

Kersch

inforum

INFormationsforum des Rechenzentrums der Universität Münster

Jahrgang 16, Nr. 1 – Februar 1992

ISSN 0931-4008

Inhaltsverzeichnis

Editorial	2
---------------------	---

RUM-Aktuell

Die neue DV-Ausstattung	3
Neues vom System	5
Ende des UTS-Betriebs	6
Benutzerverwaltung	7
Neues vom SPSS	9
Thematische Karten unter SAS	9
Tips zur Schadensvermeidung bei PCs	10
Personalien	11

RUM-Tutorial

Ein neuer Schriftauswahlmechanismus für L ^A T _E X	12
Das X-Window-System	17

RUM-Lehre

Lehrveranstaltungen im 1. Halbjahr 1992	20
---	----

RUM-Aroma

Vorläufige Benutzungsordnung	27
inforum -Quiz	31

Impressum

inforuM

ISSN 0931-4008

Redaktion: W. Bosse (Tel. 83-2461)
 St. Ost (Tel. 83-2681)
 H. Pudlatz (Tel. 83-2472)
 E. Sturm (Tel. 83-2609)

Satz: S. Arnold
 R. Wilmes

Satzsystem: L^AT_EX
 Druck: Drucktechnische
 Zentralstelle der WWU

Universitätsrechenzentrum
 Einsteinstraße 60
 4400 Münster

Auflage dieser Ausgabe: 1000

Redaktionsschluß der nächsten
 Ausgabe: 13.03.1992

Editorial

von

von E. Sturm

Wesentliches Thema dieser **inforuM**-Ausgabe sind die neuen Hard- und Softwaresysteme, die als Ergebnis der großen Beschaffungsmaßnahme 1991/92 im oder ins Haus stehen. Vier Artikel in diesem Heft und weitere Artikel in einem geplanten Sonderheft befassen sich hiermit.

Auch die **inforuM**-Redaktion wollte da nicht zurückstehen und endlich den Traum verwirklichen, unserer Informationsschrift ein besseres Äußeres zu verschaffen: keine losen Blätter, sondern eine stabil geheftete Zeitschrift. Den Pferdefuß erlebten wir, als wir dachten, mit der redaktionellen Arbeit fertig zu sein: Die Anzahl der Seiten muß nun auf einmal durch 4 teilbar sein! Diesmal sind wir aber noch glimpflich davongekommen, aus dem Pascalprogramm des letzten Artikels mußten nur ein paar Leerzeilen entfernt werden.

Aber lassen Sie mich auch ein paar inhaltliche Gedanken ansprechen - zum Thema, welche Hard- und

Software man benutzen soll (rein persönliche Gedanken - prononciert formuliert).

Manche wettern gegen proprietäre (nur von einem einzigen Hersteller lieferbare) Betriebssysteme. Manche sehen vor allen anderen Dingen die Mips (Millionen Instruktionen pro Sekunde), die ein Rechner schafft. Manche predigen die Einhaltung von ISO-Normen und verdammen die sogenannten Industrie-Standards.

Meiner Meinung nach handelt es sich hier in jedem Fall um „Sekundärtugenden“. Standpunkt des Benutzers ist es primär, daß er sein Ziel auf einfachste Weise erreicht.

Wer in einer bestimmten Zeit seine Diplomarbeit schaffen muß, dem ist es meistens egal, auf was für einem Betriebssystem seine Programme laufen, wenn er nur nicht dauernd Überraschungen erlebt, die ihn von seinem eigentlichen Thema abbringen.

Wer ein Programm gefunden hat, das eben das leistet, was er benötigt, dem ist es egal, ob es einen doppelt so schnellen Rechner zum gleichen Preis gibt, wenn dort nur ein unbequemerer Programm zur Verfügung stünde.

Gleiches gilt für Normen: Es ist keine Tugend an sich, wenn man nur GKS bei der Grafikprogrammierung verwendet, wenn der gleiche Zweck viel einfacher mit einem interaktiven PC-Programm erreicht werden könnte.

Womit ich nicht bestreiten möchte, daß Sekundärtugenden auch Tugenden sind. Wer längerfristige Projekte plant, der sollte sich schon überlegen, auf welche Standards und welche Rechner mit welchem Betriebssystem er sich stützen möchte. Eine ISO-Norm ist da dauerhafter als ein Industrie-Standard. Manche Rechner bieten mehr fürs Geld. Das Betriebssystem sollte aber vor allem in der Hinsicht geprüft werden, daß man möglichst wenig geärgert wird. Wer schon einmal unter MS-DOS aus Versehen ganze Verzeichnisse leerte, weil er auf die Frage „Are you sure (Yes/No)?“ mit „Yes“ geantwortet hat, wird verstehen, was ich meine!

Wenn Sie also die folgenden Seiten lesen, wundern Sie sich bitte nicht, daß mehrere Rechnertypen und mehrere Betriebssysteme und mehrere scheinbar ähnliche Programme beschafft wurden. Wir hoffen, daß so auch für Sie das Richtige dabei ist.

RUM-Aktuell

Die neue DV-Ausstattung

von

W. Bosse, W. Held und H.-W. Kisker

Endlich ist es soweit: Nach vielen „mageren“ Jahren – bezogen auf bisher in Münster verfügbare Computer – konnte eine Reihe von leistungsfähigen Rechnern in Betrieb genommen werden.

Als Lohn der langjährigen erheblichen Anstrengungen, die von Seiten der zuständigen Gremien und des Rektorats zusammen mit dem Universitätsrechenzentrum unternommen wurden, konnte im Verlauf der letzten 7 Monate der weit überwiegende Teil des geplanten DV-Gesamtsystems im Wert von 15 Mio DM von der Universität gekauft werden.

Die noch ausstehenden Komponenten werden in diesem Jahr installiert werden. Dem Land Nordrhein-Westfalen und dem Bund sei schon an dieser Stelle für die Bereitstellung der Mittel gedankt. Die Kommission für Rechenanlagen der Deutschen Forschungsgemeinschaft hat dankenswerterweise eine Begutachtung vorgenommen, die der Universität die Realisierung eines modernen und flächendeckenden Versorgungskonzeptes gestattet.

Welche Ausstattung wurde bisher erreicht?

Rechner

Im Rahmen eines mehrstufigen Versorgungskonzeptes wurden insbesondere Server auf der einen und Arbeitsplatzrechner/Workstations auf der anderen Seite realisiert. Zu den Servern gehören zentral im Universitätsrechenzentrum ein Vierprozessor-System IBM ES/9000-610 mit 2 Vektorzusätzen, 256 MB Hauptspeicher, 37.5 GB Magnetplattenspeicher, 32 EA-Kanälen (davon 8 mit einer Übertragungsrate von 10 MB/s) und 2 FDDI-Anschlüssen an das Backbone-Netz der Universität. Die Leistungsfähigkeit eines jeden Prozessors ist 8 bis 10 mal so groß wie die eines der bisher genutzten IBM-4381-Rechner. Durch einen Vektorzusatz erreicht man für den zugeordneten Prozessor bei geeigneten Aufgaben im Schnitt nochmals eine Leistungssteigerung um den Faktor 2 bis 3. Als Betriebssystem sollen noch eine Weile VM/SP mit

CMS und MVS im bisherigen Zustand eingesetzt werden. Für diverse Server-Aufgaben ist der Übergang auf MVS/ESA geplant. Vor allem aber soll auf dem Rechner möglichst bald ein Unix-System eingeführt werden (getestet wird in Kürze AIX/ESA). Damit kann in einem kontinuierlichen Prozeß der Übergang von einem herstellereigenen Betriebssystem (wie VM und MVS) auf das herstellerunabhängige Unix-System erfolgen, ohne daß die bisherigen Nutzer in kurzer Zeit große Umstellungsprobleme zu bewältigen hätten.

Vorrangig für die Fachbereiche Physik und Chemie wurde als Server ein Einprozessorsystem DEC VAX 9000-110 gekauft, das insbesondere für die dort schon vorhandenen VAX-Rechner als Wachstumspfad gedacht ist. Dieser Rechner ist mit einem Vektorzusatz, 256 MB Hauptspeicher, 6 GB Magnetplattenspeicher und einem FDDI-Anschluß an das Backbone-Netz ausgestattet. Die Leistungsfähigkeit dieses Rechners dürfte etwa vergleichbar sein mit der eines Prozessors des zuvor beschriebenen zentralen Systems. Als Betriebssystem wird VMS eingesetzt.

Für zahlreiche Fachbereiche wurden daneben über 90 leistungsfähige Workstations und Server gekauft. Weit überwiegend handelt es sich dabei um RISC-Rechner der Firmen Sun und IBM. Zum Teil sind sie mit besonderen Grafikfähigkeiten ausgestattet, zum Teil sollen sie als Fileserver dienen und haben dafür eine größere Magnetplattenspeicherkapazität. Wieder andere zeichnen sich durch besondere Rechenleistung aus. Periphere Ergänzungen (z. B. Magnetplatten und Kassettenlaufwerke) und Hauptspeichererweiterungen für die Workstations und Server wurden teilweise von Drittanbietern gekauft, denn diese Geräte sind mit „Standard-Hauptspeichern“ ausgestattet und die Peripherie wird jeweils über „Standard-SCSI-Bus“ angeschlossen.

Das Spektrum der Rechenleistung für diese Rechner ist relativ breit. Die gekauften Sun-Rechner haben MIPS/MFLOPS-Raten von 15.8/1.7 bis 28.5/4.2; die Rechenleistung der installierten IBM-Rechner reicht von 36.8/11.7 bis 62.2/25.2. Da neben der Rechenleistung auch andere Kriterien (z. B. Softwarevielfalt, bisherige Erfahrungen und eigene Software-Entwicklungen) bei der Kaufentscheidung zu berücksichtigen waren, ist verständlich, daß nicht nur die leistungsfähigsten Rechner gekauft wurden. Alle diese Rechner werden mit Unix-Betriebssystemen betrieben.

Schließlich wurden ca. 170 Arbeitsplatzrechner vom Typ AT386 und AT486 gekauft. Diese Rechner wurden alle mit 8 MB Hauptspeicher- und i. a. 200 MB Magnetplattenkapazität ausgestattet. Ihre Leistungsdaten lassen sich in MIPS/MFLOPS-Raten mit 8.8/0.4 für AT386 bzw. 17/1.5 für AT486 charakterisieren.

Die meisten dieser Rechner werden mit dem Betriebssystem MS-DOS genutzt, einzelne sind aber auch mit OS/2- oder Unix-Betriebssystem ausgestattet worden.

Periphere Geräte

Zentral im Universitätsrechenzentrum wurde ein hochauflösender Farbdrucker Canon CLC300 installiert, der sowohl für die Ausgabe von Zeichnungen als auch Bildern geeignet ist. Das Gerät kann in Ausnahmefällen auch als Farbscanner genutzt werden. Einige technische Daten dazu lauten: Formate DIN A4/A3, 400 Punkte/inch, über 16 Millionen Farben pro Bildpunkt. Die Inbetriebnahme wird in einigen Wochen erfolgen.

In allen Fachbereichen sind nun postscriptfähige Laserdrucker (300 Punkte/inch) vorhanden. Fast alle Fachbereiche verfügen über CD-ROM-Laufwerke und Scanner. Insbesondere Scanner und Laserdrucker können für Telefax-Dienste im lokalen Rechnernetz (LAN) genutzt werden. Ein größerer Test der Telefax-Software im LAN ist nahezu abgeschlossen. Diese Software steht anschließend allen Fachbereichen zur Verfügung. Von jedem Rechner im Netz kann dann ein Fax zu jedem anderen Rechner und zu jedem Telefax-Gerät (auch außerhalb der Universität) geschickt werden. Der Eingang einer Telefax-Nachricht kann, weil die Adressierung leider nicht genormt ist, nur nach menschlichem Eingriff zum Arbeitsplatz weitergeleitet werden. Wir hoffen, daß die Telefonzentrale diesen Dienst übernehmen wird.

Aber zurück zur Beschaffung. Vereinzelt wurden noch andere Peripherie-Geräte gekauft, zu denen einige Beispiele genannt werden sollen: hochauflösender Drucker VT600W (Rechtswissenschaften), S-Bus-Entwicklungskarte und S-Bus-Entwicklungssystem für Sun (Psychologie), Grafiktablett und Bildverarbeitungssystem (Biologie), Transputerkarten (Geowissenschaften) und Scanner (Ev. Theologie und Geschichte).

Software

Für MS-DOS-Systeme wurden in großem Umfang die bekannten Standardprodukte der Textverarbeitung, diverse Compiler, Norton Utilities, XTREE, MS-Windows usw. beschafft. An ausgefallenen Produkten sind u.a. zu nennen: Optopus OCR, Chalera Word Scan, Omnipage, PLT (Chemieformel Layout), Transputer WATFOR, Notenschriftprogramm Score sowie Geoworks.

Für VMS-Systeme stehen u. a. zur Verfügung: Pathworks, Fortran, Basic, Pascal, GKS, LSI, DXML, RDB/VMS, C, Vax Set, PL/I, Image Scan, Data-trieve, Lisp und Phigs.

Für Sun-Rechner wurden beschafft: S-PLUS, ARC/Info und als zentrale Server-Lizenzen Fortran, Pascal, Modula, C++, Phigs, ANSI-C, ONC-Toolkit und SPARC-Tools.

Für IBM RS/6000 (AIX) wurden installiert: Fortran, Pascal, GRAPHICS, Netz-Management, Gaussian, Reduce, Maple, Mathematica und Axiom.

Als leistungsfähiges Software-Paket zur interaktiven Grafik-Erstellung wurde PV~WAVE ausgewählt. Software-Lizenzen wurden für alle Workstation-Arten und für VMS-Rechner beschafft. Für den ES/9000-Rechner sind neben den bisher schon genutzten Programmen zusätzlich das Datenbanksystem DB2, die Kommunikationsssoftware TCP/IP mit dem Netzdateisystem NFS, die ISO-OSI-Protokolle für die Kommunikation und z. B. Gaussian vorhanden.

Die Programmbibliothek IMSL liegt für alle eingeführten Hardware-Plattformen (inkl. MS-DOS) als Einzellizenz vor. Für die NAG-Bibliothek gibt es – außer bei MS-DOS – eine Campuslizenz.

Für Arbeitsplatzrechner wurden inzwischen sechs verschiedene X-Window-Emulationen getestet. Ein Produkt hat sich als besonders geeignet herausgestellt. Die Beschaffung soll jetzt angegangen werden. Damit werden über das LAN auch vom Arbeitsplatzcomputer Zugänge z. B. zum Unix-System und anderen Servern möglich, so als wären es direkt angeschlossene Grafikbildschirme. Wahlweise kann über das Netz mit der X-Terminal-Emulation vom Arbeitsplatz aus einmal der eine, ein anderes mal ein anderer Rechner, so wie er gerade benötigt wird, erreicht werden.

