



# PowerKey 4+5/93

## Inhalt

### 4 Editorial

## Aktuell

5 NeXT und HP, Sorbus

6 NEXTSTEP 3.2

## NiCE Internals

9 Veranstaltungskalender

9 NiCE Markt

## Software

10 HSDSpell

12 Geheimnisse von NEXTSTEP 3.0

## Mail

14 E-mail Verbindung mit der nice (Nachtrag)

## TeX

16 TeX-Kurs, Teil 2

## NextAnswers

20 NeXT Laser Printer, Komprimierte Dateien

## Octets

21 comp.sys.next.bugs

22 Impressum

## Liebe Leserin, lieber Leser

### *In eigener Sache*

Eigentlich wollte ich dieses Jahr mindestens sechs *PowerKey*-Ausgaben herausgeben, aber das ist gar nicht so einfach. Besonders dann nicht, wenn jedes Heft auch noch einen Umfang von 28+ Seiten haben soll, es aber an den dringend benötigten Artikeln fehlt. – Um dieses Ziel dennoch noch zu erreichen, gibt's diesmal eine Doppelnummer!

Vielleicht ist es sinnvoller, *PowerKey* ab 1994 nur noch vierteljährlich erscheinen zu lassen (oder den Umfang entsprechend zu reduzieren, aber dafür ist der tolle Umschlag fast zu schade). Ob es wirklich dazu kommt, entscheiden allein die Leser (also auch Du!) durch ihre Mitarbeit – sowie der neue Redaktor. Denn ich habe vor, mein Amt nächstes Jahr einem Nachfolger (oder einer Nachfolgerin) zu übergeben. (Das gleiche gilt übrigens auch für den Verleger.) Also: InteressentInnen bitte melden!

Die amerikanische Zeitschrift *NeXTWORLD* erscheint übrigens seit August neu nun jeden Monat, nachdem sie bereits anfangs Jahr von vier auf sechs Ausgaben jährlich gewechselt hatte. Dies zeugt sicher auch vom Aufschwung, den *NEXTSTEP* in den letzten Monaten erlebt hat, und der hoffentlich noch lange anhalten wird.

### *Neuer Umschlag, neues Layout*

Ich hab nicht schlecht gestaunt, als ich die letzte *PowerKey*-Ausgabe mit dem neuen Umschlag zum ersten Mal in den Händen hielt. Hätte ich vorher gewusst, welch grossartiges Kunstwerk mich auf der dritten Umschlagseite erwartet, so hätte ich bereits im letzten Editorial den "verantwortlichen" Grafiker gebührend erwähnt (was ich hiermit gerne nachhole):

Johannes Labusch, Grafik/Illustration

Zimmergasse 8, 8008 Zürich

Tel. 01 251 52 10, Fax 01 251 92 05

Weitere Aufträge werden gerne entgegengenommen!

Nachdem ich schon früher das Aussehen von *PowerKey* ein wenig verändern wollte und nun endlich eine einigermaßen stabile Version von *PasteUp* besitze, ist es an der Zeit, die ersten "Entwürfe" zu veröffentlichen. Leider war es nicht mehr möglich, alle Artikel neu zu formatieren, und so erscheint ein Teil halt immer noch in der alten Aufmachung. – Anregungen und Verbesserungsvorschläge (aber auch Komplimente) nehme ich gerne entgegen, denn ich bin ja schliesslich kein ausgebildeter Grafiker.

### *PasteUp*

Auch die Version 2.1 von *PasteUp* stürzt zwar noch immer regelmässig ab, speichert aber wenigstens vorher noch schnell eine "CRASH"-Datei. Weit schlimmer wirkt sich jedoch der nur mässig erfolgreiche (und damit unbrauchbare) Trennungsalgorithmus aus; deshalb bitte ich, alle Trennfehler zu entschuldigen! Wenn sich das nicht bald ändern sollte, so werde ich wohl zu *WriteNow* zurückkehren und die Trennungen halt wieder selbst vornehmen. Trotzdem sind die neuen Möglichkeiten faszinierend, die sich durch die Benutzung eines solchen Programms ergeben; auch wenn noch nicht alles zur Zufriedenheit funktioniert. – Dazu *PasteUp*-Tester Labusch: "Dinge, die ich einer Demoversion verzeihe (zB. dass die Bildschirmdarstellung oft nicht oder nicht richtig auf die Eingaben reagiert), sollten bei einem doch recht teuren Programm glatt laufen."

Die Situation ist irgendwie paradox: Da haben wir einen für Grafikanwendungen hervorragend geeigneten Computer mit integriertem Display PostScript, aber bis jetzt hat es noch kein Hersteller geschafft, ein wirklich brauchbares Layout-Programm anzubieten. – War da nicht einmal von *NPE* (*NeXT Publishing Environment*) die Rede? – Wie man einen Abschnitt ansprechend formatiert und dabei korrekt trennt hat uns Donald E. Knuth mit seinem TeX-System schon vor über zehn Jahren vorgemacht!

### *NEXTSTEP 3.2*

Während das "lange erwartete" Windows NT nun endlich erhältlich ist, hat *NeXT* bereits Einzelheiten über die bevorstehende Version 3.2 von *NEXTSTEP* verlauten lassen. Mehr darüber auf Seite ??.

### *Quotes*

"NextStep. When Steve Jobs introduced this software as the operating system for his Next computers in 1988, he claimed it was years ahead of its time. He was right." — Fortune Magazine, June 14, 1993

"How about Windows NT? It, too, is a microkernel-based 32-bit multitasking system for networking environments and requires top-of-the-line hardware. But NextStep is more complete, already established by users and VARs and it is easier to develop applications for NextStep than for NT. NT's interface is Windows, which is already familiar to millions of users. NextStep's interface is better, in my opinion, but it does not come with a large supply of pretrained users." — BYTE, August 1993

**Adriano Gabaglio, Redaktor**

## NeXT und HP

Am 10. August 1993 fand an der ETH Zürich eine von Hewlett Packard (HP) organisierte Informationsveranstaltung zum Thema "NEXTSTEP auf HP Hardware" statt.

Obwohl von HP organisiert, stand diese Informationsveranstaltung ganz im Zeichen von NeXT und der Implementation von NEXTSTEP auf HP Hardware (PCs und Workstations). Roland Dietiker, Education Sales Representative bei HP, begrüßte nur kurz alle Anwesenden, bevor er das Wort an Karl Klarmann, Sales Director Central Europe bei NeXT, weitergab.

### Wo steht NeXT heute?

Im ersten Teil seines Referats erzählte Karl Klarmann einiges über die Entstehungsgeschichte von NeXT sowie den vor kurzem abgeschlossenen Übergang zu einer reinen Software-Firma. Zu diesem Schritt entschloss man sich, nachdem die PC-Hardware immer billiger geworden und deren Leistung fast gleich gut wie bei einer Workstation war. Und dies, obwohl NeXT 1991 noch Platz vier bei der Anzahl verkaufter Workstations innehatte, zwar hinter Sun und HP, aber noch vor IBM und DEC. – So wurde Anfangs Jahr der gesamte Hardware-Bereich an Canon verkauft, während die Wartung der bestehenden Geräte an Bell Atlantic / Sorbus übertragen wurde.

Auch die Organisation wurde leicht reorganisiert, die Logistik an die Firma McQueen übertragen. In Europa besitzt NeXT nur noch zwei Niederlassungen, eine in London und eine in München.

Die Beteiligung an NeXT sieht heute wie folgt aus:

Steve Jobs	46.1 %
Angestellte	23.9 %
Canon	17.9 %
Perot Group	11.3 %
Stanford University	0.4 %
Carnegie Mellon University	0.4 %

### Object + Enterprise

Das Schlagwort bei NeXT hiess von Anfang an *Object*. NEXTSTEP selbst ist von Grund auf objektorientiert, und dahinter stecken über acht Jahre Entwicklungszeit und ein Aufwand von über 100 Mio \$. Nur so ist vermutlich auch der geschätzte zweijährige Vorsprung auf die Konkurrenz zu erklären.

Wieso hat sich NeXT nun HP als Partner ausgesucht? HP ist der weltweit grösste Unix-Lieferant und leistungsmässig führend in der RISC-Technologie. HP besitzt einen sowohl starken als auch stabilen finanziellen Hintergrund und bietet mit seiner *Enterprise So-*

*lution* Gesamtlösungen vom PC bis zum Grossrechner und Drucker. Dies alles führte zum *Object + Enterprise* Zusammenschluss.

