein.

Der Zug der Zeit: grafische Benutzeroberflächen, egal welcher Provenienz, stellen sich als die Hilfe dar, die es auch einer breiteren Schicht von Benutzern ermöglicht, ohne großartige Vorkenntnisse mit dem PC umzugehen. Ob es nun WINDOWS, OS/2, X-Windows oder DESQVIEW-X ist, alle kommen jenen entgegen, die nicht die Spezialisten sind; und Hand aufs Herz: auch die Spezialisten unter uns sind es ja nur in einem kleinen Gebiet. In allen anderen Bereichen tut es gut, sich durch ein ausgeklügeltes Bedienungsinterface durch das Leistungsangebot eines komplexen Programmes leiten zu lassen.

Welches der Fensterbetriebssysteme wir unterrichten werden? Wenn wir die überhaupt unterrichten müssen - denn die Schüler eignen sich ja die Handgriffe in kurzer Zeit selbst an -, ist die Auswahl - zumindest im TGM - zugunsten von **Windows** mit X-Windows-Anteilen ausgefallen; ohne große Evaluation, einfach dem Markt und der Verfügbarkeit für einen durchschnittlichen Lehrer-PC entsprechend. Daß einzelne Spezialisten und besondere Anwendungen, wie die Mailbox His Master's Voice auch **OS/2** verwenden, sollte sich in den **PC-NEWS** durch entsprechende Beiträge dieser Leser ausdrücken.

Es wird für Softwareentwickler und natürlich auch für unsere Schüler - speziell die Techniker unter ihnen - immer wichtiger, nicht nur gut funktionierende, sondern auch gut bedienbare Software herzustellen. Dabei entwickelt sich zwischen Lehrer und Schüler ein neues Lern- und Lehrerlebnis: ein Miteinander-Lernen, denn nichts von dem, was heute unterrichtet werden soll, haben Lehrer in ihrer Lehrzeit oder in ihrem Beruf mitbekommen, alles wird mehr oder weniger nacherfunden. Ist Programmierung allein schon ein Gebiet für eine Handvoll Tüftler, so stellt uns Programmierung mit Fenstern vor ganz neue Probleme:

### Ein Vielfaches an Komplexität

Eine Komplexität, die etwa die Sprache C zum Assembler für Windows-Programmierung degradiert; C++ wird zum Muß; Bibliotheken ohne objektorientierten Ansatz gibt es praktisch nicht.

Glücklicherweise ist die Gründer-Zeit der Windows-Programmierung mit SDK und C-Kommandozeilen-Compiler schon vorbei, die neuen Compiler warten mit neuen Werkzeugen auf, die zumindest den Entwurf der Benutzeroberfläche zu einem Bildschirmspiel reduziert.

Weiters bietet sich als eine möglicherweise einfachere Einsteigersprache Visual-Basic an, die zwar nicht mit dem Tempo und mit den Datenstrukturen von C mithalten kann, dafür aber raschere Erfolge garantiert.

### Ein Vielfaches an Ressourcen

DOS-Programme konnte man mit einfachsten Mitteln zur Not auch auf einem XT entwickeln. Damit ist es jetzt endgültig vorbei. Unter 8 MB Hauptspeicher ist nicht viel zu machen, eine volle Installation eines Compilers benötigt ab 50MB Festplattenplatz. Die Informationsflut erfordert mindestens ein CD-ROM-Laufwerk an einem Arbeitsplatz; in Lehrsälen, wo ein einzelner Schüler nicht immer unmittelbaren Zugang zur CD haben kann, wären mehrere gleichzeitig angeschlossene CD-ROM-Laufwerke anzuraten.

Leider gibt es aus dem Ressourcen-Engpaß (neben der Fensterverweigerung) keinen anderen Ausweg, außer den, in einer Niedrigpreis-phase RAM, ein schnelleres Motherboard und ein bißchen Festplatte zu kaufen.

### Ein Vielfaches an Informationsmaterial

Konnte man die Programmierung unter DOS mit ein bis zwei kopierten Blättern über die DOS-Aufrufe unterrichten, wird es angesichts der sich stapelnden Windows-Book-Ware schwierig - speziell für Windows-Anfänger - ein sinnvolles Subset aus dem Material herauszuschälen.