Da Computer von Menschen genutzt und nicht immer ohne fachtechnische Unterstützung betrieben werden können, sind im Universitätsrechenzentrum und in den Fachbereichen Organisationsveränderungen vorgesehen. Das Universitätsrechenzentrum wird z. B.

auch Geräte, die in den Fachbereichen aufgestellt sind, mitbetreuen oder mitbetreiben. Dies gilt schon jetzt für die Systeme VAX 9000-110, Sun und IBM RS/6000 sowie natürlich weiterhin für DOS- und OS/2-Rechner. Wegen der großen Zahl der Computer kann dies letztlich nur geleistet werden, wenn diese Unterstützung oder der Betrieb weitgehend automatisiert werden. Das Datennetz der Universität, das noch weiter ausgebaut werden muß und an das die Rechner nach und nach angeschlossen werden, bietet dafür die Voraussetzung. Darüber hinaus wird das Universitätsrechenzentrum sein Dienstangebot auch fachbereichsweise über sogenannte DV-Moderatoren abrufbar halten. Auch auf die Fachbereiche kommt neuer Unterstützungsaufwand vor Ort zu. Dort tätige DV-Administratoren sollen Probleme sammeln und in enger Kooperation mit dem Universitätsrechenzentrum lösen helfen. Dies könnte von den Mitarbeitern auch als Zusatzqualifikation gesehen werden.

Im Mai dieses Jahres soll die offizielle Inbetriebnahme aller Anlagen erfolgen. Zu diesem Zeitpunkt ist auch ein **inforuM**-Sonderheft mit einer ausführlichen Dokumentation der beschafften Teile geplant. Zur Zeit liegt in den Dekanaten und bei den EDV-Beauftragten der Fachbereiche eine detaillierte Auflistung als Zwischenabrechnung für die in den Fachbereichen beschafften DV-Komponenten vor.

Neues vom System

von

St. Ost

Nachdem es lange Zeit recht wenig Neues vom System gegeben hat, könnte man jetzt sehr viel erzählen. Aber ich will mich auf das beschränken, was wir seit Installation des neuen Rechners getan haben und wie wir uns die unmittelbare und mittelbare Zukunft vorstellen.

Was wurde denn nun eigentlich beschafft? Im Zentralrechnerbereich kann man dies recht einfach beantworten.

- Als Zentraleinheit eine IBM 9121/610, mit vier Prozessoren, 256 MByte Zentralspeicher und 32 Kanälen. Davon haben zwei Prozessoren einen Vektorzusatz, und 8 Kanäle sind Glasfaserkanäle mit einer maximalen Übertragungsrate von 10 MByte/s. Dieses ist die Endausbaustufe, die erst im Frühjahr installiert werden wird.

- Ein Plattensubsystem IBM 3390 mit insgesamt 37 GByte Speichervermögen.

- 2 FDDI-Kontrolleinheiten.

- MVS/SP V4R2 (MVS/ESA) als Betriebssystem plus der gewohnten Grundausstattung an Software wie Compiler, Sortierprogramme, Kommunikations-Software, statistische und numerische Standard-Software sowie DB2 als Datenbank.

Am 3. Dezember 1991 wurde der Rechner installiert und bereits zwei Tage später in den Produktionsbetrieb genommen. Ermöglicht wurde diese ungewöhnlich kurze Migrationsdauer durch eine interessante Hardwareeigenschaft der heutigen Zentralrechner. Diese lassen sich mittels Hardware in zwei oder mehrere Partitionen aufteilen. Diesen Partitionen können Prozessoren ganz oder teilweise zugeordnet werden, ebenso wie Teile des Zentralspeichers. Zusätzlich kann bestimmt werden, ob eine Partition lediglich den altbekannten /370-Instruktionsvorrat hat oder ob sie den neuen Instruktionsvorrat (genannt /390) beherrscht, der die Voraussetzung für modernere Betriebssysteme wie MVS/ESA oder AIX/ESA ist.

Bisher sind zwei Prozessoren und 128 MByte installiert. Definiert wurden zwei Partitionen, genannt VMHPO und MVESA. VMHPO ist eine traditionelle /370-Partition. Ihr wurde ein Prozessor, 64 MByte Hauptspeicher und 32 MByte Erweiterungsspeicher zugeordnet. Als Betriebssystem läuft hier das in der alten A-Maschine eingesetzte VM/HPO, und unter diesem das MVS/370 der bisherigen B-Maschine. Geändert hat sich für die Betriebssysteme fast nichts und genauso wenig für die Benutzer der bisherigen A- und B-Maschine. Bis auf den kleinen Unterschied natürlich, daß ein Prozessor des neuen Rechners etwa 8 bis 10 mal schneller als eine 4381 ist. Dies erklärt die kurze Umstellungsdauer, da nicht sehr viel mehr zu tun war, als Platten- und Bändeinheiten sowie Terminal-Controller umzuverkabeln und die VMHPO-Partition einzurichten. In der MVESA-Partition haben wir das neue MVS/SP V4 installiert und bereiten die Migration des alten in das neue MVS vor. Diese Umstellung ist relativ aufwendig, da zwischen beiden MVS-Versionen ein Unterschied von drei Generationen oder, anders ausgedrückt, von etwa 10 Jahren besteht.

Wie stellen wir uns den zukünftigen Einsatz des Zentralrechners vor? Dazu zunächst einige Thesen:

1. Die Dialogkomponente eines Zentralrechners wird zunehmend unwichtiger, da die bisher im CMS vorgenommenen Arbeiten leichter und für den Benutzer komfortabler auf PCs und Workstations wahrgenommen werden können.
2. Interessant ist die Nutzung eines Zentralrechners als Server für rechen- und speicherintensive Batch-Jobs, für Datenbanken, für zentrales Backup und andere zentral wahrzunehmenden Service-Leistungen.
3. Unix wird als herstellerunabhängiges Betriebssystem für die Universität von zentraler Bedeutung sein. Herstellerabhängige Betriebssysteme wie MVS/ESA haben lediglich eine mittelfristige Bedeutung. Sie sind so lange wichtig, wie Unix-Versionen für den Zentralrechner noch nicht das gleiche Leistungsvermögen für zentrale Rechenzentrumsdienstleistungen haben wie traditionelle Betriebssysteme.

Und was folgt aus diesen Thesen? VM/CMS hat keine Zukunft mehr, es wird in Münster, wie man so schön sagt, funktional stabilisiert, d. h. wir sorgen dafür, daß es läuft, aber wir entwickeln und pflegen es nicht weiter. Sie sollten im CMS keine neuen Projekte mehr beginnen, sondern alte lediglich in Ruhe abschließen. Stattdessen sollten Sie den virtuellen PC, den das CMS ja dem Benutzer geboten hat, durch einen realen PC ersetzen.

Rechenzeitintensive Batchanwendungen können einfach und komfortabel über FTP an das MVS als zentralen Batch-Server gegeben werden, genauso wie Sie es bisher mit dem SENDJOB-Kommando des CMS getan haben. Sollten Sie Ihre Programme noch selbst geschrieben haben, so entwickeln Sie diese am PC oder einer Workstation, die Sie dann, sofern Sie sich an die Sprachstandards gehalten haben, relativ einfach ins MVS portieren können. Für Datenbanken gilt ähnliches. Die Datenhaltung selbst erfolgt unter DB2 im Zentralrechner, der Datenzugriff über den Database Manager des OS/2. Gleiches gilt für SAS. Die Benutzerschnittstelle ist der Presentation Manager des OS/2, einfache Berechnungen werden lokal, komplexere „per Knopfdruck“ im Zentralrechner durchgeführt, die Ergebnisse grafisch mit Hilfe des OS/2 dargestellt.

Dies sind natürlich nur Beispiele, kein vollständiger Migrationsplan. Sie zeigen, wie man sich die Arbeitsverteilung zwischen PC und Zentralrechner in Zukunft vorstellen kann. Dialoganwendungen sind unter TSO auch auf dem Zentralrechner möglich, sollten aber die Ausnahme sein und nur für fertig gekaufte Standardlösungen in Frage kommen.

Und was ist mit Unix? Momentan gibt es keine Unix-Zentralrechnerversion, die zu unserer Hardware paßt. Aber Abhilfe ist in Sicht. So wird die Universität als eine der ersten Installationen in Europa eine Test-AIX/ESA-Version bekommen, zeitgleich etwa mit der vollständigen Lieferung unseres Zentralrechners im Frühjahr. Dieses AIX/ESA kann die interessanten Hardware-Eigenschaften unseres Zentralrechners ausnützen, wie Multiprozessorfähigkeit, Unterstützung der Vektoreinrichtung und des Erweiterungsspeichers. Zudem ist es OSF/1-konform und wird sich relativ leicht in die Unix-Workstation-Welt der Universität einfügen.

Schlagwortartig läßt sich die mittelfristige Zukunft des Zentralrechners vielleicht so umschreiben: VM/CMS ist noch für eine Übergangszeit verfügbar. Die vier Prozessoren des Zentralrechners werden nach Bedarf zwischen AIX und MVS geteilt mit der erklärten Absicht, immer mehr Funktionen aus dem MVS in das AIX zu überführen, sobald es die entsprechenden Komponenten im AIX gibt. Zentral hierbei ist die Einbindung der Unix-Workstations.

Wenn Sie nun bei all diesen Prognosen und Absichtserklärungen konkrete Informationen vermissen, was es denn und wann es denn Neues vom System gibt, so möchte ich Sie auf die nächste Ausgabe des *inforuM* vertragen. Aber ich glaube, auch dieser Artikel ist nützlich für Sie, damit Sie ungefähr einschätzen können, wie wir die Entwicklung im Zentralrechnerbereich sehen.

Ende des UTS-Betriebs

von

R. Mersch

Am 16.12.91 wurde der Betrieb des UTS-Systems (UNIX auf dem Mainframe) eingestellt. Benutzern, die mit UNIX arbeiten wollen, stehen statt dessen UNIX-Rechner in einigen Fachbereichen der Universität sowie die RS/6000 im Rechenzentrum zur Verfügung. Letztere trägt den Namen Obelix und ist über telnet erreichbar (tnvt100 obelix).

Die Benutzer-Daten des UTS-Systems wurden archiviert. Die Daten derjenigen Benutzer, die dieses beantragt hatten, wurden nach Obelix übertragen. Sie befinden sich dort im Verzeichnis *uts* im jeweiligen Home-Verzeichnis. Für alle anderen

Benutzer besteht die Möglichkeit, dieses nachzuholen. Wenden Sie sich in diesem Fall an den Autor (Tel. 2488, URZ57@DMSWWU1A, mersch@obelix.uni-muenster.de). Sie brauchen dazu auf dem Zielrechner eine gültige Benutzerkennung, die Sie ggf. mit einem neuen Antragsformular (vgl. den Artikel „Benutzerverwaltung“ in diesem *inforuM*) beantragen müssen.

Bitte bedenken Sie, daß Sie Ihre Programme auf Obelix neu übersetzen müssen. Ferner sind einige Programmpakete (Prolog, S+, NAG) auf Obelix noch nicht verfügbar.

Benutzerverwaltung

von

St. Ost

Die neue Struktur der DV-Versorgung an unserer Universität erfordert eine einheitliche Verwaltung aller Benutzernamen.

Rechner, deren Ressourcen von mehr als einer Person genutzt werden, erfordern einen gewissen administrativen Aufwand. So soll nicht jede Datei von jedermann gelesen werden, und nicht jeder elektronische Brief öffentlich zugänglich sein. Aus diesem Grunde gibt es Kennungen, mit dem sich ein Rechnerbenutzer dem Betriebssystem des Rechners gegenüber ausweist. Dies ist wirklich nicht neu und dem Nutzer eines Mehrplatzsystems selbstverständlich. Neu dagegen (jedenfalls in diesen Ausmaßen) ist, daß es jetzt mit den Beschaffungen des letzten Jahres über die Universität verteilt eine dreistellige Zahl von Systemen gibt, in denen Benutzerkennungen verwaltet werden müssen. Ganz so schlimm, wie es klingt, ist es allerdings nicht, denn in der Regel lassen sich Rechner zu Rechnergruppen (Cluster) mit einer gemeinsamen Benutzerverwaltung zusammenfassen.

Im wesentlichen werden die Rechner von vier verschiedenen Betriebssystemen verwaltet. In alphabetischer Reihenfolge sind das:

- MVS** Steht für die Benutzerverwaltung des zentralen Rechners im Rechenzentrum.
- OS/2** Der LAN-Server und der Database Manager erfordern eine Benutzeridentifizierung.

Unix Meint AIX (IBM, sowohl im RISC- als auch im Zentralrechnerbereich), SunOS (SUN) und Ultrix (DEC).

VMS Läuft auf dem VAX-Cluster im naturwissenschaftlichen Zentrum.

Rechnet nun ein Benutzer auf mehr als einem System, so sollte er dies mit immer der gleichen Kennung tun dürfen. Eine universitätsweit einheitliche Benutzerkennung erleichtert zudem auch administrative Aufgaben. Zentrales Backup oder E-Mail sind prominente Beispiele. Von der Natur der Sache her ist eine universitätsweite Benutzerverwaltung eine zentrale Aufgabe, die vom Rechenzentrum wahrgenommen wird.

Nun ist es nicht so, als ob es bislang keine Benutzerverwaltung gegeben hätte. Im Zentralrechnerbereich ist diese seit Jahrzehnten im Gebrauch. Zum besserem Verständnis der Neuerungen sei zunächst das alte Verfahren dargestellt.

Grob gesprochen gliedert sich die Universität in Fachbereiche, diese wiederum in Institute. Hochschullehrer gehören einzelnen Instituten an und leiten die Mitarbeiter zur wissenschaftlichen Arbeit an. Dieses einfache Modell führt zu folgendem Schema: Die 21 Fachbereiche bekommen einen FB-Buchstaben zugeordnet, Fachbereich 1 den Buchstaben A, Fachbereich 2 den Buchstaben B und so fort. Institute werden mit zwei Zeichen gekennzeichnet. Diese werden an den FB-Buchstaben angehängt. Das Institut für medizinische Informatik und Biomathematik des Fachbereichs 5 (Medizin) bekommt so das Kürzel EMI, das Institut für Kernphysik des Fachbereichs 16 etwa das Kürzel PKP zugeordnet. Ein Hochschullehrer beantragt als Verantwortlicher beim Rechenzentrum eine Rechenerlaubnis für einen seiner Mitarbeiter. Dessen Benutzerkennung wird nun gebildet aus dem Kürzel des Instituts, dem der Hochschullehrer angehört, und zwei weiteren Ziffern, z. B. PKP39 oder EMI01.

Welche Gründe führten nun zur Ablösung eines zumindest aus systematischen Gründen überzeugenden Verfahrens?

1. Benutzerkennungen stehen für Menschen. Diese werden i. a. nicht mit einer Nummer, sondern mit ihrem Namen angesprochen. Man möchte eben auch bei seiner E-Mail-Adresse nicht als Nummer erscheinen.