### Portable Distributed Objects

NEXTSTEP for Intel Processors ist bereits auf HPs Vectra-PCs erhältlich, und eine NEXTSTEP Portierung für HPs PA-RISC ist auf Mitte 1994 geplant. NeXTs Portable Distributed Objects (PDO), welche auf verschiedenen Plattformen laufen sollen, werden bereits auf Ende 1993 für HP 9000 Serie 800 Server erwartet, ebenso NeXTs Netzwerkverwaltungs-Software NetInfo.

Nach diesem zweiten Teil folge eine kurze Einführung durch Adreas Sautter, Systems Engineer bei NeXT, in die Benutzung von NEXTSTEP und das Arbeiten mit InterfaceBuilder und DBKit (wie gehabt...).

### NeXT in der Schweiz...

Zum Schluss der Veranstaltung erklärte Karl Klarmann, NeXT sei darum bemüht, den Support der Schweizer NeXT Kunden zu verbessern. – Wir lassen uns überraschen!

Adriano Gabaglio

## Sorbus

**Die Firma Sorbus gibt bekannt, dass sie sämtliche auf dem Markt befindlichen NeXT-Ersatzteile europaweit aufgekauft hat.**

Sorbus als herstellerunabhängige EDV-Hardware-Wartungsfirma gehört zur Bell Atlantic Company, welche bereits in den USA die entsprechenden Wartungsverträge von NeXT für Computer und Drucker übernommen hat.

Zu Bell Atlantic Customer Service gehören die verschiedenen Sorbus Gesellschaften in den USA und Europa. Die US-Gesellschaft ist bereits seit über 30 Jahren im EDV-Servicegeschäft tätig. Weltweit ist Sorbus in der EDV-Wartung heute führend. So werden von Sorbus über 3'000 Produkte von mehr als 400 verschiedenen Herstellern gewartet. Bei einem Gesamtvolumen von rund 70'000 betreuten Installationen ist es dann auch nicht verwunderlich, dass Sorbus mehr IBM- und DEC-Geräte wartet als irgendein anderes Unternehmen, mit Ausnahme der Hersteller selbst.

Firmenanschrift:

Sorbus AG, Bösch 41, Postfach, 6331 Hünenberg  
Tel. 042 38 22 88, Fax. 042 36 29 62





## NEXTSTEP 3.2

**Während viele NeXT Benutzer noch mit NEXTSTEP 3.0 (oder gar 2.1!) arbeiten und sich überlegen, ob sich ein Upgrade auf 3.1 überhaupt lohnt, gibt NEXT bereits Einzelheiten zum bevorstehenden Release 3.2 bekannt.**

NEXTSTEP 3.2 wird auf Ende 1993 ("early-mid Fall") erwartet und ist kostenlos für alle 3.1 Besitzer, welche sich bis zum 31.10.93 registrieren lassen. Dieser Zusatz wurde von NeXT erst nachträglich gemacht, vermutlich um alle NEXTSTEP Benutzer dazu zu bewegen, noch schnell auf 3.1 aufzurüsten. Eine Benutzerlizenz kostet \$795 (Fr. 1480.–) und eine Entwicklerlizenz \$1995 (Fr. 3480.–), während man für ein Upgrade \$195 (Fr. 400.–) bzw. \$495 (Fr. 900.–) bezahlen muss. Die Preise sind gleich für die NeXT und die Intel-Version.

Bei diesen Preisen stellt sich vor allem für PC-Besitzer die Frage, ob er/sie nicht doch lieber Windows NT kaufen soll, welches in der Schweiz bereits ab Fr. 844.– erhältlich ist (Preis gültig bis Ende 1993, Advanced Server Version Fr. 2'399.–). – Natürlich ist NEXTSTEP besser, aber bei vielen Käufern dürfte die endgültige Entscheidung zugunsten des "bekannteren" Windows ausfallen, zumal NEXTSTEP für viele PC-Händler leider auch noch kein Thema ist.

**Adriano Gabaglio**

Hier eine Übersicht über die neuen Features von 3.2:

### ***Intel Platform Support Enhancements***

#### **Third-parties Boot Drivers support**

Release 3.2 has been enhanced to enable the dynamic loading of boot drivers for devices such as SCSI drives and CD-ROM drives. This feature will greatly increase the number of configurations NEXTSTEP can support.

#### **Third-parties Sound Drivers support**

A new set of public API's has been added to the SoundKit to enable developers to develop sound drivers that can be dynamically loaded into NEXTSTEP at run time.

#### **32 bit Color support**

The WindowServer has been enhanced to support third-party 32 bit graphics drivers.

#### **Enhanced Pentium support**

Release 3.2 has been enhanced to take advantage of the Pentium processor for better graphics performance.

#### **Additional Graphic Drivers support**

The ATI Graphics Ultra Pro and S3 805 drivers have been enhanced to provide support for additional RAM DAC and resolutions. In addition, S3 928 support has been added to Release 3.2.

#### **Additional Sound Drivers support**

Support for Compaq Business Audio, Microsoft Sound System, and Intel GX Professional on-board sound system has been added to Release 3.2.

### ***Bundled SoftPC Demo***

#### **SoftPC**

A demo version of SoftPC will be bundled with Release 3.2. To unlock this demo version into a full working copy, users can simply call Insignia and purchase a license number to unlock it into a full working copy. NeXT has been working closely with Insignia to develop the best solution possible to run DOS/Windows applications under NEXTSTEP for Intel Processors. In order to optimize the performance of SoftPC, NeXT has enhanced NEXTSTEP's window server to dramatically improve the performance of Microsoft Windows applications in a NEXTSTEP window while providing full cut/paste graphics and text integration with your NEXTSTEP applications. In addition SoftPC will support a full screen mode to allow DOS/Windows applications direct access to the PC's graphics system.



## **Other NEXTSTEP Enhancements**

### **Network Install**

To simplify the installation of NEXTSTEP for Intel processors at large sites, the core technology for network installation have been added to this Release. NeXT will ship some additional tools after 3.2 ships which will enable users and system administrators to install NEXTSTEP with a boot floppy and a network server.

### **Startup and Shutdown Graphics mode**

To improve the consistency of NEXTSTEP's "look and feel", the startup and shutdown of NEXTSTEP will be displayed with a graphical user interface.

### **Configuration cleanup**

The configuration application has been refined to improve "ease of use" and enhanced to support additional configurations.

### **Additional On-line help**

On-line help has been added to the Edit (in developer mode) and PrintManager application.

### **Bug fixing**

Over 500 bug fixes have been made based on customer feedback.

## **NEXTSTEP Developer Release 3.2 Feature Set**

### **HeaderView Application Modifications**

Update Menu Item - This feature allows the developer to reload precompiled headers into the application, if a new version has been created. Previously, developers had to quit the application and then relaunch it to get a newer version. This feature is useful for developers using HeaderViewer to work with their own object libraries, that are dynamically changing.

Kit Filter Preference - A new preference setting has been added to the application to allow more control over the contents being browsed by the developer. This

feature enables object classes to be filtered out that are referenced by other object kits that the developer is browsing, providing the developer with more concise information. For example, a developer may want to only view the DBKit object classes, and not see the AppKit object classes in the browser, even though DBKit includes appkit.h in its definition. The default behavior for HeaderViewer is to italicize the objects in the browser that are included as opposed to being a part of the actual kit.

On-Line Documentation - In addition to the application's online Help facility, full documentation for the application has been added to the online library of technical documentation.

### **ProjectBuilder.app Modifications**

Finder Mode - This is a new view option in the ProjectBuilder main window, to allow a developer to find all instances of words/strings inside of the files that live in the different project categories. For example, a developer could search for a particular word or string in the object header and implementation files within the project. The developer can choose whether the search mechanism uses grep syntax for wildcard searches, or the standard UNIX cshell wildcarding.

Improved Bundle Attribute Support - More support for modifying the attributes of bundle projects has been added. For example, the developer is no longer limited to the .bundle extension, but can create an extension unique to their needs.

Object-Server Support - A new project type has been added to ProjectBuilder, called Tools. This project builds applications that are not dependent on the NEXTSTEP GUI, and therefore do not have the standard AppKit event loop processing, etc. The primary goal of this feature is to allow developers to build object services that run in a network, and can be accessed through DO/PDO via NEXTSTEP client applications. This project type also supports the building of standard UNIX command-line programs that do not interact with the NEXTSTEP Window Server.