Glücklicherweise kommt uns aber die sich rasch entwickelnde Technologie der Informationsdarstellung zu Hilfe; die Compiler haben die Handbücher on-line, kontextsensitive Hilfe macht Nachschlagen praktisch entbehrlich.

### Und was soll dabei eine kleine Zeitung?

Es ist für eine kleine Zeitung unmöglich und auch für große Zeitschriften schwierig, ein großes Kapitel, wie es das der Windows-Programmierung ist, in einem Schwerpunkt einigermaßen geschlossen darzustellen. Immerhin widmen sich diesem Thema viele namhafte Zeitschriften. Wenn die Windows-Programmierung in Zukunft den Stellenwert haben wird, den heute die DOS-Programmierung noch hat, bedeutet unsere heutige Ausgabe lediglich einen Startschuß für eine neue Phase der Hilfestellung durch die **PC-NEWS** für den EDV-Unterricht und für alle, die sich mit diesem Problem auseinandersetzen wollen oder müssen. **Alle aktiven "Fenster"-Programmierer sind herzlich eingeladen, Beiträge und Tips an die Redaktion einzusenden; die Leser der PC-NEWS werden es ihnen danken. "Fenster" ist dabei stellvertretend für alle modernen grafischen Benutzeroberflächen.**

### Wie also kann man sich einarbeiten?

Zunächst an Hand eines der großen Compiler: Schwerpunkt Windows-Programmierung wäre eigentlich ein Anlaß, die für diesen Zweck verfügbaren Compiler zu vergleichen. Welche sind das?

- **BORLAND:** BORLAND-C++ (noch 3.1 bald 4.0)
- **MICROSOFT:** Visual C++, 1.0
- **SYMANTEC:** C++, Version 6.0

Bisher hatte BORLAND die meisten Freunde bei den Schulen, wie etwa die Skripten der **ADIM** zeigen. Der Grund ist ziemlich sicher die einheitliche Benutzeroberfläche für PASCAL und C verbunden mit einer gut für den PC erweiterten Bibliothek. Je nachdem, ob in einer Schule eher PASCAL/C oder Visual-BASIC/C unterrichtet wird, kann den Ausschlag eher zu Gunsten des BORLAND- oder des MICROSOFT-Compilers ergeben, denn in den Schulen unterliegt die Bewertung der Compilerqualitäten anderen Maßstäben als in der Industrie. Ein hohes Maß an Einheitlichkeit bei der Bedienung verschiedener Compiler (BASIC-PASCAL-C) hilft, die Informationsflut zu bewältigen.

Nun, BORLAND-C++ Version 4.0, vielleicht als Weihnachtsgeschenk gedacht, ist noch nicht eingetroffen. Dazu kommt, daß die bestehenden Installationen nicht gerade nach Erneuerung schreien, denn die Zahl der Windows-Programmierer in den Schulen ist (noch) klein. Wenn man also vergleichen will, kann man das nur mit der alten Version des BORLAND-Compilers. Daher wollen wir über Compiler-Vergleiche, wenn überhaupt, in einer späteren Ausgabe berichten.

Wir müssen aber damit rechnen, daß bei Installation eines neuen Compilers auch noch für DOS entwickelt werden soll. Das läuft bei Visual-C unter der einheitlichen Windows-Oberfläche, bei Turbo ist bei Version 3.1 eine Trennung von DOS- und Windows-IDE vorgesehen; in der Version 4.0 des Compilers soll aber die DOS-Entwicklungsumgebung überhaupt fehlen und Turbo-Vision, die Fensterbibliothek für DOS, nur in einem Zusatzpaket zu erwerben sein. Man wird sehen.

Um einen einigermaßen fairen Vergleich durchzuführen, beschränken wir uns in der heutigen Ausgabe auf die Darstellung eines "HELLO-WORLD"-Programms für Windows unter Benutzung der Compiler VISUAL-C++ 1.0 und BORLAND-C++ 3.1, wobei jeweils zwei Methoden angewendet werden a) die DOS-kompatible (QuickWin/EasyWin) und b) die durch eine Klassenbibliothek unterstützte (Foundation-Classes/Object Windows). Daß der dritte Compiler (SYMANTEC) nicht verglichen wird, hat einen guten Grund: er benutzt ebenfalls die Foundation-Classes, der Unterschied liegt dann also bei der Benutzeroberfläche, die aber hier nicht verglichen werden soll.