2. Betriebssysteme bieten häufig die Möglichkeit, den Zugriff auf bestimmte Ressourcen auf Benutzergruppen einzuschränken. Nun wurden bisher in der Regel alle Mitarbeiter eines Instituts als eine Gruppe von Benutzern mit den gleichen Zugriffsrechten aufgefaßt. Dies mag für kleine Institute noch angängig sein, für die von der Personenzahl her großen Institute der naturwissenschaftlichen Fachbereiche ist es dagegen problematisch.
3. Neuere Entwicklungen wie die Gründung fachbereichsübergreifender wissenschaftlicher Zentren lassen sich mit dem alten Schema nur ungenügend abbilden.

In Zusammenarbeit mit Mitgliedern der Fachbereiche Physik und Chemie anlässlich der Inbetriebnahme der dortigen Fachbereichsrechner wurde ein neues Schema zur Verwaltung von Benutzerkennungen entwickelt und angewandt. Kern des neuen Schemas ist die Einführung von *sprechenden* Benutzerkennungen. Die Eckpunkte des neuen Schemas sind:

1. Ein Benutzer gehört zumindest einem Projekt an. Ein Projekt wird von einem Hochschullehrer beantragt (verwenden Sie bitte den „Antrag für DV-Projekte in der Universität“). Dabei wird definiert, auf welche Rechner und sonstige Ressourcen Mitarbeiter dieses Projektes Zugriff haben sollen. Die Benutzung der zentralen Rechner im Rechenzentrum braucht nicht eigens beantragt zu werden, sie ist automatisch in jedem Projektantrag enthalten. Da Zugehörigkeit zu einem Projekt Zugriffsrechte auf Ressourcen (Rechner, Dateien, Programme, Drucker, usw.) bedeutet, wird man bei der Projektdefinition (welche Mitarbeiter an welchem Projekt) weniger die Struktur der Arbeitsgruppe abbilden wollen als pragmatische Gesichtungspunkte (Ist es notwendig, die Mitarbeiter in ihren Zugriffsrechten zu unterscheiden?) berücksichtigen. Es können sich auch mehrere Hochschullehrer innerhalb eines Fachbereiches als auch fachbereichsübergreifend zu einem Projekt zusammenschließen. Der Hochschullehrer benennt einen Projektleiter, der die weitere Projektverwaltung übernimmt (in der Regel ist dies ein wissenschaftlicher Mitarbeiter).
2. Projektnamen können maximal 8 alphanumerische Zeichen lang werden. Projektnamen müssen sich von Benutzerkennungen unterscheiden; einige Betriebssysteme erfordern dies. Deshalb gibt es folgende zusätzliche Regel zur Bil-

dung von Projektnamen. Projektnamen beginnen mit dem oben eingeführten Fachbereichsbuchstaben, gefolgt von einer im Prinzip frei wählbaren Ziffer, in der Regel die Ziffer Null. Die restlichen 6 Zeichen sind vom Hochschullehrer frei wählbar, müssen nur innerhalb des Fachbereichs eindeutig sein. So steht der Name AOMEYER für ein Projekt des Fachbereichs 1 (also A) und meint die Mitarbeiter der Arbeitsgruppe Meyer. Um es aber noch einmal zu betonen: Die Projektdefinition regelt den Zugriff auf Betriebsmittel, und bildet *nicht* notwendigerweise die Gruppenstruktur der Mitarbeiter des Hochschullehrers ab.

3. Beantragt ein Benutzer die „Nutzung von Rechnern und DV-Diensten in der Universität“ (auch dieses Formular erhalten Sie im Rechenzentrum), so gibt er neben den Angaben zu seiner Person auch das Projekt an, mit dem er verbunden werden will. Den Projektnamen erfährt er vom Projektleiter, der auch den Antrag unterschreibt. Er wird vom Rechenzentrum (d. d. Hd. des Projektleiters) schriftlich informiert, auf welche Rechner er nun Zugriff hat, wie sein Eröffnungspasswort lautet und über andere evtl. geltenden Spielregeln.
4. Die Benutzerkennung wird aus dem Nach- und dem Vornamen des Antragstellers gebildet. Er ist maximal 7 Zeichen lang und darf nur aus alphanumerischen Zeichen bestehen, von denen die ersten beiden alphabetisch sein müssen. Umlaute oder ähnlich ausgezeichnete Vokale werden auf den jeweiligen Basisvokal abgebildet. Sind die ersten 7 Zeichen des Nachnamens nicht eindeutig, so wird durch Anhängen der ersten Buchstaben des Vornamens versucht, eine eindeutige Kennung zu erzeugen.
5. Für Studenten, die noch keine Studienabschlußarbeit schreiben und somit noch an keinem Fachbereichsprojekt teilnehmen können, wird ein allgemeines Projekt U0STUD des Rechenzentrums eingerichtet, über das sie die zentralen Rechner des Rechenzentrums benutzen können.

Das hier beschriebene Verfahren wird seit Beginn des Wintersemesters für die Rechner der Fachbereiche Physik und Chemie sowie für den AIX-Rechner des Rechenzentrums („Obelix“) angewendet. Mit der Einführung der neuen Betriebssystemversionen im Zentralrechnerbereich werden auch dort die bisherigen systematischen Nutzerkennungen abgelöst. Über den Zeitpunkt und die Modalitäten der Umstellung informieren wir Sie rechtzeitig.

Kritik an dem neuen Schema wendet sich vor allem an die Art und Weise, wie die Benutzerkennungen generiert werden. Die einen haben sich an ihre bisherige Kennung gewöhnt und möchten sie behalten, die anderen finden ihren Nachnamen verstümmelt wieder und möchten lieber ihren Vornamen und noch andere finden ihre Initialen am schönsten. Warum also generieren wir die Benutzerkennung automatisch und lassen nicht den Benutzer eine Wunschkennung angeben? Die Antwort hängt einfach mit der erwarteten Anzahl von Benutzeranträgen zusammen. Bisher hatten wir im Zentralrechnerbereich etwa 1600 Benutzer, wir erwarten jetzt universitätsweit die etwa vier- bis fünffache Anzahl. Sollen wir mit jedem Benutzer, dessen Wunsch wir nicht erfüllen können, telefonieren und zusammen mit ihm eine Kennung herausuchen, die seinen Vorstellungen entspricht und noch nicht vergeben ist? Dies erscheint nicht praktikabel. Eventuell kann zu einem späteren Zeitpunkt Abhilfe geschaffen werden, wenn der Benutzerantrag nicht mehr schriftlich, sondern via E-Mail gestellt werden kann, aber das ist Zukunftsmusik.

Neues vom SPSS

von

S. Zörkendörfer

Auf dem Zentralrechner gibt es eine neue SPSS-Version. Auch für den PC sind Verbesserungen angekündigt.

Zu unserer MVS-Großrechnerlizenz des „Statistical Package for the Social Sciences“ SPSS^X ist mit der Bezeichnung „SPSS Release 4.1“ eine neue Version ausgeliefert worden, sie kann über die katalogisierte Prozedur SPSSX41 aufgerufen werden. Zu dieser Version 4 ist ein Satz englischsprachiger Original-Handbücher aufgelegt: *SPSS Reference Guide*, *SPSS Base Systems User's Guide*, *SPSS Advanced Statistics User's Guide*. Mit dieser Version wird ferner kurzfristig eine Probeinstallation des Zusatzproduktes *SPSS Categories* (conjoint analysis, optimal scaling, correspondence analysis, homogeneity analysis by alternating least squares, nonlinear component analysis, nonlinear canonical correlations) angeboten, Anregungen zur langfristigen Anmietung nehmen wir entgegen. Ins SPSS^X ist weiterhin LISREL integriert, jetzt mit der Version 7, neu erschienen sind die zugehörigen Handbücher *SPSS LISREL 7*

and *PRELIS, User's Guide and Reference* sowie *LISREL 7, A Guide to the Program and Applications, 2nd Edition*. Schließlich sei aufmerksam gemacht auf eine neue Ausgabe der *SPSS Statistical Algorithms*. Bisher konnten wir die neue Version noch nicht ausgiebig erproben und in der gewohnten Form zur Verfügung stellen, insbesondere konnten die Anbindung der USERPROC CLUSTER und die Aufbereitung des LOKALEN INFO noch nicht erfolgen.

Für das über die Hochschul-Landeslizenz auf Universitäts-PCs angemietete MS-DOS-Paket SPSS/PC+ ist zur Version 4.0 eine verbesserte Version 4.0.1 angekündigt. Eine Liste der Nachbesserungen kann bei mir eingesehen werden. Im Software-Verteilungssystem VAMP werden wir die neu ausgelieferten Disketten (voraussichtlich Un, Bn, Tn) am Etikett mit der Versionsnummer 4.0.1 kennzeichnen, die übrigen Disketten behalten die Versionsbezeichnung 4.0.

Thematische Karten unter SAS

von

S. Zörkendörfer

Für die Erzeugung thematischer Karten mit SAS gibt es eine neue Koordinatendatei für Deutschland (1990) auf Kreisbasis. Auch andere Karten wurden aktualisiert.

Unsere CMS-Großrechnerinstallation SAS Version 6.06 enthält mit dem Produkt SAS/GRAPH umfangreiche SAS-Koordinatendateien zur Einbindung thematischer Karten in Präsentationsgrafiken. Der SAS-Anwender kann sich einen Überblick mit dem Directory-Kommando DIR zur Libref MAPS verschaffen, der CMS-Kundige kann nach „ENV SAS“ eine Dateiliste durch „FL * MAPS *“ anfordern. Im Rahmen der ständigen Wartungsarbeiten konnten wir folgende sechs Dateien zufügen: BRDKRS mit 16687 Koordinatenbeobachtungen und BRDKRS2 mit 544 Kreisnamenbeobachtungen für die Bundesrepublik Deutschland in den Kreisgrenzen, BRDLAN mit 3731 Koordinatenbeobachtungen und BRDLAN2 mit 16 Ländernamenbeobachtungen für die Bundesrepublik Deutschland auf Länderebene, EUROPA mit 8741 Koordinatenbeobachtungen und EUROPA2 mit 27 Staatennamenbeobachtungen für die europäischen Staaten. Insbesondere zur letzten Einteilung werde ich bei Bedarf eine von Benutzern bereitgestellte aktuellere Datei gerne SAS-gemäß aufbereiten.

Tips zur Schadensvermeidung bei PCs

In letzter Zeit vermehrt aufgetretene Schadensfälle im Zusammenhang mit PCs veranlassen uns, folgende allgemeine Hinweise zur Vermeidung von Schäden zu geben.

Computer der WWU können nicht gegen Diebstahl oder andere Schäden versichert werden. Entstandene Schäden müssen daher i. a. von der WWU oder dem nutzenden Institut behoben werden.

Nachfolgend sind deshalb einige Anregungen aufgeführt, die den Diebstahl erschweren und andere Schäden vermeiden helfen.

Diebstahlschutz

- Es bietet sich eine elektronische Sicherung aller Komponenten eines DV-Systems mit Alarmanlagen an. Eine Lösung wird dazu vom Universitätsrechenzentrum angeboten. Der Nachbau dieser Anlage und die Installation könnte von einigen hochschuleigenen Werkstätten geleistet werden; eine Installation durch das Universitätsrechenzentrum ist z. Z. nicht möglich. Die Kosten belaufen sich bei Selbstinstallation auf 2.000 bis 3.000 DM für 20 bis 30 Geräte.

Elektronische Sicherungen können auch im Handel gekauft und von Betrieben installiert werden. Dies dürfte aber i. a. deutlich mehr Haushaltsmittel erfordern und ist in der bisher installierten Form nur schwer handhabbar.

- In manchen Fällen können auch mechanische Sicherungen (Ketten, Metallbänder) sinnvoll sein. Entsprechende Produkte können im Handel erworben werden. Diese Form der Sicherung kann aber leider mit einem Bolzenschneider leicht umgangen werden.
- In anderen Fällen mag auch eine Kamera-Überwachung eines Raumes sinnvoll sein.
- Eine Aufsicht durch studentische Hilfskräfte wird an einigen Stellen ebenfalls erfolgreich eingesetzt.

Gegebenenfalls kann man sich bei der Polizei über adäquate Sicherungsmaßnahmen beraten lassen. Es ist zu bedenken, daß mit dem Diebstahl eines Computers auch wichtige Daten verschwinden können.

Schadensvermeidung und -begrenzung bei Wassereinwirkung

- Computer sollten möglichst nicht in Kellerräumen oder in Räumen, die durch Wasserrohrbrüche z. B. in darüberliegenden Etagen gefährdet sind, untergebracht werden. Gegebenenfalls müssen die Geräte auf Sockel oder Tische gestellt werden. Feuchtigkeitsmelder könnten im Notfall an geeigneter Stelle Alarm auslösen.
- Sind Geräte feucht oder naß geworden, so machen Sie die Geräte stromlos und sorgen umgehend für eine Trocknung. Nicht mehr behebbare Schäden treten oft erst nach längerer Einwirkung von Feuchtigkeit auf.

Gefahren durch Blitzschlag

Computer und periphere Geräte sind in hohem Maße durch Blitzschlag gefährdet. Wenn der Hersteller keine anderen Vorgaben macht, sollte bei Gewitter nicht an Arbeitsplatzrechnern oder Workstations gearbeitet werden. Die Geräte sollten ausgeschaltet und der Stecker möglichst gezogen werden. Eine sehr geringe Gefährdung des Gerätes kann u. U. auch vom lokalen Rechnernetz ausgehen, eine Abschaltung ist hier aber nicht möglich.

Vermeidung von Brandschäden

Durch einen Kurzschluß im Rechner oder Ausfall des Lüfters kann ein Brand entstehen. Geräte, die momentan nicht genutzt werden, sollten deshalb abgeschaltet werden. Insbesondere gilt dies für die Nachtstunden. Bei Geräten, die ganztägig verfügbar sein müssen – z. B. bei Servern – ist über Nacht zumindest der Monitor abzuschalten.

Datenverluste

- Der Diebstahl personenbezogener Daten kann erhebliche Konsequenzen haben. Beachten Sie bei der Verarbeitung personenbezogener Daten die Datenschutzgesetze.

Personalien

Herr Dipl.-Phys. M. Hort, der als wissenschaftliche Hilfskraft im Bereich „Rechnernetze“ tätig war, hat uns zum 31.10.1991 verlassen.

Bei den studentischen Mitarbeitern haben sich folgende Veränderungen ergeben:

Ausgeschieden sind Herr A. Ibach zum 31.10.1991 und Frau K. Sievert zum 30.11.1991. Die **inforuM**-Redaktion dankt Herrn Ibach besonders für seine engagierte Mitarbeit bei der Erstellung von 11 Ausgaben und gratuliert zu seinem bestandenen Diplom in Physik.

Anfang Januar 1992 haben die Herren T. Keßler und R. Wilmes ihre Tätigkeit bei uns aufgenommen.