Initial PDO Support - ProjectBuilder has always been able to build objects/applications remotely over the network, on another NEXTSTEP machine. Changes have been added to allow ProjectBuilder to now communicate via Portable Distributed Objects (PDO) to a non-NEXTSTEP machine, and build objects/applications in the target environment. This will allow PDO developers to build their object services for their servers from a NEXTSTEP client utilizing the same ProjectBuilder facilities they use for NEXTSTEP applications.

## **FileMerge.app**

FileMerge is a new application being added as a demo this release, with the intent of making it a fully supported and documented application in the next release. FileMerge is used to compare 2 directories, and show which files are added, deleted, or modified. For a pair of files that are different, the application will show them side-by-side in a view that marks all differences in a very graphical manner. The developer can choose to create a third file, which is viewed below the 2 files being compared, and then selectively merge the two files by choosing which of the above files the changes should come from, for each set of differences. Using this tool, a developer can quickly tell what changed, and quickly merge changes from different source branches with a few mouse clicks. This application will also support an open protocol to allow other tools, such as a Software Configuration Management System, to access and utilize it.

## **Gdb Debugger**

Data Breakpoints - Data breakpoints was a feature that existed in NEXTSTEP Release 3.0, but was taken out in Release 3.1 because of its platform dependent implementation. A new platform independent implementation has been added, that is actually more accurate than previous implementations.

## **C++ Support**

libg++ - The GNU libg++, v2.3, C++ class libraries are being added to the release. On-line documentation for the library is being added as well.

ld++ Command Wrapper - This wrapper will prevent C++ developers from having to look at mangled names when there is a link error. The errors will be expressed in the native C++ names.

## **GNU Sources**

A new package, GNUSource.pkg, is being added to the NEXTSTEP Developer product. This includes all the standard GNU sources that NeXT is required to make available to the public. The package will install the sources in the directory /NextDeveloper/Source/GNU/ by default.

In addition to these new features, 3.2 includes both bug fixes and performance enhancements.



## Veranstaltungen

### *NiCE-Meetings*

**Wann:** jeden zweiten Dienstag im Monat  
Beginn 19.00 Uhr (bis ca. 21.30)

**Wo:** ETH Zürich, Hauptgebäude  
Raum HG F3  
Adresse: Rämistr. 101  
Tram 6/9/10, Haltestelle ETH/Uni'spital

**Themen:** Immer aktuell! Werden nach Möglichkeit im **PowerKey** bekanntgegeben.  
Beiträge von Mitgliedern sind jederzeit willkommen.

Falls auch Du gerne mal eines unserer Meetings mitgestalten möchtest (zB. Vorstellen eines von Dir benutzten oder gar selber entwickelten Programmes usw.), so setze Dich doch bitte mit unserem Präsidenten in Verbindung!

### *NiCE-Agenda*

14. Sept.: Meeting  
Themen noch nicht bekannt

28. Sept.: talk & copy

12. Okt.: Meeting: Internet  
Organisation und Benützung  
Simon Poole, CHUUG/EUnet

26. Okt.: talk & copy

9. Nov.: Meeting  
Themen noch nicht bekannt

23. Nov.: talk & copy

14. Dez.: Meeting  
Themen noch nicht bekannt

### *talk & copy*

**Wann:** jeden vierten Dienstag im Monat  
Beginn 19.00 Uhr

**Wo:** Informatik-Gebäude (IFW) der ETH  
Raum IFW A44  
Adresse: Haldeneggsteig 4/Weinbergstr.  
Tram 6/7/10/15, Haltestelle Haldenegg

**Themen:** Hier haben Mitglieder die Möglichkeit, Fragen zu stellen, Erfahrungen auszutauschen sowie die neuste Software zu kopieren. Bring doch einfach Deine Disketten oder besser Deine Festplatte mit! Die NiCE besitzt (fast) alle dazu erforderlichen SCSI-Kabel und -Terminatoren.

## NiCE-Markt

---

### ***NEXTSTEP for Intel Processors 3.1***

User- und Developer-Version, Originalverpackt  
inkl. alle Manuals, gratis Upgrade auf 3.2

Preis: Fr. 1900.–

Christof Marti, Tel. 052 267 77 18 (tagsüber)  
e-mail: christof@mcnext.pr.net.ch

---

### ***NeXTstation, NeXT Laser Printer***

- 250 MB HD, 20 MB RAM, NEXTSTEP 3.1  
- Megapixel Monodisplay, Tastatur, Maus  
- NeXT Laser Printer  
- Preis: Fr. 10'000.–

Katharina Stärk, Tel. 01 462 21 24 P, 031 848 92 52 G  
e-mail: staerk@ivi.ch

---

# Rechtschreibprüfung mit HSDSpell

Seit NEXTSTEP 3.0 werden zwar verschiedene System-Sprachen unterstützt, die mitgelieferte Rechtschreibprüfung kann aber nur englische Texte auf ihre Richtigkeit prüfen. – Wer öfters “fremdsprachige” (dh. nicht-englische) Texte bearbeitet, der wird die Dienste von HSDSpell zu schätzen wissen!



## Command-: und Command-;

Wer hat nicht schon die beiden obenerwähnten Short-Cuts im Bearbeiten-Menü entdeckt? Damit werden das Rechtschreibung-Panel geöffnet bzw. direkt die Rechtschreibprüfung gestartet. – Doch wie funktioniert dieser Dienst?

Rechtschreibprüfprogramme sind nicht wirklich in der Lage, Fehler zu entdecken. Sie vergleichen lediglich jedes zu überprüfende Wort mit allen intern gespeicherten. Kommt es dabei zu keiner Übereinstimmung, so wird das betreffende Wort als unbekannt markiert, und es liegt nun am Benutzer, eine Entscheidung zu treffen. Es kann auch vorkommen, dass ein falsch geschriebenes Wort nicht als solches erkannt wird, da es zufällig mit einem anderen korrekt geschriebenen Wort übereinstimmt (Beispiel: mahlen / malen).

“Weitersuchen” sucht das nächsten unbekanntes Wort im Text. Dieses wird markiert, und manchmal werden sogar Vorschläge zur richtigen Schreibweise angegeben, welche dann direkt übernommen werden können (“Korrigieren” oder Doppelklick).

Ein noch unbekanntes Wort kann in ein persönliches Wörterbuch aufgenommen werden (“Aufnehmen”); oder man kann dem Programm mitteilen, dass das Wort zwar richtig geschrieben ist, aber nicht ins Wörterbuch aufgenommen werden soll (“Ignorieren”).

## Rechtschreibprüfung mit System

Die in NEXTSTEP 3.0 implementierten Klassen NXSpellChecker und NXSpellServer ermöglichen es, eigene Rechtschreibprüfprogramme zu entwickeln, welche von anderen Applikationen aus aufgerufen werden können. Damit wollte NeXT wohl vor allem Drittanbietern eine Chance geben; dies, nachdem man früher sogar gemunkelt hatte, eine Version des Duden würde (ähnlich Digital Webster) bald mit dem System ausgeliefert werden.

Bis jetzt hat aber einzig die amerikanische Firma HSD diese Chance wahrgenommen und bietet ihr Produkt, HSDSpell, in folgenden Sprachen an: Englisch, Deutsch, Französisch, Italienisch, Spanisch, Portugiesisch, Holländisch, Dänisch, Finnisch, Norwegisch und Schwedisch. – Getestet habe ich die Deutsche Version.

Neben Englisch ist eine weitere Sprache im Lieferumfang enthalten, weitere Sprachen können jederzeit dazugekauft werden.



## HSDSpell

Leider unterstützen nicht alle Programme die oben beschriebene Schnittstelle zu HSDSpell! WriteNow bietet zB. eine eigene Rechtschreibprüfung, (allerdings nur in Englisch), ebenso das Layout-Programm PasteUp. – Damit auch diese Programme HSDSpell benutzen können, bietet es seine Dienste auch über das Services-Menü an. Die gemachten Korrekturen müssen aber nachträglich noch im Originaldokument nachgeführt werden!



Bei einem Rechtschreibprüfprogramm kommt es nicht so sehr darauf an, wieviele Wörter es als möglicherweise fehlerhaft erkennt, sondern wieviele Wörter es als richtig erkennt, dh. wie gross sein Grundwortschatz ist! Auch bei einem Programm mit einem grossen Wortschatz müssen meist einige unbekannte Wörter ins persönliche Zusatzverzeichnis aufgenommen werden ("Lernphase"), bis man effizient damit arbeiten kann. – Leider existieren keine Angaben darüber, wie gross dieser Grundwortschatz bei den einzelnen Sprachen von HSDSpell ist. Mein subjektiver Eindruck: Erstaunlich viele Wörter sind dem System bekannt, so dass sich bereits zu Beginn recht angenehm damit arbeiten lässt.

Die Deutsche Version wurde übrigens von Langenscheidt zusammengestellt, und man kann zwischen ss- und ß-Schreibweise wählen. (Wird das Programm als root systemweit zB. in /LocalApps installiert, so muss dies – wegen einem Fehler im Preference-Panel – einmalig und für jeden User einzeln mit "dwrite HSDSpell German 1" in einer Shell geschehen.)

Beim Testen eines neuen Programmes fallen Fehler schnell auf, während die wirklich guten Eigenschaften oft erst mit der Zeit entdeckt werden. – Ich habe folgende Fehler entdeckt:

- Abkürzungen (zB. usw.) werden nicht als solche erkannt, obwohl sie im Wörterbuch aufgeführt sind. Dies liegt daran, dass der Punkt nicht als Teil der Abkürzung erkannt wird. Im obigen Beispiel wird für "usw" als Vorschlag "usw." angeboten, was dazu führt, dass nach der Korrektur zwei Punkte hinter der jeweiligen Abkürzung stehen...

- Wer gewisse Wort-Kombinationen lieber mit Bindestrich schreibt, hat auch nicht immer Glück: HSDSpell bevorzugt die Schreibweise ohne Bindestrich ("Wortkombinationen").
- Aus irgend einem unerfindlichen Grund ist es nicht möglich, die gelernten Wörter zu speichern!!!

Dies ist bereits eine beachtliche Anzahl von Fehlern, wobei der letzte besonders schwer wiegt! Die Entwickler haben jedoch versprochen, all diese Fehler in der nächsten Version zu bereinigen.

## **PowerKey-Wertung**

Insgesamt kann ich HSDSpell jedoch nur empfehlen, insbesondere, falls die beschriebenen Fehler in der nächsten Version auch tatsächlich bereinigt sind. Lediglich der recht hohe Preis wirkt etwas abschreckend: US\$ 195 für die Grundversion (Englisch plus eine weitere Sprache), US\$ 99 für jede zusätzliche Sprache. Das Programm wird von HSD prompt geliefert, der Versand via US Mail in die Schweiz dauerte nur gerade eine Woche (es ist also nicht immer erforderlich, gleich alles via FedEx oder UPS zu versenden, wie dies die meisten amerikanischen Firmen zu tun pflegen).

**Adriano Gabaglio**

Bezugsquelle:  
 HSD Microcomputer U.S., Inc.  
 1095 East Duane Ave., Ste. 209  
 Sunnyvale, CA 94086  
 Tel. (408) 774-1400, Fax (408) 774-1402  
 e-mail: info@hsd.com

# Geheimnisse von NEXTSTEP 3.0

**NEXTSTEP 3.0 ist zwar schon seit einiger Zeit erhältlich, und bereits denken viele an den bevorstehenden Upgrade auf 3.1. Trotzdem veröffentlichen wir hier einen etwas älteren Artikel, der aber einige interessante Tips enthält, die noch lange nicht allen Anwendern von 3.0 bekannt sind!**

## *A Power User's Guide to the Shortcuts, Space-savers, and Hidden Features of NEXTSTEP 3.0*

basierend auf dem gleichnamigen Bericht von Lee Sherman aus dem Magazin NeXTWORLD von Feb/-Mar 1993.

NeXTSTEP 3.0 sieht auf den ersten Blick gegenüber 2.1 recht ansprechend aus. Neue Icons, Online-Hilfe, neue Animationen für Ordnersymbole und Recycler, – vor allem in Farbe kommt unter 3.0 richtig Freude auf. Aber versteckt unter der Oberfläche findet man viele zusätzliche Features, die einem die tägliche Arbeit oft sehr erleichtern konnten, wenn man nur von ihnen wüsste. Und es sind oft die kleinen Dinge, die einem die Arbeit erleichtern, wenn man täglich vor dem NeXT hängt. Zwar gibt es scheinbar immer noch keinen Weg, um aus einem Panel raus zu kommen, ohne die Maus zu benutzen, jedoch wurden ansonsten bereits eine Unmenge an zeitsparenden Tastenkombinationen entdeckt. Manche dieser Tips sind in NeXT's neuem Users Guide aufgeführt, der mit den neuesten Maschinen ausgeliefert wird oder aber für \$50 erhältlich ist, andere sind schlicht und einfach undokumentiert.

### **Shortcuts**

- Hält man die Command-Taste gedrückt und zieht ein File auf ein Application Icon (egal, ob im Dock oder Workspace), so öffnet die Applikation dieses File.
- Hält man die Command-Taste gedrückt und führt einen Doppelklick auf eine Applikation aus, so werden alle anderen laufenden Applikationen ausgeblendet.
- Selektiert man einen Text in irgendeiner Applikation, so kann er nun einfach mit Command-e ins Find-Panel übernommen werden, und man kann direkt mit Command-g und Command-d danach suchen.
- Hält man die Alternate-Taste gedrückt, während man einen Doppelklick auf einen Ordner (bzw. ein .app) durchführt, so wird das gleiche Ergebnis wie "Open as Folder" erzielt.

### **Multitasking ohne Einschränkungen**

- Hält man die Alternate-Taste gedrückt und führt einen Doppelklick auf eine Applikation aus, so wird diese nicht nur in den Vordergrund geholt, falls sie bereits läuft, sondern sie wird erneut gestartet. Es ist somit einfach möglich, eine Applikation mehrmals laufen zu lassen.
- Will man in eine Applikation wechseln, findet aber das zugehörige Programm-Icon nicht, so kann man im Process-Monitor auf das zugehörige Icon klicken.
- Jedem wird bereits aufgefallen sein, dass gestartete Applikationen nun ihre Icons in einer Reihe am unteren Bildschirmrand ablegen. Zwischenräume werden nun korrekt geschlossen. Interessant ist allerdings, dass der Recycler bei gedrückter Command-Taste nun in die linke untere Ecke verschoben werden kann. So erhält man einen weiteren freien Platz im Dock.
- Fenster werden erst beim Loslassen der Maustaste in den Vordergrund geholt.
- Applikationen merken sich nun, wo man ihre Menus zuletzt hingelegt hat. Dadurch muss man sich nicht jedes Mal seine Arbeitsumgebung in einem Programm neu anpassen.

### **Inspector Gadgets**

- Ein sehr gut verstecktes persönliches Lieblingsfeature von mir: Ein Trippelklick auf den Pfad im Inspector selektiert diesen, so dass man ihn z.B. ins Terminal oder Editor übernehmen kann (Command-1).
- RTF und RTFD Format werden im Inspector angezeigt (Command-2).
- Das Fontpanel hat auch zwei neue Features bekommen: Nun lässt sich der Preview-Button durch Klicken und gedrückthalten der Shift-Taste locken, so dass man dann ständig ein Preview der aktuellen Schrift bekommt und sich so durch alle Schriften "WYSIWYG" durchscrollen kann. Ausserdem lassen sich im Textfeld zur Schriftgröße nun mit "+" und "-" relative Schriftgrößen angeben, z.B. "+4": vier Point grösser.
- Das Colorpanel an sich hat sich natürlich auch geändert, eine interessante Neuerung ist allerdings, dass sich ein Farbbutton nun durch Klicken auf dessen Rand locken lässt. So ändert sich die aktuelle Farbe ebenfalls kontinuierlich, ähnlich wie beim Fontpanel. Testen lässt sich diese Funktion gut bei der Hintergrundfarbe des Workspace in den Preferences.