- Da Datenverluste (Diebstahl, Magnetplattenfehler, fehlerhafte Handhabung) sehr teuer werden können, sollte für eine angemessene Datensicherung gesorgt werden. Das Universitätsrechenzentrum wird eine zentrale Sicherung über das Rechnernetz anbieten und informiert Sie gern hierüber. Wenn Sie selber sichern, sorgen Sie u. a. dafür, daß Ihre Kopien an einem sicheren Ort aufbewahrt werden.
- Sogenannte Computer-Viren können besonders Ihre MS-DOS-Rechner gefährden. Das Universitätsrechenzentrum kann Ihnen Lösungen anbieten, die geeignet sind, eine Gefährdung durch Viren so früh wie möglich zu erkennen und damit mögliche Schäden zu begrenzen. Lassen Sie sich beraten.

RUM-Tutorial

Ein neuer Schriftauswahlmechanismus für L^AT_EX

von

W. Kaspar

Allen L^AT_EX-Anwendern, die sich einen flexibleren Mechanismus zur Schriftaus- und -anwahl wünschen, kann nun geholfen werden. Es gibt nämlich eine L^AT_EX-Erweiterung, mit der eine größere Auswahl an unterschiedlich gestalteten Schriftzeichen zur Verfügung gestellt werden kann, die einen einfachen Wechsel der Schrift für den gesamten Text (z.B. Sans Serif anstatt Roman) ermöglicht und die – für den Entwickler von L^AT_EX-Stilen – einen einfachen Mechanismus zum Einbinden neuer Fonts bietet.

Neben den L^AT_EX-Basiskomponenten steht in der aktuellen T_EX-PC-Installation des Universitätsrechenzentrums auch ein neuer Schriftauswahlmechanismus (New Font Selection Scheme, abgekürzt auch NFSS) zur Verfügung. Er wurde von Mittelbach und Schöpf Anfang 1990 in einer Vorläuferversion veröffentlicht und ist Teil einer umfangreichen Überarbeitung des L^AT_EX-Makropaketes.

Auch wenn sich Teile dieser Vorläuferversion noch ändern können, lohnt sich schon jetzt der Einsatz dieser Version. Hinweise zur Installation finden Sie in [1, S. 16] unter der Zusatzkomponente L^AT_EXFS.

Beispiel zum bisherigen L^AT_EX

Die Vorteile des neuen Schriftauswahlmechanismus gegenüber dem bisherigen Konzept lassen sich am besten an ein paar Beispielen demonstrieren.

Der folgende Text sei ein Ausschnitt der Datei SDEMO.TEX.

```
{\sf\bf Mustertext f"ur den neuen
Schriftauswahlmechanismus}
```

```
{\it\tt Mustertext f"ur den neuen
Schriftauswahlmechanismus}
```

```
{\bf\it Mustertext f"ur den neuen
Schriftauswahlmechanismus}
```

Mit dem Befehl `tex sdemo` würde dieser Text folgendermaßen gesetzt:

Mustertext für den neuen
Schriftauswahlmechanismus

Mustertext für den neuen
Schriftauswahlmechanismus

Mustertext für den neuen
Schriftauswahlmechanismus

Das obige Beispiel führt uns noch einmal das bisherige Verhalten von L^AT_EX vor Augen. Von den beiden Markierungen `\sf\bf` hebt die letzte (`\bf`) die Wirkung der vorhergehenden (`\sf`) auf. Wir erhalten einen fett gesetzten Text, aber nicht, wie von uns vielleicht gewünscht, in serifenlosen Schriftzeichen. Analog ist das Verhalten auch bei `\it\tt` und `\bf\it`. Wir erhalten keine kursiven, sondern normale Schreibmaschinenzichen und auch nur einfache Kursivzeichen anstelle der fetten.

Dies liegt offensichtlich daran, daß z.B. mit `\bf` nicht nur auf fette Schriftzeichen, sondern auch auf die Standardzeichen mit Serifen umgeschaltet wird. Die Markierung `\sf` im obigen Beispiel zur Umschaltung auf serifenlose Zeichen wurde hiermit wieder außer Kraft gesetzt. Ähnlich verhält sich das bisherige L^AT_EX auch bei den anderen Markierungen, mit denen das Aussehen der Schriftzeichen beeinflußt werden kann.

Beispiel zum neuen Auswahlmechanismus

Wir wollen nun das obige Beispiel zusammen mit einem L^AT_EX formatieren, das den neuen Schriftauswahlmechanismus enthält. Hierzu fügen wir beim Aufruf von T_EX als zweites Argument `latexgfs` hinzu:

`tex sdemo latexgfs`

L ^A T _E X- Markierung	Geänderte Merkmale der Schriftzeichen nach Absetzen der Markierung				
	Schrift- größe*[pt]	Schriftschnitt			Schriftfamilie
		Breite	Stärke	Sonstiges	
<code>\bf</code>		breit	fett	normal	Computer Modern Roman
<code>\it</code>		normal	normal	kursiv	Computer Modern Roman
<code>\sc</code>		normal	normal	Kapitälchen	Computer Modern Roman
<code>\sl</code>		normal	normal	gekippt	Computer Modern Roman
<code>\sf</code>		normal	normal	normal	Computer Modern Sans Serif
<code>\tt</code>		normal	normal	normal	Computer Modern Typewriter
<code>\rm</code>		normal	normal	normal	Computer Modern Roman
<code>\small</code>	9	normal	normal	normal	Computer Modern Roman
<code>\large</code>	12	normal	normal	normal	Computer Modern Roman
<code>\LARGE</code>	17	normal	normal	normal	Computer Modern Roman

*Die hier eingetragenen Werte beziehen sich auf die standardmäßig eingestellte Grundschriftgröße von 10pt. Wenn in der `\documentstyle`-Markierung mit den Teilstilen 11pt oder 12pt eine andere Grundschriftgröße eingestellt wird, ändern sich die Schriftgrößenwerte, die den einzelnen L^AT_EX-Markierungen zugeordnet sind.

Tabelle 1: Auswirkungen einiger L^AT_EX-Markierungen auf die Merkmale der Schriftzeichen in der bisherigen Standard-L^AT_EX-Version (Formatdatei L^AT_EXG).

L^AT_EXGFS ist der Name einer T_EX-Formatdatei. Formatdateien enthalten Informationen über Trennmuster, Fonts und Befehle, die von T_EX zur Formatierung eines Textes benötigt werden. Standardmäßig wird bei unserem `tex`-Befehl die Formatdatei L^AT_EXG eingebunden, in der auch die Wirkung der im obigen Beispiel verwendeten L^AT_EX-Markierungen `\sf`, `\bf` usw. festgelegt ist. Da in L^AT_EXGFS – das G steht für „German“ und FS für „New Font Selection Scheme“ – diese Markierungen für den neuen Schriftauswahlmechanismus verändert wurden, wird unser Beispieltex nun deutlich anders als bisher gesetzt:

Mustertext für den neuen
Schriftauswahlmechanismus

Mustertext für den neuen
Schriftauswahlmechanismus

Mustertext für den neuen
Schriftauswahlmechanismus

Wie wir sehen, wird nun z.B. durch `\bf` nicht mehr die Wirkung von `\sf` aufgehoben. Die Umschaltung auf die serifenlosen Schriftzeichen bleibt also bestehen, auch wenn danach noch zusätzlich auf fette Zeichen umgeschaltet wird. Im obigen Beispiel werden also nach den Markierungen `\sf\bf` fette, serifenlose Schriftzeichen nach `\it\tt` kursive Schreibmaschinenzichen und nach `\bf\it` fette Kursivzeichen verwendet.

Merkmale von Schriftzeichen

Um systematisch die verschiedenen Schriftzeichen mit den zum Umschalten benötigten L^AT_EX-Markierungen beschreiben zu können, wollen wir jedem Schriftzeichen die drei Merkmale *Schriftgröße*, *Schriftschnitt* und *Schriftfamilie* zuordnen.

Die *Schriftgröße* (engl. *typesize*) ist ein Maß für die maximale Höhe der Schriftzeichen. Jeder Schriftgröße ist auch ein Grundlinienabstand zugeordnet. Er ist normalerweise etwas größer als die Schriftgröße und legt den Abstand zwischen zwei aufeinanderfolgenden Zeilen fest. So werden z.B. die Absätze dieses Artikels in einer Schriftgröße von 10 pt und einem Grundlinienabstand von 12 pt gesetzt, wobei „pt“ für „Point“ steht, der kleinsten Einheit eines typographischen Maßsystems, das vor allem in angelsächsischen Ländern verwendet wird (1 pt entspricht 0,351 mm).

Der *Schriftschnitt* (*type style*) beschreibt die *Breite* (*width*), *Stärke* (*weight*) und sonstige *Gestalt* (*shape*) des Schriftzeichens. Bezeichnungen für die Breite sind u.a. „eng“, „normal“ oder „breit“, für die Stärke „mager“, „normal“ oder „halbfett“ und für die sonstige Gestalt „kursiv“ oder „Kapitälchen“. Beispiele:

L ^A T _E X-Markierung	Geänderte Merkmale der Schriftzeichen nach Absetzen der Markierung			
	Schriftgröße*[pt]	Schriftschnitt		Schriftfamilie
		Breite	Stärke	
<code>\bf</code>		breit	fett	
<code>\it</code>				kursiv
<code>\sc</code>				Kapitälchen
<code>\sl</code>				gekippt
<code>\sf</code>				Computer Modern Sans Serif
<code>\tt</code>				Computer Modern Typewriter
<code>\rm</code>				Computer Modern Roman
<code>\small</code>	9			
<code>\large</code>	12			
<code>\LARGE</code>	17			

*Die hier eingetragenen Werte beziehen sich auf die standardmäßig eingestellte Grundschriftgröße von 10pt. Wenn in der `\documentstyle`-Markierung mit den Teilstilen 11pt oder 12pt eine andere Grundschriftgröße eingestellt wird, ändern sich die Schriftgrößenwerte, die den einzelnen L^AT_EX-Markierungen zugeordnet sind.

Tabelle 2: Auswirkungen einiger L^AT_EX-Markierungen auf die Merkmale der Schriftzeichen mit dem neuen Schriftauswahlmechanismus (Formatdatei L^AT_EXGFS).

Dieser Text wurde in einem normalen kursiven Schriftschnitt gesetzt.

Dieser Text wurde in einem fetten Schriftschnitt gesetzt.

Dieser Text wurde in einem fetten und breiten Schriftschnitt gesetzt.

Dieser Text wurde in einem fetten, breiten und kursiven Schriftschnitt gesetzt.

DIESER TEXT WURDE IN KAPITÄLCHEN GESETZT.

Die Schriftfamilie (*type family*) gibt Auskunft über das zugrundeliegende Schriftdesign. Bezeichnungen hierfür sind „Times“, „Helvetica“, „Futura“ und „Computer Modern“. In der letztgenannten ist zum Beispiel diese inforum-Ausgabe gesetzt. Weitere Beispiele:

Dieser Text wurde in der „Computer Modern Sans Serif“-Schriftfamilie gesetzt.

Dieser Text wurde in der „Computer Modern Typewriter“-Schriftfamilie gesetzt.

Umschaltung von Schriftzeichenmerkmalen mit `\it`, `\bf` etc.

Wie diese Merkmale im bisherigen Standard-L^AT_EX und unter dem neuen Schriftauswahlmechanismus eingestellt werden können, zeigen die Tabellen 1 und 2. Wir sehen, daß nur der neue Schriftauswahlmechanismus eine sinnvolle Kombination der in den Tabellen aufgeführten L^AT_EX-Markierungen ermöglicht. So schaltet z.B. in Tabelle 2 `\it` auf einen kursiven Schriftschnitt, ändert aber nicht die Breite, Stärke und Familie der Schriftzeichen. Hingegen werden in Tabelle 1 durch `\it` alle Merkmale bis auf die Schriftgröße beeinflusst.

Noch extremer sind die Auswirkungen der Schriftgrößenumschaltung im bisherigen L^AT_EX. So werden z.B. durch `\large` bisher alle Merkmale des Schriftschnittes auf normal und die Familie auf „Computer Modern Roman“ zurückgesetzt (Tabelle 1). Unter dem neuen Schriftauswahlmechanismus (Tabelle 2) wird jedoch nur die Schriftgröße umgeschaltet. Alle anderen Merkmale werden unverändert übernommen.

In Tabelle 3 sind alle Schriftschnitte und -familien aufgeführt, die sich unter dem neuen Schriftauswahlmechanismus mit Standard-L^AT_EX-Markierungen anwählen lassen. Für alle nicht aufgeführten Kombinationen sind keine Schriftzeichen vorhanden. Trifft

L ^A T _E X-Markierung	Merkmale der Schriftzeichen
<code>\rm</code>	Computer Modern Roman
<code>\rm\bf</code>	Computer Modern Roman, breit, fett,
<code>\rm\bf\it</code>	Computer Modern Roman, breit, fett, kursiv
<code>\rm\bf\sl</code>	Computer Modern Roman, breit, fett, gekippt
<code>\rm\it</code>	Computer Modern Roman, kursiv
<code>\rm\sl</code>	Computer Modern Roman, gekippt
<code>\rm\sc</code>	COMPUTER MODERN ROMAN, KAPITÄLCHEN
<code>\sf</code>	Computer Modern Sans Serif
<code>\sf\sl</code>	Computer Modern Sans Serif, gekippt
<code>\sf\bf</code>	Computer Modern Sans Serif, breit, fett
<code>\tt</code>	Computer Modern Typewriter
<code>\tt\it</code>	Computer Modern Typewriter, kursiv
<code>\tt\sl</code>	Computer Modern Typewriter, gekippt
<code>\tt\sc</code>	COMPUTER MODERN TYPEWRITER, KAPITÄLCHEN

Tabelle 3: Aufstellung aller Schriftschnitte und -familien, die unter dem neuen Schriftauswahlmechanismus (Formatdatei L^AT_EXGFS) mit Standard-L^AT_EX-Markierungen angewählt werden können.

L^AT_EX bei der Formatierung auf eine solche Kombination, so wird ein Ersatz gewählt.

Wurden nur die L^AT_EX-Basiskomponenten ohne weitere zusätzliche Schriften installiert, so sind einige der Schriftzeichen nicht in allen Schriftgrößen vorhanden. Auch hier werden automatisch Ersatzzeichen gesetzt. In Kürze wird ein L^AT_EX-Zusatzstil zur Verfügung stehen, der zusammen mit der Schriftkomponente RESTCM die noch fehlenden Schriftgrößen zugänglich macht.

Formatieren alter L^AT_EX-Texte

Die meisten L^AT_EX-Anwender werden die Markierungen `\bf` in der Form „`{\bf ... }`“ und nicht als „`\bf ... \rm`“ verwendet haben. Für sie ist die folgende Markierung, die zusammen mit dem neuen Schriftauswahlmechanismus zur Verfügung gestellt wird, nur in Ausnahmefällen interessant.