## **Der neue Finder**

Der Finder wurde stark überarbeitet. So kann man jetzt wie im Librarian Files und Ordner in der Shelf ablegen, um sie als Ziele zu benutzen. Ausserdem lässt sich innerhalb von Files nach Daten suchen. Um aber effektiv mit dem neuen Finder arbeiten zu können, gibt es wieder viele versteckte Kleinigkeiten:

- Der Finder unterstützt nun die csh Features zum Editieren der Kommandozeile.
- Find-Button und Return-Taste funktionieren unterschiedlich. Der Button bringt das gesuchte File ins Finder-Window, wohingegen Return das gesuchte direkt startet.
- Wird nach einem Ordner gesucht, so erscheint dieser "highlighted" im File Viewer.
- Die Escape-Taste bewirkt bei Ordnern, dass der Inhalt des Ordners ausgegeben wird, bei Files wird der Pfad ausgegeben.

## **Die interaktive Helpfunktion**

Für reine Anwender und Hilfsbedürftige ist sicherlich die neue interaktive Hilfe eines der wichtigsten Features von 3.0. Sie ist für ein System, das sich kommerziell am Markt durchsetzen will, sicher unverzichtbar, und bei 3.0 ist sie noch besser gelöst als bei Apples 7.0, da sie das eigentliche Arbeiten nicht behindert. Wer jedoch auf dem alten Keyboard die Help-Taste vermisst, der sollte mal die Tastenkombination Control-Alternate versuchen. Klickt man mit dem erscheinenden Fragezeichen auf ein Bildschirmobjekt, so sollte zu diesem eine Hilfe erscheinen.

## **Nachtrag**

Zum Zeitpunkt, als dieser Artikel verfasst wurde, war NeXTSTEP 3.0 gerade erst ausgeliefert worden. Heute dürften die meisten der beschriebenen Features bereits jedem Enduser ausreichend bekannt sein. Ich möchte es mir jedoch nicht nehmen lassen, noch zwei zusätzliche Eigenschaften zu dokumentieren, die beim Verfassen dieses Artikels (mir) noch unbekannt waren:

- Zieht man ein File aus dem Browser in das Terminal, so wird der Pfad zu diesem Objekt in die aktuelle Kommandozeile übernommen.
- Zieht man ein File aus dem Browser in ein Open- oder Save-Panel, so wird der Pfad zu diesem Objekt in das Panel übernommen. Man spart sich auf diese Weise eine Menge doppelter Sucharbeit in einem zweiten Browser.

**übersetzt von Michael Mossal**  
<mmo@sydney.bo.open.de>

## E-mail Verbindung mit der nice (Nachtrag)

Im letzten PowerKey habe ich einen Artikel über das Erstellen einer E-mail Verbindung zwischen dem heimischen NeXT Computer und der nice (unserem Vereins-Computer) geschrieben. Dazu einige zusätzliche Bemerkungen und Korrekturen.

### UUCP Verbindungsaufbau

Ein Unix-Guru hat den Artikel gelesen und dabei zwei kleine Unsauberkeiten festgestellt:

Anstelle von `/dev/cua` (bzw. `/dev/cub`) sollten man überall `/dev/cufa` (bzw. `/dev/cufb`) benützt werden. Dies sollte eine stabilere Verbindung zur Folge haben, da der sendende Rechner auf den empfangenden wartet, wenn dieser überlastet ist.

Im Abschnitt "Telefonverbindung automatisieren" (Seite 18, linke Spalte oben) sollte `ogin:` durch `ogin:--BREAK--ogin:` ersetzt werden. Dies kann bei manchen Arten von Störungen der Telefonleitung den Abbruch des Verbindungsaufbaus verhindern.

Beide Änderungen sind nicht kritisch (bei mir läuft es jetzt seit acht Monaten so wie beschrieben), aber wenn man gerade am eintippen ist, sollte man es gleich richtig machen.

### cu ist nicht mehr zeitgemäss

Ein weiterer Unix-Guru hat darauf hingewiesen, dass die Verwendung von `cu` eigentlich nicht mehr zeitgemäss ist. Dieses Programm stammt noch vom alten AT & T Unix; im BSD Unix (auf dem das NeXT Unix basiert) ist ein leistungsfähigerer Ersatz in Form von `tip` vorhanden.

Naja, mit der Gewohnheit ist das so eine Sache. Für die hier beschriebene Anwendung (Test der Verbindung zur ETH) ist `cu` noch ausreichend leistungsfähig. Wer aber generell unter Unix in andere Rechner einsteigen will, sollte auf jeden Fall `tip` anschauen.

### Langsame Modems

Im Artikel habe ich (Seite 15, linke Spalte oben) erwähnt, wie man mit älteren, langsamen Modems umgehen sollte. Womit ich nicht gerechnet hatte, ist, dass jemand noch versuchen würde, ein 1200 Bit/s Modem zu verwenden. Leider hat `cu` damit ein Problem und bricht ab mit der Fehlermeldung "cu1200: missing phone number".

Im NeXT Unix existiert kein eigenes Programm für `cu`. Statt dessen ruft der Befehl `cu` einfach das Programm `tip` auf. Dessen eigene Konfigurationsdatei

(`/etc/remote`) beinhaltet einen Eintrag für 1200 Bit/s Modems, aber nicht für schnellere. Dieser Eintrag funkt offensichtlich dazwischen. Da das betroffene Mitglied sich inzwischen um ein schnelleres Modem bemüht, hab ich das ganze nicht weiter verfolgt. Wer sonst noch ein 1200 Bit/s Modem hat, sollte sich ebenfalls ein neues besorgen, ausser natürlich, er/sie bezahlt gerne hohe Telefonrechnungen. Wer `tip` verwendet, hat eventuell einen Eintrag für die Modem-Geschwindigkeit; dieser könnte ebenfalls stören.

### Tippfehler im Abschnitt Telefonverbindung automatisieren

Tippfehler sind die Geissel jedes Schreibenden. Wenn es inmitten eines Befehls passiert, ist es für den Leser besonders unangenehm. Leider hat der Redaktor beim Umbrechen des Abschnittes Telefonverbindung automatisieren ein Return-Zeichen zuviel eingesetzt.

Der ganze Text zwischen `nice Any [...] diesei-` `hasp` gehört auf eine Zeile (mit Leerschlägen getrennt), das Return nach `OK` ist fehl am Platz (*mea culpa! Der Redaktor*). Mit dem Return an diesem Ort bleibt der Verbindungsaufbau hängen. Dieser Fehler ist ernsthaft, da das ganze Projekt scheitert, falls man es nicht korrigiert. Falls man den Artikel bereits befolgt hat und wegen dieses Fehlers nichts lief, kann man einfach diesen Abschnitt mit Leerschlag statt Return wiederholen.

### Passwörter – und wie man damit umgeht

Von einem weiteren Unix-Guru gab es einen Aufschrei wegen des im Beispiel abgedruckten Passwortes. Selbstverständlich ist dies nicht das Passwort, das ich auf der nice verwende.

Einer, der von Matthias Heubi ans UUCP anschliessen werden wollte, hat mir (nicht etwa Matthias!) sein erwünschtes Passwort per Mail geschickt. Dazu folgendes:

- Nie ein verwendetes Passwort aufschreiben.
- Nie ein Passwort in einer Datei speichern.
- Nie ein abgedrucktes Passwort benutzen.
- Nie ein Passwort via E-mail verschicken (wird als Datei gespeichert, kann Adressaten verfehlen).



## **Inkompatibilität zwischen meiner und Matthias' Methode**

Auf Seite 14, linke Spalte unten, steht folgende Warnung:

*Achtung: Die hier beschriebene Methode ist nicht identisch mit der Methode von Matthias (seine ist professioneller, diese hier einfacher zum Selbermachen), daher muss man entweder das eine oder das andere machen. Ein Gemisch dürfte wohl nicht funktionieren.*

Diese Warnung war ernst gemeint. Einer hat das nicht geglaubt oder übersehen (vielleicht hätte ich es auch in Schrägschrift setzen sollen). Als Folge davon läuft seine Mail, die früher lief, nicht mehr. Wenigstens wissen wir jetzt definitiv, dass meine Annahme, dass dies nicht funktionieren würde, richtig war.

## **NiCE Internet-Aktion**

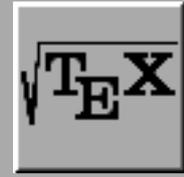
Seit ich den Artikel geschrieben habe, sind ein paar Monate vergangen, in denen viel passiert ist. Für Mail-Interessierte wichtig ist vor allem die NiCE Internet-Aktion. Dabei geht es darum, via CHUUG/E-Unet weltweite E-mail plus alle anderen Internet-Dienste (z.B. NetNews, FTP, Archie) zu bekommen.

Die Verbindung mit UUCP, wie in diesen beiden Artikeln beschrieben und vom NiCE UUCP-Manager Matthias Heubi gemacht, ist nur noch für solche zu empfehlen, die ausschliesslich an Mail mit NiCE Mitgliedern interessiert sind oder aus anderen Gründen nicht an der Internet-Aktion teilnehmen können oder wollen.

## **NiCE ZyXEL-Aktion**

Wer noch kein schnelles Modem hat, sollte sich die Internet-Aktion begleitende NiCE ZyXEL-Aktion anschauen (insbesondere, wenn er/sie an der Internet-Aktion teilnimmt). Das ZyXEL ist eines der momentan schnellsten Modems der Welt. Es ist ausserdem das in der Schweiz verbreitetste schnelle Modem. Dass es erst noch eine PTT-Genehmigung hat und trotzdem bezahlbar ist, grenzt an ein Wunder. Jeder, der mit grossen Datenmengen übers Telefonnetz hantiert, sollte dieses und kein anderes wählen.

**Neil Franklin**



## Was sie schon immer über TeX wissen wollten - Teil 2

Am Ende der letzten Folge haben wir ein Beispieldokument erstellt und nach ein paar (unwesentlichen) Bearbeitungsschritten im Previewer ausgegeben. Natürlich war das nicht ganz fair von mir, denn die eigentliche Magie der neuen Version liegt genau in diesen unwesentlichen Bearbeitungsschritten. Machen wir uns also auf die Suche nach dem geheimnisvollen TeX und seinen Tools.

### Ja wo sind sie denn?

Die Installation hat uns sämtliche ausführbaren Programme ins Verzeichnis */usr/bin* kopiert. Darunter befindet sich so wohlklingende Programme wie *initex* und *virtex*, die zwar schon ziemlich nach TeX tönen, aber noch nicht ganz dem entsprechen, was wir eigentlich suchen. Es scheint weit und breit kein Programm zu existieren das den Namen *tex* oder *latex* trägt und die eigentliche Bearbeitung von Dokumenten mit TeX steuert! Dabei hatte ich doch unter NeXTSTEP 2.x genau dieses jetzt fehlende Programm, das (nach Murphy) durch die Neuinstallation natürlich gelöscht wurde und auf keiner Diskette mehr zu finden ist – ausser vielleicht auf der, mit der sich mein kleiner Bruder und seine neue Knetmasse gerade ein paar vernünftige Stunden bereitet haben! Aber wer will denn da gleich verzweifeln! Don't panic!

### Entwarnung

Der Grund für das beharrliche Nichtvorhandensein ist relativ simpel, denn *tex* und *latex* sind im Grunde genommen nichts anderes, als spezielle Kopien von *virtex* – und *virtex* befindet sich ja bekanntlich in */usr/bin*. Der Name *virtex* steht für *virgin TeX*. Es stellt das TeX-Grundprogramm dar, das allen Implementationen von TeX (oder Erweiterungen davon) gemeinsam ist. Neben grundlegenden Dingen wie Speicherverwal-

tung und File-I/O, stellt *virtex* aber auch die TeX-Grundbefehle (sog. TeX-Primitive) zur Verfügung. Diese haben alle Eigenschaften einer modernen Programmiersprache wie zum Beispiel lokale Blöcke und Variablen, Prozeduren mit Parametern, Rekursivität und I/O-Befehlen. Das TeX-Grundsystem lässt sich jederzeit durch die Definition von Makros (zusammengesetzt aus Grundbefehlen oder anderen Makros) erweitern. Notwendig ist eine Erweiterung allemal, denn mit den Grundbefehlen könnte man sich zwar ein paar hübsche Primzahlen ausrechnen, aber zur Textverarbeitung sind sie kaum zu gebrauchen! Da sich Makros aus TeX-Befehlen zusammensetzen, unterscheiden sie sich eigentlich nicht von TeX-Dokumenten, die wiederum aus TeX-Befehlen und dem darzustellenden Text bestehen. Das hat zur Folge, dass Makros jederzeit innerhalb eines Dokumentes definiert oder umgeändert werden können.

### Makropakete

Eine Ansammlung von Makros ergeben ein Makropaket, das normalerweise in Form eines separaten Files auftritt. Verbreitet sind neben dem Standardpaket *plain-TeX* von Knuth, vor allem *LaTeX* von Leslie Lamport und *AMSTeX* von Michael Spivak. Jedes dieser Pakete liegt zunächst als eine Reihe von Makrodefinitionen in reinem ASCII-Text vor. Müsste TeX diese jeweils Zeichen für Zeichen einlesen, würde einige Zeit vergehen, bevor auch nur ein einziger Buchstabe des eigentlichen Textes gesetzt würde. Da TeX aber eine interne Darstellung von Makros kennt, liegt es nahe, die Makropakete (bereits vorcompiliert) in dieser internen Darstellung einzulesen und sich so die wiederkehrende Arbeit mit der Erkennung von Befehlsnamen zu sparen. Ein solches vorcompiliertes Makropaket, das von *virtex* sehr schnell eingelesen werden kann, nennt man *Format-Datei*.

Zu jedem TeX-System bekommt man wenigstens das File *plain.tex* ausgeliefert. Dieses enthält die Definition von ca. 600 plain-TeX Makros, die wie die TeX-Grundbefehle aufgerufen und verwendet werden können. Tatsächlich sind es die hier definierten Makros, die TeX für die Textformatierung erst handhabbar machen. Bei



*tex*, wie man es gemeinhin kennt, handelt es sich also um das Grundprogramm *virtex*, ergänzt um die Makros, die in *plain.tex* definiert sind.

### ***initex***

Makropakete, wie sie oben beschrieben wurden, können mit dem Programm *initex* vorbearbeitet werden. Das Programm erhält als Eingabe den Namen eines Makropaketes, also einer Datei, die alle Makrodefinitionen enthält, die später bekannt sein sollen. Das Ergebnis dieser Verarbeitung ist eine Format-Datei mit dem gleichen Grundnamen wie das Makropaket und der Extension *.fmt*.

Wie *initex* in der Praxis arbeitet haben wir schon zu Ende der letzten Folge gesehen. Ich möchte trotzdem an dieser Stelle den Aufruf noch einmal wiederholen, um auf eine Besonderheit hinzuweisen. Als erstes sollte man ins Verzeichnis */NextLibrary/TeX/tex/inputs* bzw. */usr/lib/tex/inputs* wechseln, da sich an dieser Stelle die Definitionsdatei *plain.tex* befindet. Als nächstes rufen wir *initex* auf und übergeben als Argument den Grundnamen des Makropaketes, also in unserem Fall *plain*:

```
/usr/lib/tex/inputs> initex plain
This is CTeX, NeXT Version 3.141 (INITEX)
(plain.tex Preloading the plain format:
codes, registers, parameters, fonts, more
fonts, macros, math definitions, output
routines, hyphenation (hyphen.tex)
)
*
```

Der Stern auf der letzten Zeile fordert uns zur Eingabe eines Befehles auf. Die Makrodefinitionen wurden bereits vorverarbeitet und befinden sich (in der erwähnten internen Darstellung) im Speicher. Das gleiche gilt für alle Schriftdefinitionen durch den `\font`-Befehl; diese sind bereits vollzogen. (Die zugehörigen *.tfm*-Dateien werden später nicht noch einmal eingelesen; ein weiterer Punkt der zur Geschwindigkeitssteigerung beiträgt). Um eine Format-Datei zu erhalten, muss der Speicherinhalt in ein File geschrieben werden. Dies bewirken wir durch den `\dump`-Befehl, den wir auf der

Eingabezeile (markiert durch den Stern) eingeben. Der Bildschirmoutput entspricht daraufhin dem in der letzten Folge gezeigten.