Die Markierung `\mediumseries`¹ schaltet nämlich explizit auf die normale Breite und Stärke der Schriftzeichen zurück. Falls wir in unserem Text Markierungen der Form „`\bf ... \rm`“ benutzt haben und ihn nun zum ersten Mal mit der Formatdatei L^AT_EXGFS

¹Unter dem engl. Begriff *series* werden im neuen Schriftauswahlmechanismus die beiden Merkmale Breite und Stärke zusammengefaßt.

formatieren wollen, muß diese neue Markierung anstelle von `\rm` eingefügt werden; denn die Markierung `\rm` schaltet unter dem neuen Schriftauswahlmechanismus nicht mehr den normalen Schriftschnitt ein, sondern beeinflusst nur noch die Schriftfamilie (siehe `\rm` in Tabelle 2). Der obige Textauszug muß also lauten: „`\bf ... \mediumseries`“.

Entsprechendes gilt für die Markierungen `\it`, `\sc` und `\sl`. Hier müssen wir, falls `\rm` zum Wiederherstellen des normalen Schriftbildes benutzt wurde, `\normalshape` anstelle von `\rm` angeben.

Um es noch einmal deutlich zu wiederholen: Alle Anwender, die in ihren Texten Markierungen der Form „`{\bf ... }`“, „`{\it ... }`“ usw. verwendet haben, müssen *keine* Änderungen vornehmen, da diese Schreibweise unter dem alten und neuen Schriftauswahlmechanismus die gleiche Wirkung hat.

Zwei weitere Formatierungsprobleme ergeben sich daraus, daß der Schriftauswahlmechanismus als Teil des neuen L^AT_EX, an dem zur Zeit noch gearbeitet wird, entwickelt wurde. Sie werden nach dem Einsatz der neuen Version verschwinden [2, S. 298].

Das erste Problem kann bei L^AT_EX-Markierungen, wie z.B. `\footnote`, auftreten, bei denen automatisch

die Schriftgröße (z.B. für den Fußnotentext) umgeschaltet wird. Bisher wurde dabei auch der Schriftschnitt und die Familie auf die normalen Einstellungen zurückgesetzt. Unter dem neuen Schriftauswahlmechanismus wird jedoch nur noch die Schriftgröße verändert. Hieraus folgt, daß z.B. der Text, einer Fußnote kursiv gesetzt wird, wenn sie innerhalb eines kursiven Textes abgesetzt wird. Solange das neue $\text{\LaTeX}$ noch nicht zur Verfügung steht, können wir dieses Verhalten dadurch korrigieren, indem wir zu Beginn des Fußnotentextes die Markierung `\normalshape` einfügen.

Das zweite Problem betrifft nur die `\cal`-Markierung im Mathematiksatz. Soll z.B. ein $\mathcal{A}$ gesetzt werden, so mußte bisher `\cal A` eingegeben werden. Unter dem neuen Schriftauswahlmechanismus wird jedoch die Markierung `\cal{A}` erwartet. Um unseren Text hieran anzupassen, können wir entweder die Markierungen ändern, oder wir geben `nomargid2` als Teilstil in der `\documentstyle`-Markierung an. Dieser Teilstil wird vom neuen Schriftauswahlmechanismus zur Verfügung gestellt und schaltet die alte Eingabekonvention wieder ein [2, S. 301].

Unverträglichkeiten mit Stildateien

Alle $\text{\LaTeX}$ -Stile, die bestimmte Mechanismen der bisherigen Schriftauswahl benutzen oder modifizieren, werden naturgemäß Probleme mit dem neuen Schriftauswahlmechanismus haben. Die $\text{\LaTeX}$ -Standardstile (außer dem `letter`-Stil, siehe hierzu [3]) sind hiervon nicht betroffen. Alle übrigen zusätzlichen $\text{\LaTeX}$ -Stile müssen auf ihre Verträglichkeit hin noch überprüft werden. Hier sollten wir mit unerwünschten Nebenwirkungen rechnen, wenn wir sie zusammen mit dem neuen Mechanismus benutzen.

Zur Zeit sind uns folgende unverträgliche $\text{\LaTeX}$ -Stile bekannt:

`letter`, `letterg` und `doublesp`

Für alle Leser, die sich selbst mit dem Schreiben von Styles befassen, möchte ich kurz ein paar interne $\text{\LaTeX}$ -Befehle notieren, die vom neuen Schriftauswahlmechanismus modifiziert werden. Es sind dies

```
\@begintheorem,
\@opargbegintheorem,
\@outputpage,
\document
```

²Der Name `nomargid` steht als Abkürzung für „no math argument identifier“. Dies ist die Kurzform von „math alphabet identifier with no argument“. Ein „math alphabet identifier“ ist z. B. `\cal`, mit dessen Hilfe im mathematischen Modus Zeichen aus dem kalligraphischen Alphabet gesetzt werden können. Bei der Schreibweise `\cal{A}` wird der in kalligraphischen Zeichen zu setzende Text als Argument (`{A}`) eingegeben. In der bisherigen $\text{\LaTeX}$ -Version hat `\cal` kein Argument (`{\cal A}`).

und natürlich alle in `LFONTS.TEX` definierten Befehle. Alle Stil-Dateien, in denen diese Befehle geändert werden, müssen überarbeitet werden.

Im nächsten *inforuM* werden wir weitere Möglichkeiten des neuen Schriftauswahlmechanismus kennenlernen.

Literatur

Die Artikel [1] bis [3] sind in unserem $\text{\TeX}$ -Installationspaket enthalten. [1] befindet sich im Archiv `TDR.ZOO` (Datei `TDRI.DVI`) auf der `SETUP-1`-Diskette. [2] und [3] werden bei der Installation der `LATEXFS`-Komponente im Verzeichnis `\TEX30\INITEX\FONTSEL` abgelegt. Der Artikel [2] befindet sich in der Datei `FONTSEL.TEX` bzw. `FONTSEL.DVI`.

- [1] WOLFGANG KASPAR: *$\text{\TeX}$ - Installationshinweise für den PC*, Westfälische Wilhelms-Universität Münster, Universitätsrechenzentrum, Softwareinformation, 1991.
- [2] FRANK MITTELBACH UND RAINER SCHÖPF: *The new Font Family Selection — User Interface to Standard $\text{\LaTeX}$, TUGboat 11 1990*, pp. 297–305.
- [3] FRANK MITTELBACH UND RAINER SCHÖPF: *Datei FONTSEL.BUG, Bestandteil des Fontsel-Softwarepaketes.*

Das X-Window-System

von

R. Mersch

Im Bereich grafischer Benutzerschnittstellen hat in letzter Zeit das X-Window-System stark an Bedeutung gewonnen. Dies ist sicherlich auch eine Folge der zunehmenden Verbreitung von UNIX-Systemen, wenngleich X-Window-Systeme nicht nur im UNIX-Umfeld verwendet werden. Dieser Artikel gibt einen kurzen Überblick über das X-Window-System und versucht, die wichtigsten Begriffe zu klären.

Das X-Window-System, häufig kurz **X** genannt, ist ein System für grafisch-interaktive Bedienoberflächen. Basierend auf dem X-Protokoll, einem netzwerktransparenten, rechner- und herstellerunabhängigen Kommunikations-Protokoll, ermöglicht es **X**, daß mehrere Anwendungen, die u. U. auf unterschiedlichen Rechnern laufen, sich ein und desselben grafischen Arbeitsplatzes zur Interaktion mit dem Benutzer bedienen.

Netzwerktransparenz bedeutet hier, daß die Kommunikation zwischen Anwendung und grafischem Arbeitsplatz über beliebige Transportmechanismen erfolgen kann, etwa mittels Interprozeßkommunikation innerhalb eines Systems oder unter Verwendung von Transportdiensten (z.B. TCP/IP oder DECnet) in Netzen.

Die geforderte Hardware-Unabhängigkeit bedeutet, daß die Anwendungen idealerweise nichts über die Hardwareeigenschaften des grafischen Arbeitsplatzes wissen müssen. Sie sollten sich unterschiedlichster grafischer Arbeitsplätze bedienen können, ohne daß dazu die Anwendung angepaßt werden muß. Dies wird erreicht durch den X-Server, ein auf dem Arbeitsplatz ablaufendes Programm, das diesen verwaltet und die jeweiligen Hardware-Eigenschaften weitgehend verbirgt. Die Anwendung, der X-Client, schickt seine Aufträge an den X-Server, der diese dann mit den vorhandenen Mitteln realisiert. Natürlich kann der X-Server nicht alle Hardwareeigenschaften verbergen, so z. B. nicht die Fähigkeit, Farben darzustellen. Solche Eigenschaften können dann aber erfragt werden, sollten sie für die Anwendung wichtig sein.

Dem X-Window-System liegt also ein *Client/Server-Modell* zugrunde, wobei die Begriffsbildung hier auf

den ersten Blick überraschen mag: Man neigt dazu, mit dem Begriff *Server* einen leistungsfähigen Rechner zu verbinden. Hier ist es eher umgekehrt: Der X-Server läuft in der Regel auf einem Arbeitsplatzrechner, während die X-Clients auf beliebigen Rechnern, bei aufwendigen Anwendungen also sinnvollerweise auf Hochleistungs-Rechnern, ablaufen. Der X-Server erbringt für die X-Clients einen Dienst: er stellt einen grafischen Arbeitsplatz zur Verfügung. Wie bereits erwähnt, kann ein X-Client auch auf demselben Rechner laufen wie der X-Server, wenn dieser Rechner multitasking-fähig ist.

Geschichte

X wurde ab 1984 am MIT (Massachusetts Institute of Technology) als Nachfolger des an der Stanford University entwickelten Fenstersystems W (für Window) entwickelt. Die erste öffentliche Version von X war die Version 10 (X10). Seit 1987 gibt es die Version 11 (X11). Das aktuelle Release von X11 ist Release 5 (X11R5).

1988 wurde die Pflege und Weiterentwicklung von X vom X-Konsortium übernommen, das ist ein Zusammenschluß von Firmen und Institutionen, in dem insbesondere alle großen Hardware-Hersteller vertreten sind. Das X-Konsortium legt auch Schnittstellendefinitionen und Richtlinien fest.

X ist urheberrechtlich geschützt, aber einschließlich der Quelltexte frei verfügbar.

Der X-Server

Der X-Server hat folgende Aufgaben:

- Er verwaltet genau einen Arbeitsplatz (*Display*) mit seinen Geräten, wie Tastatur, Zeigegerät (Maus), Bildschirm(e) (*Screens*).
- Er stellt komplexe Datenstrukturen zur Verfügung (*Ressourcen*¹):
 - Fenster (Windows) (s. u.)
 - Pixmaps (Farbmuster zum Füllen von Flächen)
 - Farbtabellen
 - Graphics-Contexts (Informationen für Zeichenoperationen)
 - Fonts

¹Der Begriff *Ressource* taucht im X-Window-System oft und mit unterschiedlichen Bedeutungen auf.

- Er bearbeitet Ausgabe- und Verwaltungsanforderungen der X-Clients.
- Er schickt Ereignisse (Events), die beispielsweise durch Aktionen des Benutzers ausgelöst werden, an die zuständigen X-Clients.

Ein **Fenster (Window)** ist ein rechteckiger Bildschirm-Bereich, in dem grafische Objekte oder Text ausgegeben werden können. Ein Fenster kann weitere Fenster, sog. *Kind-Fenster*, enthalten, so daß eine Hierarchie von Fenstern entsteht, in der jedes Fenster, bis auf das oberste in der Hierarchie, ein *Eltern-Fenster* hat. Das oberste Fenster ist der gesamte Bildschirm und heißt *Root-Fenster*. Kind-Fenster können nur innerhalb ihrer Eltern-Fenster dargestellt werden, werden also u. U. abgeschnitten, wenn sie aus ihrem Elternfenster herausreichen würden. Fenster dürfen sich überlappen.

Jeder Anwendung ist ein Kind-Fenster des Root-Fensters zugeordnet. Es ist das oberste Fenster dieser Anwendung (*Toplevel-Fenster*), unter dem sich die Fenster-Hierarchie dieser Anwendung aufspannt.

Eine wichtige Aufgabe des X-Servers ist die Zuordnung einer Aktion des Benutzers zu einem Fenster und damit zu einem X-Client, der dann mittels eines Ereignisses hiervon unterrichtet werden muß. Andere Ereignisse sind eher indirekte Folgen von Benutzer-Aktionen, wie z. B. das *Expose Event*. Es wird erzeugt, wenn ein Fenster bewegt oder gelöscht wird und dadurch ein zuvor überdeckter Bereich eines anderen Fensters wieder dargestellt werden muß, was möglicherweise, aber nicht in jedem Fall, vom X-Server erledigt werden kann. Ein X-Client sollte also stets auf das Auftreten eines *Expose-Events* vorbereitet sein.

X-Anwendungen

Es gibt eine Reihe von X-Anwendungen, die gewissermaßen zum X-Window-System gehören. Die wohl wichtigste ist der **Window-Manager**, dessen Aufgabe es ist, die Toplevel-Fenster der Anwendungen zu verwalten. Er ermöglicht es dem Benutzer, die Größe und Lage der Fenster sowie ihre Reihenfolge (bzgl. ihrer Überlagerung) festzulegen und Fenster zu Piktogrammen (Icons) zu verkleinern. Hierzu versieht der Window-Manager die Fenster mit zusätzlichen Dekorationsfenstern, über die die Manipulation der Fenster erfolgt.

Wichtig ist in diesem Zusammenhang, daß der Window-Manager ein X-Client ist. Das bedeutet insbesondere, daß er austauschbar ist. Es ist aber gerade

der Window-Manager, der das Aussehen und Verhalten des Systems (das „Look and Feel“) in hohem Maße beeinflußt. Es gibt eine Vielzahl unterschiedlicher Window-Manager, aus denen der Benutzer den ihm am meisten zusagenden aussuchen kann.

Weitere Standard-X-Anwendungen sind **xclock** (eine Uhr), **xcalc** (ein Taschenrechner) und **xterm** (ein Terminal-Emulator).

Entwicklung von X-Anwendungen

Von großer Bedeutung für die Entwicklung von X-Anwendungen sind die Konstrukte der folgenden drei aufeinander aufbauenden Ebenen:

- Xlib,
- X-Toolkit-Intrinsics und
- die Widget-Sets.

Basis der Entwicklung von X-Anwendungen ist die **Xlib**. Die Xlib ist eine Sprachanbindung des X-Protokolls an die Programmiersprache C in Form einer Funktionsbibliothek. Diese Basisbibliothek des X-Window-Systems stellt nur einfachste grafische Funktionen zur Verfügung.

Die Xlib bietet nahezu uneingeschränkte Möglichkeiten, jedoch ist die Programmierung auf dieser Basis ziemlich aufwendig. Erleichterung schaffen die **X-Toolkit-Intrinsics**, oder kurz **Xt-Intrinsics**. Sie stellen Mechanismen zum Erzeugen wiederverwendbarer grafischer Grundobjekte zu Verfügung. Die Xt-Intrinsics setzen auf der Xlib auf.