Der Grund, warum ich die ersten Zeilen des Bildschirmoutputs oben noch einmal wiederholt habe, liegt in der folgenden Bildschirmmeldung von *initex*:

```
hyphenation (hyphen.tex)
```

Jeder einigermaßen des Englischen kundige, erkennt sofort, dass es sich hier um irgendwas mit Trennung handeln muss. Tatsächlich liegen die Trennmuster für die englische Sprache in Form eines ASCII-Files (*hyphen.tex*) vor. Ein gewagter Blick zeigt uns, dass hier der Befehl `\patterns` in Spiel kommt, auf den ich in einer späteren Folge zurückkommen werde. Für den Moment sei nur soviel gesagt, dass das Laden der Trenntabelle innerhalb *plain.tex* durch den Befehl `\input hyphen` ausgelöst wird, und tunlichst nicht entfernt werden sollte, da TeX sonst überhaupt nicht trennt! Zudem möchte ich darauf hinweisen, dass selbstverständlich auch Trenntabellen für die deutsche Sprache existieren. Nicht nur das, es gibt ganze Makropakete, die eingedeutscht wurden, damit beispielsweise ein Kapitel auch als 'Kapitel' bezeichnet wird und nicht als 'chapter'. Das deutsche Pendant zu *plain.tex* heisst *plaing.tex* und findet sich zusammen mit der deutschen Trenntabelle *ghyphen.tex* auf dem NiCE-Server unter */pub/Library/TeX/GermanDifferences.tar.Z*.

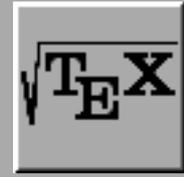
Da der `\dump`-Befehl sowieso jedesmal am Ende eines *initex*-Laufes eingegeben werden muss, kann man den Ablauf etwas verkürzen, indem man einfach

```
/usr/lib/tex/inputs> initex plain \\dump
```

eingibt. Somit entfällt die Eingabezeile mit dem Stern und die Format-Datei wird unaufgefordert gedummt. Führt man dies mit der eben erwähnten deutschen Version von plain-TeX durch, also

```
/usr/lib/tex/inputs> initex plaing \\dump
```

erhält man die Format-Datei *plaing.fmt*, mit deren Hilfe man nun deutsche Texte setzen kann.



## Dem Kinde einen Namen geben

Was haben wir bisher gelernt: *virtex* kennt nur die TeX-Grundbefehle, mit denen sich jedoch Makros definieren lassen. Makros lassen sich in Paketen zusammenfassen und durch *initex* vorcompilieren. Damit erhalten wir *.fmt*-Files, die wiederum von *virtex* eingelesen werden. Genau an diesem Punkt stehen wir im Moment. Auf der einen Seite haben wir *virtex* und die Format-Datei, auf der anderen Seite unser Dokument, also ein Textfile versehen mit TeX-Befehlen und Makros.

Der Aufruf zur Textbearbeitung unter Einbeziehung der plain-TeX Makros kann durch

```
localhost> virtex \&plain <textfile>
```

erfolgen. Möchte man lieber die deutsche Version verwenden, so lautet der Aufruf entsprechend

```
localhost> virtex \&plaing <textfile>
```

Häufig wird statt der eben gezeigten Kommandozeile der entsprechend wirkende Text, z.B. `virtex \&plain $*` für UNIX, unter dem Filenamen *tex* gespeichert. Wird dieses Shellscript dem Betriebssystem gegenüber dann als ausführbar erklärt, so kann mit dem Aufruf

```
localhost> tex <textfile>
```

nunmehr dieselbe Wirkung wie mit der vorangehenden Kommandozeile erzielt werden. Zugegeben, diese Methode erinnert etwas sehr an die Batch-Files unter DOS, aber schliesslich gibt es ja noch weitere Möglichkeiten, so dass man die Batch-Files mit verächtlicher Nichtbeachtung strafen kann.

Benutzer einer C- oder TC-Shell können selbstverständlich einen Alias definieren, der die gleiche Wirkung hat:

```
localhost> alias tex 'virtex \&plain'
```

Um diesen Alias dauerhaft verwenden zu können, sollte man die Definition im File *.cshrc* vornehmen.

Als dritte und cleverste Lösung bieten sich Soft- oder Hardlinks an. *virtex* (in der vorliegenden Version 3.141) überprüft nämlich beim Start mit welchem Programmnamen (`argv[0]`) es gestartet wurde. Hat man zuvor mit

```
localhost> ln -s virtex tex
```

einen Softlink auf *virtex* mit dem Namen *tex* definiert, so ist beim Aufruf des Links `argv[0] = tex`. *virtex* versucht nun innerhalb des Suchpfades der durch die Environmentvariable `TEXTFORMATS` bestimmt ist, eine Formatdatei namens `<argv[0]>.tex`, also in unserem Fall *tex.fmt* zu finden! Wir brauchen also nur noch die Formatdatei *plain.fmt* in *tex.fmt* umzubenennen (auch hier tut's natürlich ein Link) und schon haben wir individuelle Versionen von TeX.

Die vierte und letzte Möglichkeit hat nichts mit zurückgebliebenen Batch-Files oder magischen Links zu tun. Es ist nämlich möglich, das Formatfile, das zur Bearbeitung eines Dokumentes geladen werden soll, innerhalb des Dokumentes selbst zu bestimmen. Um dies zu tun, sollte die erste Zeile im Dokument die Form

```
%&<format name>
```

haben, was zur Folge hat, dass `<format name>.fmt` geladen wird. Zum Beispiel sorgt die Zeile `%&plain` dafür, dass die Formatdatei *plain.fmt* zur Bearbeitung benutzt wird. Diese letztgenannte Variante hat die höchste Priorität aller vorgestellten Möglichkeiten. Egal mit welchem Aufruf oder Kommandozeile TeX gestartet wird, sobald die erste Zeile im Dokument die oben geschilderte Form hat, bestimmt sie allein über die geladene Formatdatei.

## Die Pfadfinder kommen

Im Laufe der Bearbeitung eines Dokumentes durch TeX werden eine Menge Files benötigt, die in den verschiedensten Verzeichnissen liegen können. TeX benützt eine Reihe von Environment-Variablen, die mögliche Suchpfade spezifizieren, und die an die persönlichen Verzeichnisstrukturen angepasst werden können. Eine



Environment-Variable besteht aus einem Namen und einem zugeordneten Inhalt, und kann mit Hilfe des Kommandos `setenv` verändert werden.

Die meisten Environment-Variablen des TeX-Systems sind Suchpfade, also Listen von Verzeichnissen, die durch einen Doppelpunkt voneinander getrennt sind. TeX berücksichtigt die folgenden Environment-Variablen. Falls sie nicht definiert sind, wird der in Klammern angegebene Default-Wert verwendet.

`TEXFORMATS` (`./usr/lib/tex/formats`) Hier sucht TeX nach Format-Dateien.

`TEXINPUTS` (`./usr/lib/tex/inputs`) Hier sucht TeX nach Input-Files. Darunter fallen sowohl Style-Files als auch PostScript-Grafiken, die in ein Dokument integriert werden sollen.

`TEXFONTS` (`./LocalLibrary/Fonts/TeXFonts/tfm: /usr/lib/tex/fonts/tfm`) Hier such TeX nach den `.tfm`-Files.

`TEXPOOL` (`./usr/lib/tex`) Dieser Pfad wird von `initex` verwendet um die Datei `tex.pool` zu finden. `tex.pool` beinhaltet alle Strings des TeX-Grundprogramms, wie z.B. die Fehler- und Systemmeldungen.

`TEXEDIT` (`/usr/ucb/vi +%d %s`) Diese Environment-Variable spezifiziert die Kommandozeile die auszuführen ist, wenn als Antwort auf den TeX Error-Prompt 'e' eingegeben wurde. `%d` wird dabei durch die Zeilennummer ersetzt, in der ein Fehler aufgetreten ist. `%s` dient als Platzhalter für den Filenamen. Wer lieber `Edit` aufgerufen haben möchte, ersetzt die Kommandozeile durch `'/usr/bin/openfile %s:%d'`.

Ein Blick auf die Defaultwerte der Environment-Variablen zeigt uns einen Anhaltspunkt, um die vielen TeX-Files etwas zu ordnen. Verschiebt man alle `.fmt`-Files in ein neu geschaffenes Verzeichnis namens `/usr/lib/tex/formats`, so findet TeX diese, ohne dass wir die Environment-Variable `TEXFORMATS` anpassen müssen. Um das Verzeichnis `/usr/lib/tex/inputs` weiter zu entlasten, empfehle ich, alle Makrodefinitionsdateien `*plain*.tex` sowie die Trenntabellen `*hyphen.tex` in ein spezielles Verzeichnis zu kopieren, da sie normalerweise nur zur Erzeugung der Format-Dateien benötigt werden.