Diese grafischen Grundobjekte heißen **Widgets**. Es handelt sich dabei insbesondere um Interaktionsbausteine, wie Bildschirmknöpfe, Scrollbars, Menüs und Texteingabefelder.

Ein Widget ist ein *aktives Objekt*. Dabei ist *Objekt* im Sinne der objektorientierten Programmierung zu verstehen; Widgets werden beschrieben durch *Widget-Klassen*, die u. U. auf bereits existierende Widget-Klassen aufbauen (Vererbung). Ein Widget ist in dem Sinne *aktiv*, daß es in der Lage sein kann, Ereignisse zu verarbeiten. Kann ein Widget etwa auf das *Expose-Event* reagieren, so braucht sich der Anwendungs-Entwickler, der dieses Widget verwendet, nicht mehr darum zu kümmern.

Die verschiedenen Aspekte des Aussehens von Widgets werden durch Widget-Parameter, die *Ressource-Felder* oder auch *Ressourcen* genannt werden,

festgelegt. Hierfür gibt es Voreinstellungen sowie Ressource-Datenbanken, so daß die Anwendung bei der Verwendung eines Widgets nur die von diesen Standardwerten abweichenden Parameter spezifizieren muß.

Jedem Widget ist ein Fenster zugeordnet. Es gibt also analog zur Fenster-Hierarchie eine Widget-Hierarchie. Insbesondere werden die Toplevel-Fenster von sogenannten *Shell-Widgets* verwaltet.

Sammlungen von Widget-Beschreibungen werden **Widget-Sets** genannt. Es gibt eine Reihe von Widget-Sets, z. B. die Athena-Widgets (Xaw), die mit den X11-Quelltexten vertrieben werden, sowie die Widget-Sets von OSF/Motif und OPEN/LOOK (s. u.).

Xlib und Xt-Intrinsics beinhalten keine Einschränkungen bzgl. der aufzubauenden Bedienoberfläche; erst mit der Auswahl eines Widget-Sets werden Aspekte dieser Oberfläche festgelegt.

Grafische Oberflächen auf X

Es gibt einige grafische Benutzerschnittstellen auf der Basis von X. Genannt werden sollen hier die beiden Oberflächen *OSF/Motif* der Open Software Foundation und *OpenLook* von SUN und AT&T. Diese Oberflächen bieten ein Widget-Set sowie einen

Window-Manager, die aufeinander abgestimmt sind. Zum OSF/Motif gehören ferner die Beschreibungssprache *UIL* (User Interface Language), die die textuelle Beschreibung von Einzelheiten der Oberfläche außerhalb des Programms erlaubt, sowie der *Style Guide*, eine Sammlung von Stil-Richtlinien für Motif-Oberflächen.

X in unserer Universität

Die X-Software mit den Standard-X-Clients ist auf vielen Rechnern in der Universität vorhanden. Zunehmend gibt es Software-Pakete, die X für ihre grafischen Oberflächen benutzen, wie z. B. das Grafik-Paket PV~WAVE.

Um mit X arbeiten zu können, muß man einen X-Server auf seinem Arbeitsplatz installieren. Die dazu nötige Software gibt es von verschiedenen Herstellern. Einige dieser Produkte für PCs werden zur Zeit vom Rechenzentrum getestet.

BERLAGE, T.: *OSF/Motif und das X-Window System*, Addison-Wesley, 1991

BREDE, H.-J., JOSUTTIS, N., LEMBERG, S., LÖRKE, A.: *Programmieren mit OSF/Motif*, Addison-Wesley, 1991

JONES, O.: *Introduction to the X Window System*, Prentice-Hall, 1988

MUTH, M.: *Das X-Window-System*, Informatik-Spektrum 14 (1), S. 34-36, 1991

RUM-Lehre

Lehrveranstaltungen im 1. Halbjahr 1992

Beratung zum Lehrangebot durch Herrn W. Bosse jeweils Di, Do 11 bis 12 Uhr (Tel. 83-2461).

Zu den Ferienkursen

Vor Beginn des SS 1992 werden vom Universitätsrechenzentrum einige ganztägige Intensivkurse durchgeführt, in denen Stoffvermittlung und Übungen integriert sind. Diese Veranstaltungen sollen durch entsprechende Betreuung der Teilnehmer eigene Programmierübungen fördern. Das bedingt eine Begrenzung der Teilnehmerzahl der einzelnen Lehrveranstaltungen. Interessenten werden deshalb gebeten, sich möglichst bald, spätestens jedoch eine Woche vor Beginn der entsprechenden Veranstaltung, im Dispatch des Rechenzentrums (Raum 02) in die Anmelde listen einzutragen, und sollten unbedingt zu dem angekündigten Beginn anwesend sein.

Die Teilnehmer dieser Kurse werden gebeten, diese im SS 1992 zu belegen. Dies ist neben der bestandenen Abschlußklausur Voraussetzung für die Aushändigung eines Scheines über die erfolgreiche Teilnahme.

Zu den Semesterkursen

Eine Anmeldung ist nur für diejenigen Lehrveranstaltungen erforderlich, die nachfolgend besonders gekennzeichnet sind.

Lehrveranstaltungen in den Semesterferien (Februar bis April 1992)

320016	Aktuelle Systemumgebungen am Arbeitsplatz und auf zentralen Servern 17.2 bis 28.2.1992 vormittags Hörsaal: M2, Beginn: 17.2.1992, 10 Uhr	Kisker, H.-W./ Lange, W./ Mersch, R./ Mertz, K.-B./ Richter, G./ Speer, M.
320020	Datenverarbeitung in einer Unix-Arbeitsplatzumgebung 9. bis 20.3.1992 vormittags Hörsaal: M2, Beginn: 9.3.1992, 9 Uhr	Ost, St./ Speer, M./ Süselbeck
320035	Statistische Datenanalyse mit dem Programmsystem SPSS/PC+ vom 24.2. bis 6.3.1992 ganztägig Hörsaal: M4, Beginn: 24.2.1992, 9 Uhr	Nienhaus, R.
320040	Programmieren in FORTRAN vom 23.3. bis 3.4.1992 ganztägig Hörsaal: M2, Beginn: 23.3.1992, 9 Uhr	Reichel, K.
320054	Programmieren in Pascal 23.3. bis 3.4.1992 ganztägig Hörsaal: M5, Beginn: 23.3.1992, 9 Uhr	Stöckelmann, D.
320069	Programmieren in PL/I für Fortgeschrittene 17.2 bis 28.2.1992 ganztägig Hörsaal: M6, Beginn: 17.2.1992, 9 Uhr	Sturm, E.

320073	Computerunterstütztes Publizieren mit $\text{\LaTeX}$ für Fortgeschrittene 9.3 bis 20.3.1992 ganztägig Hörsaal: M4, Beginn: 9.3.1992, 9 Uhr	Kaspar, W.
--------	---	------------

Lehrveranstaltungen in der Vorlesungszeit

320088	Einführung in die ADV Di 15-17 Hörsaal: M4, Beginn: 28.4.1992	Pudlatz, H.
320092	Programmieren in FORTRAN Di 15-17 Hörsaal: M2, Beginn: 28.4.1992	Mertz, K.-B.
320107	Programmieren in Pascal Di 15-17 Hörsaal: M5, Beginn: 28.4.1992	Bosse, W.
320111	Programmieren in C Mi 9-11 Hörsaal: M4, Beginn: 29.4.1992	Mersch, R.
320126	FORTRAN und Vektorisierung Mi 15-17 Hörsaal: M4, Beginn: 22.4.1992	Neukäter, B.
320130	Objektorientierte Programmierung Mi 13-15 Hörsaal: M6, Beginn: 29.4.1992	Sturm, E.
320145	DV-Anwendungen für Geisteswissenschaftler ¹ Mi 13-15 Hörsaal: M5, Beginn: 29.4.1992	Kamp, H.
320150	Programmierung statistischer Methoden zu Fragebogenauswertungen Mo 13-17 Hörsaal: M4, Beginn: 27.4.1992	Zörkendörfer, S.
320164	Datenvisualisierung Mo 13-15 Hörsaal: M2, Beginn: 27.4.1992	Süselbeck, B.
320179	Computerunterstütztes Publizieren mit $\text{\LaTeX}$ Mi 13-15 Hörsaal: M2, Beginn: 22.4.1992	Perske, R.
320183	Textverarbeitung mit WordPerfect Mo 15-17 Hörsaal: CIP-Raum 405, Bisinghof 24-25 Beginn: 27.4.1992	Stöckelmann, D.
320198	Relationale Datenbanken Mi 11-13 Hörsaal: M2, Beginn: 22.4.1992	Hölters, J.
320202	Mikrorechner-Praktikum ¹ Mo 13-15 Hörsaal: CIP-Raum 106, Einsteinstr. 60 Beginn: 27.4.1992	Kisker, H.-W.

- 320217 Systemprogrammierung von Mikrorechnern
Do 15-17
Hörsaal: M4, Beginn: 23.4.1992
Lange, W.
- 320221 Einführung in das Betriebssystem Unix
Do 15-17
Hörsaal: M2, Beginn: 30.4.1992
Grote, M.
- 320236 Datenkommunikation
Do 13-15
Hörsaal: M2, Beginn: 30.4.1992
Richter, G./
Speer, M.
- 320240 Kolloquium über Themen der Informatik
Fr 15-17
Hörsaal: M2
Held, W./
die wiss. Mitarbeiter
des Rechenzentrums
- 320255 Anleitung zum DV-Einsatz bei wissenschaftlichen
nach Vereinbarung
die wiss. Mitarbeiter
des Rechenzentrums

¹ Wegen der Begrenzung der Teilnehmerzahl ist für diese Lehrveranstaltung eine frühzeitige Anmeldung im Dispatch des Rechenzentrums erforderlich.

Aktuelle Systemumgebungen am Arbeitsplatz und auf Servern (320016)

Die Veranstaltung gibt eine systematische Einführung in die Universitäts-Rechnerwelt, wie sie sich aus den aktuellen Beschaffungsmaßnahmen zur DV-Gesamtversorgung ergibt. Im Vordergrund stehen dabei die Dienste, die am Arbeitsplatz verfügbar oder von diesem aus nutzbar sind.

Da die Veranstaltung in unterschiedliche Themenblöcke unterteilt ist, ist es auch möglich, nur an bestimmten Terminen teilzunehmen. Die Themenblöcke sind:

- DOS-Arbeitsplätze
- Kommunikation
- Informationsdienste im Universitätsnetz
- Dienste im weltweiten Rechnernetz
- Zugang zu Diensten auf Unix-Systemen
- Zugang zu Diensten auf VMS-Systemen
- Zugang zu Diensten auf MVS-Systemen

Der genaue Zeitplan wird zu Beginn der Veranstaltung bekanntgegeben.

Datenverarbeitung in einer Unix-Arbeitsplatzumgebung (320020)

Der Einsatz von Unix wird sich durch die aktuelle Beschaffungsmaßnahme zur DV-Gesamtversorgung erheblich ausweiten. Für das ausgewählte System soll

eine systematische Einführung unter Einbeziehung der Kommunikationsmöglichkeiten und verschiedener Anwendungen gegeben werden.

Der genaue Themenplan wird zu Beginn der Veranstaltung bekanntgegeben.

Einführung in die ADV (320088)

Es werden Grundkenntnisse der automatisierten Datenverarbeitung (ADV) vermittelt, die für das tiefere Verständnis der Vorgänge in und um den Computer hilfreich sind. Die Lehrveranstaltung ist nicht nur als Vorbereitung für weiterführende Veranstaltungen des Rechenzentrums gedacht, sondern sie soll auch neben der Wissensvermittlung der Themen, für die in Programmierkursen in der Regel die Zeit fehlt, einen allgemeinen Überblick über Einsatzmöglichkeiten von Anwendersoftware bieten.

DVORATSCHEK: *Grundlagen der Datenverarbeitung*, de Gruyter
SOMMER: *Informatik – eine PC-orientierte Einführung*

Statistische Datenanalyse mit dem Programmsystem SPSS/PC+ (320035)

Das statistische Programmsystem SPSS (Statistical Package for the Social Sciences) wird in der am Rechenzentrum implementierten Version SPSS/PC+

vorgestellt. Mit diesem System stehen bequem aufzurufende Programme zu den gebräuchlichen univariaten und multivariaten statistischen Verfahren sowie zur Datenmanipulation zur Verfügung. SPSS wird z. B. zur statistischen Auswertung von Fragebögen eingesetzt.

In dieser Veranstaltung wird das programmtechnische Rüstzeug zur Durchführung derartiger Auswertungen vermittelt. Solide Grundkenntnisse bezüglich der anzusprechenden statistischen Verfahren sowie Kenntnisse der Anwendungsmöglichkeiten dieser Verfahren im jeweiligen Fachgebiet sind erwünscht und bei den praktischen Übungen von großem Nutzen.

STEINHAUSEN/ZÖRKENDÖRFER: *Statistische Datenanalyse mit dem Programmsystem SPSS^X und SPSS/PC+*, Oldenbourg (Hörerscheine bei den Autoren erhältlich)

Programmieren in FORTRAN (320040, 320092)

FORTRAN ist eine Programmiersprache, die vorwiegend für die Formulierung von Problemlösungen aus dem Bereich der Naturwissenschaften (Numerik, Statistik) geeignet ist. Unverwundlichkeit, leichte Erlernbarkeit und Anwendbarkeit auf Computern fast aller Hersteller haben dieser Programmiersprache eine weite Verbreitung gesichert.

BRAUER: *Programmieren in FORTRAN 77*, Hüthig

BRAUER: *FORTRAN 77 – Ständig im Griff*, Hüthig

KIESSLING/LOWES: *Programmierung mit FORTRAN 77*, Teubner Studienskripten

METCALF: *Effective FORTRAN 77*, Oxford University Press

WEHNES: *FORTRAN 77*, Hanser

VS FORTRAN Release 4.0 (erhältlich im Rechenzentrum)

FORTRAN und Vektorisierung (320126)

Diese Veranstaltung wendet sich in erster Linie an diejenigen, die ihre FORTRAN-Programmierkenntnisse auffrischen oder vertiefen möchten. Im ersten Teil der Vorlesung sollen die FORTRAN-Sprachelemente kurz und systematisch zusammengefaßt werden. Im zweiten Teil soll speziell auf die Möglichkeiten der Effizienzverbesserung durch Vektorisierung und Parallelisierung eingegangen werden.

Programmieren in Pascal (320054, 320107)

Die Programmiersprache Pascal ist aufgrund ihres didaktischen Konzepts für Programmieranfänger besonders zu empfehlen. Andererseits ist Pascal durch die Vielzahl von Datentypen und Strukturierungsmöglichkeiten für Anwendungen numerischer wie nichtnumerischer Art gleichermaßen geeignet.

JENSEN/WIRTH: *Pascal, User Manual and Report*, Springer

MARTY: *Methodik der Programmierung in Pascal*, Hanser

OTTMANN/WIDMEYER: *Programmieren mit Pascal*, Teubner

WILSON/ADDYMAN: *Leichtverständliche Einführung in das Programmieren mit Pascal*, Hanser

Alle Lehrbücher zu Turbo Pascal ab Version 4.0

Programmieren in C (320111)

C ist eine Programmiersprache, deren Einsatzmöglichkeiten einerseits durch Assembler-ähnliche Sprachelemente und andererseits durch Elemente moderner blockstrukturierter Sprachen sehr vielseitig sind. Durch einen hohen Grad an Portabilität ist C eine Sprache mit stark zunehmender Verbreitung; Compiler-Implementierungen stehen praktisch für alle Betriebssysteme und Rechnerarten zur Verfügung.

Die Lehrveranstaltung ist eine Einführung in die Programmiersprache C. Zum Verständnis sind aber Kenntnisse einer anderen Programmiersprache von Nutzen.

HARBISON/STEELE: *C Reference Manual*, Prentice Hall

KERNIGHAN/RITCHIE: *Programmieren in C*, Hanser

Programmieren in PL/I für Fortgeschrittene (320069)

Die Vorlesung wendet sich an Hörer, die bereits an einem PL/I-Grundkurs teilgenommen haben. Behandelt werden Themen wie Ein- und Ausgabe auf Platten und Bändern, Datenorganisation, Unterprogrammtechniken, Listenverarbeitung, Multitasking und die Benutzung des PL/I-Preprocessors. Der Hörer soll einen Einblick in die Möglichkeiten einer universellen Programmiersprache erhalten und in die Lage versetzt werden, effiziente und gut strukturierte Programme zu schreiben.

STURM: *Programmieren in PL/I*, Vieweg

Objektorientierte Programmierung (320130)

Nach der sogenannten Strukturierten Programmierung setzt sich eine neue Methode zur Konstruktion von Software durch: Objektorientierte Programmierung. Wesentlicher Begriff ist allerdings nicht das Objekt, sondern die Klasse, eine Verallgemeinerung des Begriffs Datentyp. Daten und die auf ihnen operierenden Programme werden zu einer Klasse zusammengefaßt. Ein Objekt gehört immer zu einer Klasse und kann direkt nur auf Daten dieser Klasse zugreifen und mit Programmen dieser Klasse bearbeitet werden. Interessant ist weiterhin, daß Daten und Methoden einer Klasse von anderen Klassen „geerbt“ werden können.

Die Terminologie der Objektorientierten Programmierung ist allerdings noch im Fluß. Urväter sind vor allem die Programmiersprachen SIMULA und SMALLTALK. Neueren Datums sind die C-Erweiterung C++ sowie die allein zur Unterstützung der Objektorientierten Programmierung geschaffene Programmiersprache Eiffel. Auf letztere wird sich vor allem diese Lehrveranstaltung stützen, da sie nicht nur allgemeiner als andere ist, sondern auch übersichtlicher. Für Übungen steht ein Eiffel-Compiler unter MS-DOS zur Verfügung.

B. MEYER: *Object Oriented Software Construction*, Prentice Hall

B. MEYER: *Eiffel: The Language*, Prentice Hall

Computerunterstütztes Publizieren mit $\text{\LaTeX}$ (320179)

$\text{\LaTeX}$ ist eine Erweiterung des Satzsystems $\text{\TeX}$, die das Arbeiten mit $\text{\TeX}$ wesentlich erleichtert. Dem Autor werden z. B. fertige Layouts für Bücher, Reports und Artikel zur Verfügung gestellt, die er selbst in gewissen Grenzen seinen eigenen Vorstellungen leicht anpassen kann. $\text{\LaTeX}$ steht auf IBM-kompatiblen PCs und auf vielen anderen Rechnersystemen zur Verfügung.

In dieser Veranstaltung werden die Einsatzmöglichkeiten von $\text{\LaTeX}$ im Publikationsprozeß vorgestellt. Es wird gezeigt, wie Texte mit Editoren wie KEDIT unter DOS für $\text{\LaTeX}$ erfaßt, mit $\text{\TeX}$ formatiert, zur Kontrolle am Bildschirm angezeigt und auf unterschiedlichen Druckern ausgegeben werden können.

Die Hörer sollten Grundkenntnisse im Umgang mit PCs besitzen.

RUM-Lehre

KOPKA: *$\text{\LaTeX}$ – Eine Einführung*, Addison Wesley

LAMPOR: *$\text{\LaTeX}$ – A Document Preparation System*, Addison Wesley

PARTL/SCHLEGEL/HYNA: *$\text{\LaTeX}$ Kurzbeschreibung*

WONNEBERGER: *Kompaktführer $\text{\LaTeX}$* , Addison Wesley

KNUTH: *The $\text{\TeX}$ -book*, Addison Wesley

Computerunterstütztes Publizieren mit $\text{\LaTeX}$ für Fortgeschrittene (320073)

Diese Veranstaltung richtet sich an Autoren, die schon erste Erfahrungen mit $\text{\LaTeX}$ gemacht haben und die Erweiterungen und Veränderungen durchführen wollen.

Zusammen mit dem Makropaket $\text{\LaTeX}$ unterstützt das Satzsystem $\text{\TeX}$ die Trennung der Logik- und Layout-Struktur eines Dokumentes, d.h. der Autor kann sich ganz auf den Inhalt und die logische Struktur seines Textes, der aus Textelementen wie Überschriften, Absätzen und Anmerkungen besteht, konzentrieren und muß sich nicht gleichzeitig, sondern in einem separaten Arbeitsschritt um das Aussehen dieser Textelemente im gedruckten Dokument kümmern.

In $\text{\LaTeX}$ sind eine Reihe von Textelementen definiert, denen mehrere Layout-Strukturen zugeordnet werden können. Ein Dokument, dessen Textelemente mit $\text{\LaTeX}$ -Markierungen versehen sind, kann so in unterschiedlicher Gestalt formatiert und gedruckt werden.

Die in $\text{\LaTeX}$ standardmäßig vorgegebenen Strukturen decken allerdings nicht den Bedarf aller Publikationsformen an unserer Universität ab, so daß es für viele Autoren unumgänglich ist, neue Textelemente einzuführen und die vorgegebene Layout-Struktur zu verändern.

ABRAHAMS: *$\text{\TeX}$ for the Impatient*, Addison Wesley

APPELT: *$\text{\TeX}$ für Fortgeschrittene*, Addison Wesley

KOPKA: *$\text{\LaTeX}$ - Eine Einführung*, Addison Wesley

KOPKA: *$\text{\LaTeX}$ Erweiterungsmöglichkeiten*, Addison Wesley

LAMPOR: *$\text{\LaTeX}$ - A Document Preparation System*, Addison Wesley

SCHWARZ: *Einführung in $\text{\TeX}$* , Addison Wesley

DV-Anwendungen für Geisteswissenschaftler (320145)

Die Veranstaltung möchte zunächst eine theoretische Einführung in Grundeinheiten und Arbeitsweise von Mikrorechnern geben. Danach folgen praktische

Übungen an den Rechnern zu den Themen Textverarbeitung, Einsatz von Makros, Sortierung und Recherche gespeicherter Daten.

Textverarbeitung mit WordPerfect (320183)

Die Veranstaltung gliedert sich in folgende Teile:

- Grundfunktionen von WordPerfect 5.1
- Einfache Textbearbeitung
- Drucken, Löschen, Transfer von Texten
- Erweiterte Textgestaltung

Im letzten Teil sollen behandelt werden: Fuß- und Endnotengestaltung, Kopf und Fußzeilenorganisation, Tabulatoren, Tabellenbearbeitung, parallele Bearbeitung von Dokumenten, Fenstertechnik, Systembefehle zur Dateiverwaltung, Sicherungsmaßnahmen, Makrodefinitionen, Serienbriefbearbeitung.

KAMP: *Textverarbeitung mit WordPerfect, Version 5.1, Software-Information 22* (im Rechenzentrum erhältlich)

Programmierung statistischer Methoden zu Fragebogenauswertungen (320150)

Zum Zwecke einer Fragebogenauswertung unter Einsatz der EDV werden Methoden der computergeordneten Datenaufbereitung und statistischer Auswertungsverfahren demonstriert und von den Teilnehmern erprobt.

STEINHAUSEN/ZÖRKENDÖRFER: *Informationbearbeitung und Datenanalyse mit dem Programmsystem SAS*, Oldenbourg 1992 (Hörerscheine bei den Autoren erhältlich)

Datenvisualisierung (320164)

Datenvisualisierung bedeutet die Kombination grafischer und analytischer Werkzeuge zur Darstellung komplexer Zusammenhänge. In dieser Vorlesung sollen Produkte vorgestellt werden, die dieses Konzept in einer Unix-Workstation-Umgebung realisieren.

Handbücher der Hersteller

Relationale Datenbanken (320198)

Datenbanken gewinnen immer mehr an Bedeutung zur automatischen Gewinnung von Informationen. Insbesondere der Typ Relationale Datenbank mit der standardisierten Abfragesprache SQL stellt vielfältige Anwendungsmöglichkeiten zur Verfügung.

C.J. DATE: *A Guide to the SQL Standard*

C.J. DATE: *An Introduction to Database Systems*

Mikrorechner-Praktikum (320202)

Das Praktikum führt in den Umgang mit den technischen Komponenten eines Mikrorechners ein. Es ist eingeteilt in 10 Themenbereiche:

- Festplatte
- Diskette
- Tastatur
- RAM- und ROM-Speicher
- V.24-Kommunikation
- Drucker
- Video
- Leistungsmessungen
- Konfiguration von Programmen
- Systemplatine

Jeweils zwei bis drei Personen beschäftigen sich in einer Stunde mit einem dieser Themenbereiche. Für die Teilnahme ist eine Voranmeldung bis zum 1.4.1991 erforderlich.

Systemprogrammierung von Mikrorechnern (320217)

Die Vorlesung führt in die Architektur der Hardware und des Betriebssystems eines Mikrorechners ein. Im Mittelpunkt steht dabei die Nutzung der von diesen Komponenten angebotenen Leistungen durch einen Programmierer. Insbesondere werden folgende Themen behandelt:

- die Hardware eines Mikrorechners
- die Schnittstellen des BIOS
- die Schnittstellen des Betriebssystems MS-DOS
- die Nutzung von Hilfsmitteln zur Software-Entwicklung

Grundkenntnisse der Programmierung in C (oder Pascal) sind erforderlich.

RUM-Lehre

NORTON: *Programmierhandbuch für den IBM-PC*, Microsoft Press Vieweg

Microsoft MS-DOS 3.1 *Programmierhandbuch*, Markt & Technik

DUNCAN: *MS-DOS für Fortgeschrittene*, Microsoft Press Vieweg

Einführung in das Betriebssystem Unix (320221)

Unix ist ein weitverbreitetes Mehrbenutzerbetriebssystem. Es ist auf Rechnern verschiedener Hersteller und unterschiedlicher Leistungsklassen ablauffähig. Damit steht dem Unix-Anwender vom Mikrorechner bis zum Großrechner die gleiche leistungsfähige und komfortable Programmier- und Arbeitsumgebung zur Verfügung. Hardware-Unterschiede der einzelnen Maschinen werden weitgehend verdeckt.

Vorkenntnisse der Programmiersprache C sind erwünscht, aber nicht unbedingt erforderlich.

BANAHAN/RUTTER: *Unix*, Hanser

CHRISTIAN: *The Unix Operating System*, Wiley-Interscience

GULBINS: *Unix*, Springer

Datenkommunikation (320236)

In dieser Veranstaltung werden grundlegende Begriffe und Verfahren der Datenkommunikation erläutert.

Hierbei wird auch ausführlich auf ein international genormtes Architekturmodell zur Strukturierung von Kommunikationsvorgängen eingegangen.

Neben den grundlegenden Aspekten der Datenkommunikation ist ein weiterer Schwerpunkt der Veranstaltung die Beschreibung ausgewählter Methoden bei der Datenfernkommunikation und in lokalen Rechnernetzen.

Abschließend sollen die an der Universität eingesetzten Verfahren erläutert und die vom Arbeitsplatz aus nutzbaren Dienstleistungen des Universitätsnetzes beschrieben werden.

ECKHARDT/NOWAK: *Standard-Architekturen für Rechnerkommunikation*, Oldenbourg

FRANCK: *Rechnernetze und Datenkommunikation*, Springer

KAFKA: *Grundlagen der Datenkommunikation*, Datacom Buchverlag

Kolloquium über Themen der Informatik (320240)

Im Rahmen des Kolloquiums werden Vorträge über neuere Entwicklungen der Informatik gehalten. Vortragstermine werden durch Aushang im Universitätsrechenzentrum bekanntgegeben.

RUM-Aroma

Vorläufige Benutzungsordnung

0 Allgemeines

01 Öffnungszeiten

Ein- und Ausgaberräume	Mo-Fr	8 - 17 Uhr
Beratungsraum	Mo-Fr	10 - 16 Uhr
Locherraum	Mo-Fr	8 - 17 Uhr
Zusatzmaschinen	Mo-Fr	9 - 16 Uhr
Testzeiten:	IBM 360/50	Mo-Fr 10 - 12 Uhr
		14 - 16 Uhr
	Z23	Mo-Fr 9 - 16 Uhr

Außerhalb der angegebenen Zeiten ist die Arbeit im Rechenzentrum nur nach Vereinbarung möglich.

02 Benutzerkreis und Anmeldung

Das Rechenzentrum steht den Angehörigen aller Institute der Westfälischen Wilhelms-Universität Münster sowie den Teilnehmern von Programmierkursen und Übungen über Praktische Mathematik kostenlos zur Verfügung. Für die Benutzung der Datenverarbeitungsanlagen ist grundsätzlich die Teilnahme an einem Programmierkurs Voraussetzung!

Anmeldeformulare für die Teilnahme am Rechenbetrieb sind bei der Beratung erhältlich: Bei Abgabe des ausgefüllten und vom Institutsdirektor bzw. Dozenten unterschriebenen Formulare erhält der Benutzer eine Job-Nummer (siehe unter 12) und Rechenzeit. Die Rechenzeit beträgt für Testzwecke 1/2 Stunde; weitere Zeitreservierungen erfolgen nach Rücksprache beim Beratungsdienst; Job-Nummern von Benutzern, die den Rechenbetrieb stören, können gesperrt werden.

03 Programmierkurse, Beratungsdienst und Kolloquium

Vom Rechenzentrum werden regelmäßig für Hörer aller Fakultäten Programmierkurse abgehalten (siehe Vorlesungsverzeichnis). Diese erfordern i. a. keinerlei Vorkenntnisse.

Die Programmierung selbst wird vom Rechenzentrum in keinem Fall übernommen: Zur Beantwortung von Fragen, die bei der Programmierung auftreten, steht der Beratungsdienst zur Verfügung.

Im Rechenzentrum findet ein Kolloquium statt. Jeder Benutzer des Rechenzentrums verpflichtet sich, auf eine Einladung hin in diesem Kolloquium über seine Arbeiten zu referieren. Termine hierfür werden durch Aushang bekanntgegeben.

1 IBM 360/50

11 Der Rechenbetrieb

Jeder Benutzer kann im Rahmen der ihm zur Verfügung gestellten Rechenzeit beliebig viele Lochkartensätze (Jobs, siehe unter 12) zum Auflisten, Duplizieren, Testen oder Rechnen bei der Dispatcherin abgeben. Als Anweisung zur Bearbeitung gilt nur die Lochung der Karten.

Nach der Bearbeitung des Jobs auf der Rechenanlage werden Karten und Listen von der Dispatcherin in den Rückgabekasten des Benutzers gelegt und können dort abgeholt werden; Rückfragen bei der Dispatcherin bewirken keine Beschleunigung der Bearbeitung.

Fragen bei Unklarheit über Ergebnisse können unter Vorlage aller Unterlagen an den Beratungsdienst gerichtet werden.

Das Betreten des Maschinenraums ist verboten, Rückfragen bei den Operateuren sind nicht gestattet.

Jobs, die nicht nach den Standardschemata zusammengesetzt sind, sollten mit dem Beratungsdienst besprochen werden.

Vorrang der Arbeiten während der Testzeiten:

1. LlST-Läufe für das Auflisten von Kartensätzen und DUPL-Läufe für das Duplizieren von Kartensätzen
2. TEST-Läufe (1000 Zeilen oder Karten, 3 Minuten)
3. PROD-Läufe (5000 Zeilen oder Karten, 15 Minuten)
4. Langläufe mit Zeitangabe auf der Jobkarte

12 Der Job

Jeder Arbeitsauftrag für die Rechenanlage ist in Gestalt eines Jobs bei der Dispatcherin abzugeben. Ein Job besteht aus einem vom Benutzer zusammengestellten Kartensatz und ggf. aus Lochstreifen. Jobs sollen nach folgenden Konventionen zusammengestellt sein:

121 Programm- und Datenkarten

Der Benutzer locht nur die Quellenprogramme und die Datensätze auf Lochkarten ab, (rote) Job-Kontroll-Karten dürfen nicht selbst erstellt werden, sie liegen aus und müssen vom Benutzer in seinen Kartensatz einsortiert werden: Programme und Datensätze sollen in der Regel auf weißen Karten abgelocht sein, rote Karten sind ausschließlich den Job-Kontroll-Karten vorbehalten und sind sonst nicht zu verwenden.

122 JOB-Karte

Die erste Karte jedes Jobs ist die (rote) JOB-Karte. In die bereitliegenden Job-Karten muß der Benutzer folgende Angaben gelocht haben:

1. ab Spalte 3 einen Jobnamen, der aus höchstens 8 alphanumerischen Zeichen gebildet wird, von denen das erste ein Buchstabe sein muß; jeder Job kann mit einem eigenen Namen versehen werden;
2. in die Spalten 17 bis 21 die dem Benutzer zugeteilte Job-Nummer (3 Buchstaben und 2 Ziffern);
3. in die Spalten 23 bis 26 die Art der Arbeit (LIST, DUPL, TEST, PROD) oder die beanspruchte Zeit in Minuten (4-stellig);

4. ab Spalte 29 der Name des Benutzers (wie bei der Erteilung der Jobnummer angegeben).

Darüber hinaus können in die Spalten 47 bis 71 Bemerkungen für die Dispatcherin und den Operateur gelocht werden:

Telefon-Nr., falls nach Durchführung eines PROD- oder Langlaufes eine telefonische Benachrichtigung erfolgen soll;

LOCHSTREIFEN, falls Lochstreifen bearbeitet werden sollen;

TPxxxx, falls ein dem Benutzer zur Verfügung gestelltes Magnetband (Nr. des Bandes TPxxxx) für die Bearbeitung des Jobs benötigt wird.

Beispiel für eine JOB-Karte:

//PROGRAMM JOB (NNN77,TEST),MEIER

TEL5566 TP0044

123 Steuerkarten

Neben der JOB-Karte werden für die Bearbeitung eines Jobs weitere Job-Kontroll-Karten benötigt. Für die Standard-Aufgaben liegen die Steuerkarten aus und müssen nach den aushängenden Schemata in den Kartensatz des Benutzers von diesem eingefügt werden.

Für Sonderaufgaben werden die zugehörigen Steuerkarten vom Beratungsdienst erstellt.

124 Endkarte

Die letzte Karte jedes Jobs muß eine gelbe Endkarte sein:

//

ENDKARTE

Auch Endkarten liegen aus.

125 Lochstreifen

Zu einem Job können 5-, 6-, 7- und 8-Kanal-Lochstreifen gehören. Soll ein Lochstreifen bearbeitet werden, muß er am Anfang und am Ende mindestens 400 Leerzeichen (Transportlochungen) enthalten. Auf den Anfang und auf das Ende müssen der Name und die Job-Nummer des Benutzers sowie der Jobname des zugehörigen Programms mit der Hand geschrieben sein.

13 Das Operating System /360

Im OS/360 stehen Übersetzer für FORTRAN IV, PL/I, COBOL und RPG sowie andere Dienstprogramme zur Verfügung. Der Aufruf von Programmen des OS/360 und die Einheiten-Zuordnung erfolgt durch Job-Kontroll-Anweisungen. Für Standard-Aufgaben werden die benötigten Job-Kontroll-Anweisungen zu katalogisierten Prozeduren zusammengefaßt, die der Benutzer durch eine einzige Steuerkarte aufrufen kann.

2 ZUSE Z 23

Für spezielle Rechenarbeiten steht die Rechenanlage Zuse Z 23 zur Verfügung. Arbeiten auf der Z 23 werden im „offenen Betrieb“ durchgeführt. Zur Benutzung der Z 23 ist die Genehmigung des Leiters des Rechenzentrums einzuholen.

3 Zusatzgeräte und Arbeitsräume

Für Nebenarbeiten stehen den Benutzern im Locherraum mehrere Kartenlocher, Fernschreiber und eine Sortiermaschine zur Verfügung. Vor der ersten Benutzung mit diesen Geräten soll sich jeder Benutzer in die Handhabung einweisen lassen.

Für Arbeiten am Schreiblocher, die länger als 30 Minuten dauern, ist eine Platzreservierung bei der Dispatcherin erforderlich. Für kurze Korrekturarbeiten ist ein Expresflocher ausgezeichnet, der anderweitig nicht in Anspruch genommen werden darf.

Maschinenfehler sind bei der Dispatcherin zu melden, eigenmächtige Reparaturversuche sind untersagt. Für die Beschädigung von Programmen bzw. den Verlust von Rechenzeit durch Maschinenfehler oder andere Störungen haftet das Rechenzentrum nicht.

Es wird gebeten, im Locherraum und in den Arbeitsräumen auf Ordnung zu achten und die Tischplatten schonend zu behandeln. Lochkarten, Papier und sonstiger Abfall sind getrennt (!) in die jeweils dafür vorgesehenen Kästen und Körbe zu werfen. Den Benutzern können in beschränktem Maße Lochkartenkästen in den aufgestellten Lochkartenschränken von der Dispatcherin zur Verfügung gestellt werden. Herumliegende Lochkarten werden vernichtet.

Im Keller und im Erdgeschoß ist das Rauchen verboten!

Wenn Sie die voranstehenden Vorschriften bis hierher interessiert verfolgt haben, werden Sie sich wohl verwundert am Kopf gekratzt haben: das soll die neue Benutzungsordnung des Rechenzentrums sein?! Zwar ist sie als vorläufig gekennzeichnet – steht doch eine endgültige Fassung wegen der fehlenden Zustimmung durch die zuständigen Universitätsgremien noch aus, wir dürfen unsere Benutzer aber versichern, daß die hier abgedruckte Ordnung niemals genehmigt werden wird, da sie bereits total veraltet ist: sie stammt nämlich vom 2. Mai 1967. Dieses Datum liegt nunmehr fast ein Vierteljahrhundert zurück, so daß wir mit dieser Ordnung ein Stück Rechenzentrums-geschichte präsentieren können. Die älteren von uns mögen sie mit leicht wehmütigen Erinnerungen, die jüngeren wahrscheinlich mit ironischem Grinsen gelesen haben.

Zwischen dem jetzt beschafften Zentralrechner „ES/9000“ und der damaligen „/360-50“ liegen einerseits fünf Zehnerpotenzen an Rechen-Power, aber auf der anderen Seite nur noch eine weitere größere Installation: die „3032“ aus dem Jahre 1978, die für uns den Übergang zur /370-Architektur kennzeichnete. In der Universität Münster haben wir also jeweils 12 bis 13 Jahre gebraucht, um einen neuen Rechner vom Geldgeber-Duo Bund-Land zu erhalten.

In unserer historischen Rechenzentrumsordnung von 1967 wird sicherlich bei denjenigen Lesern, die erstmals durch einen Spielcomputer mit der Datenverarbeitung bekannt geworden sein mögen, die häufige Erwähnung der Lochkarte Befremden hervorgerufen haben, jenes dollarscheingroßen Stücks Pappe, in das

auf mechanische (!) Weise Löcher hineingestanzte wurden, um sie anfänglich ebenfalls mechanisch, später optisch abzufragen. Nun, die letzten Lochkarten haben wir schon vor etwa 10 Jahren abgeschafft, was uns durch die obige Vorschrift „Herumliegende Lochkarten werden vernichtet“ leicht gemacht wurde.

Ihnen werden nicht nur die aus heutiger Sicht benutzerunfreundlichen Öffnungszeiten aufgefallen sein, sondern vielleicht der reichlich mittelbare Umgang der Benutzer mit dem Rechenzentrum, der damals über eine sehr wichtige Person abgewickelt wurde: die Dispatcherin. Sie wird gleich acht mal in diesem kurzen Reglement (das man auch als „Lex Dose“ bezeichnen könnte) erwähnt und soll sogar für die Entgegennahme von Maschinenfehlern verantwortlich gewesen sein! Die erste Dispatcherin unseres Rechenzentrums war eine ebenso attraktive wie resolute Blondine, die nach der Heirat mit dem damaligen Leiter der Systemgruppe zusammen mit diesem das Rechenzentrum verließ. Ihr folgte für kurze Zeit eine weitere Dispatcherin, deren Aufgaben später das jetzige allseits bekannte „hochqualifizierte, intelligente Team“ übernahm.

Heute nach 25 Jahren sind die Wege ins Rechenzentrum dank der universitätsweiten Vernetzung selten geworden, wahrscheinlich werden sie überwiegend dann unternommen, wenn man zum Abholen von Disketten, Grafik-Ausgabe oder Übungsscheinen ins Dispatch muß, der Einrichtung, die noch immer an die damalige Dispatcherin erinnert.

(H. Pudlatz)

inforu-Quiz

von

E. Sturm

Hinweise und Beispielprogramm zum Auto- und Ziegenproblem.

Zunächst möchte ich mich für die Zuschriften auf das letzte inforu-Quiz bedanken. Ich habe Programme in C, Pascal, PL/I, REXX und SAS zugeschickt bekommen. Manche der Autoren brachten ihr anfängliches Erstaunen über das Ergebnis ihres Simulationsprogramms zum Ausdruck. Es scheint tatsächlich so zu sein, daß man zuerst der Fernsehgesellschaft Recht gibt und meint, daß es egal sei, ob man beim ersten Tip bleibt oder seine Entscheidung noch einmal ändert.

Nichtsdestotrotz kommt bei einer Simulation heraus, daß es doppelt so wahrscheinlich ist, einen Wagen zu gewinnen, wenn man die Strategie verfolgt, immer die dritte noch verbliebene Tür zu wählen. Wie kann man sich das klarmachen?

1. Wer immer bei seiner ersten Wahl bleibt, für den ist und bleibt die Wahrscheinlichkeit ein Drittel, da er den Hinweis des Showmasters nicht beachtet.
2. Wer, nachdem der Showmaster eine Ziegentür geöffnet hat, noch einmal eine Münze wirft, für den gilt tatsächlich Gleichwahrscheinlichkeit.
3. Wer sich aber immer für die dritte Tür entscheidet, für den gilt: Hat er beim ersten Tip eine Ziege getroffen, so verhilft ihm der Showmaster, da er eine weitere Ziege verrät, zu einem Auto! Und da es zwei Ziegen und nur ein Auto gibt, erreicht man auf diese Weise doppelt so häufig ein Auto wie eine Ziege.

Nun ein Beispielprogramm von inforu-Leser B. Schlagheck, aus didaktischen Gründen in der Programmiersprache Pascal:

```
program Let_s_make_a_deal;
```

```
type Tuer = (Tuer1, Tuer2, Tuer3);
const Ewigkeit = 10000;
      GewinnStrategie1 : longint = 0;
      GewinnStrategie2 : longint = 0;
      GewinnStrategie3 : longint = 0;
```

```
var Versteck      : Tuer;
    Wahl          : Tuer;
    Zeigen        : Tuer;
    TV_Show       : longint;
```

```
function Waehle_nicht(T1, T2: Tuer): Tuer;
(*- Wählt zufällige Tür <> T1, T2 -*)
var Zufall : Tuer;
begin
  repeat
    Zufall := Tuer(random(3));
  until (Zufall <> T1) and (Zufall <> T2);
  Waehle_nicht := Zufall;
end;
```

```
begin (* Hauptprogramm *)
  randomize;
  for TV_Show := 1 to Ewigkeit do begin
    Versteck := Tuer(random(3));
    Wahl := Tuer(random(3));
    Zeigen := Waehle_nicht(Versteck, Wahl);
    (*- Strategie 1: Tür beibehalten -*)
    if Wahl = Versteck then
      inc(GewinnStrategie1);
    (*- Strategie 2: andere Tür -*)
    Wahl := Waehle_nicht(Wahl, Zeigen);
    if Wahl = Versteck then
      inc(GewinnStrategie2);
    (*- Strategie 3: beliebige Tür -*)
    (*- mit gleicher Wahrscheinlichkeit -*)
    Wahl := Waehle_nicht(Zeigen, Zeigen);
    if Wahl = Versteck then
      inc(GewinnStrategie3);
  end;

  (*- Die auf Anhieb nicht zu erwartenden Wahrscheinlichkeiten: -*)
  writeln('Häufigkeit für Tür beibehalten: ',
    GewinnStrategie1/Ewigkeit*100 :6:2, '%');
  writeln('Häufigkeit für andere Tür: ',
    GewinnStrategie2/Ewigkeit*100 :6:2, '%');
  writeln('Häufigkeit für beliebige Tür: ',
    GewinnStrategie3/Ewigkeit*100 :6:2, '%')
end.
```

Wer keine Lust hat, dieses Programm abzutippen oder einzuscannen, um das Ergebnis der Simulation zu erhalten, für den sei hier die Ausgabe eines Probelaufs angegeben:

```
Häufigkeit für Tür beibehalten: 34.44%
Häufigkeit für andere Tür:      65.56%
Häufigkeit für beliebige Tür:   49.85%
```