Arbeitet man in einem Mehrbenutzersystem, so möchte man vielleicht für sich (oder für einen beliebigen Benutzer) ein lokales Verzeichnis `~/TeX` erzeugen, um

seine Dokumente darin aufzubewahren. In diesem Fall sollte man die Environment-Variable `TEXINPUTS` so abändern, dass der lokale Pfad zuerst durchsucht wird. Beispiel:

```
setenv TEXINPUTS :~/TeX:/usr/lib/tex/inputs
```

Für die dauerhafte Verwendung ist auch diese Definition im File `.cshrc` vorzunehmen. – Das Arbeiten mit lokalen TeX-Verzeichnissen ist unbedingt zu empfehlen. Wer viel mit TeX arbeitet, weiss, was für eine Flut von verschiedenartigen Dateien für einen einzigen Text kreiert werden. Benützt man gar LaTeX oder BibTeX, so steigt die Anzahl der Dateien noch einmal an. Während der FWW-User (FWW = *Feld-Wald-Wiesen*) in seinem lokalen Verzeichnis seine (lokalen) Styles, Format-Dateien und Sourcen nach eigenem Gusto verändern oder löschen kann, sollten das Verzeichnis `/usr/lib/tex` nur von TeXperten verändert werden können!

## Ausblick

Soviel vorerst zur TeX-Implementation auf der NeXT. Die nächsten Folgen befassen sich zunächst einmal mit den implementationsunabhängigen Aspekten von TeX. Zuerst werden die Grundbefehle zur Textsetzung betrachtet, was eine kleine Theorie zum Thema Seitenaufbau erfordert. Später werden der Mathematik-Satz und die Makro-Programmierung zur Sprache kommen.

Ich kann bereits jetzt die verzweifelten Schreie aus der letzten Reihe hören: “*Wo bleibt LaTeX?*” All den Rufem sei versichert, dass LaTeX natürlich den gebührenden Platz bekommen wird – aber erst wenn die Zeit dazu reif ist! Da LaTeX nichts weiter als eine Makrosammlung für TeX ist, gelten alle in der Zwischenzeit besprochenen Themen uneingeschränkt auch für LaTeX. Zudem kann es für das tiefere Verständnis nur förderlich sein, wenn man einmal gesehen hat, wie plain-TeX arbeitet. Als letztes Argument – um auch den ausdauerndsten Zweifler umzustimmen – sei noch erwähnt, dass man LaTeX erst richtig schätzen lernt, wenn man’s selber einmal in plain-TeX versucht hat!

**Dominik Moser**



## NeXT Laser Printer

Wenn der NeXT Computer eingeschaltet ist, läuft immer auch der angeschlossene NeXT Laser Drucker. Gibt es eine Möglichkeit, den Drucker auszuschalten, wenn man ihn nicht braucht?

Beim NeXT Laser Drucker muss das Ausschalten softwaremässig geschehen. Dazu dient der Befehl **nppower off**, welcher in einer Shell (Terminal) einzugeben ist:

Eine komfortablere Lösung stellt das folgende Shellskript dar:

```
#!/bin/sh
#
# /usr/local/etc/npguard
# Turns off the Local_Printer if it
# has been idle for over 10 minutes
#

PATH=/bin:/usr/bin:/usr/ucb:/usr/etc
export PATH
umask 077

mark=/usr/spool/NeXT/Local_Printer/lock
stamp=/usr/adm/npguard.stamp

if lpq -PLocal_Printer | grep -s Rank; then
:
# Do nothing (not even the ugly logging
# which fills the console window):
# (echo -n 'NeXTPrinter turned off @ ' ; \
# date) >/dev/console 2>&1
else
if [ ! -f $mark ]; then
touch $mark
fi
if [ ! -f $stamp ]; then
touch $stamp
nppower off
exit
fi
find $mark -newer $stamp \
-exec touch $stamp \; -exec nppower off \;
fi
```

Vorgehen: Die Datei `/usr/local/etc/npguard` genau nach Vorlage erstellen und darauf achten, dass nur der User `root` Schreibberechtigung besitzt!

```
% su
Password: geheim (root Passwort eingeben)
# paste > /usr/local/etc/npguard
oder Datei in Edit erstellen...
# chmod 700 usr/local/etc/npguard
# exit
```

Zusätzlich muss in der Datei `/etc/crontab.local` folgende Zeile angefügt werden (als eine Zeile eingeben und mit Return abschliessen!):

```
1,11,21,31,41,51 * * * * root
/bin/sh /usr/local/etc/npguard >>
/tmp/console.log 2>&1
```

Viel Spass beim Ausprobieren!

## Komprimierte Dateien

Frage: Nach welchem Verfahren wurden die vom *WorkspaceManager* erzeugten Dateien mit der Endung *.compressed* komprimiert?

Bei den Dateien mit der Endung *.compressed* handelt es sich um gewöhnliche *.tar.Z* Dateien, welche auch folgendermassen erzeugt werden können:

```
tar cf- Auswahl | compress > Auswahl.compressed
```

Die *Auswahl* kann sowohl eine einzelne Datei, aber auch ein ganzer Ordner (Verzeichnis) sein. Der Inhalt einer komprimierten Datei lässt sich auflisten mit

```
cat Auswahl.compressed | zcat | tar tf-
```

Neuerdings sieht man immer mehr Dateien mit der Endung *.z* bzw. *.gz*. Diese wurden mit dem Programm **gzip** komprimiert und müssen daher mit **gunzip** dekomprimiert werden; **uncompress** funktioniert nicht! Die Kompressionsrate bei **gzip** beträgt ca. 60 – 70 %, bei **compress** "lediglich" 50 – 60 %.



## comp.sys.next.bugs

### Hier zwei (nützliche) Beispiele aus dem riesigen Internet Newsgroup-Angebot:

From: huhr@wien.uni-paderborn.de  
(Holger Uhr)  
Subject: **Bugfix für Listingmode  
im Workspace 3.0**  
Date: 4 Dec 1992 10:29:27 +0100  
Organization: Universität Paderborn, Germany

Marcel Waldvogel (...) wrote:

Wer sich schon über die falsche Uhrzeitangabe im (deutschen) Listingmodus des Workspace geärgert hat, kann in

```
/usr/lib/NextStep/Workspace.app/WM.app \  
/German.lproj/MainWindow.strings
```

das "%Std" durch ein "%H" ersetzen (da war wohl ein überfleissiger Übersetzer am Werk :-))

Und wenn man in der Zeile darüber noch das %J durch %Y ersetzt, bekommt man sogar richtige Jahreszahlen statt eines Fragezeichens.

From: gad@eclipse.its.rpi.edu  
(Garance A. Drosehn)  
Subject: **Installation of NS-3.1 includes  
about 30 meg of useless files**  
Date: 30 Aug 1993 02:33:18 GMT  
Organization: Rensselaer Polytechnic Institute,  
Troy NY, USA

I finally got around to doing some of my "Snapshot HD" tests on the NS-3.1 release for NeXT hardware. As others have noted, the installation of NS-3.1 takes up about 40 meg more disk space than NS-3.0 did.

A large chunk of that is due to the same installation oversight that NeXT had in NS-3.0, only it's even more noticeable in NS-3.1. The first place to look is in */NextLibrary/Receipts*. Do you notice any pkg-file receipts in there for languages that you did not install? Something about the installation process leaves around receipts for all the languages, including the ones that you haven't installed.

To make matters worse, if you *\*don't\** install the language then the pkg receipt includes the tar.Z file that could be used to install it. Due to this, the German.pkg receipt takes up over 8 meg of disk space. The French.pkg file takes up another 8 meg. Most of the other language files only take up about 1.5 meg each. Note that this tar file does you absolutely no good, since the installer won't let you install from these pkg files.

To remedy this: logon as root, and click on each pkg file in */NextLibrary/Receipts* that you do *\*not\** want to have installed. When the Installer.app starts up, click on the button to delete the package. If you *\*do\** want to have the package installed, then install it from pkg files on the CD-ROM disc and it will put in a new receipt which will be much smaller.

Another file that might be wasting disk space on you is in */NextLibrary/References/Webster-Dictionary/CompactDictionary*. This file is over 14 meg. If you have installed the Literature package, then you will see a file called Dictionary in the very same directory. *\*If\** you have that Dictionary file, then the CompactDictionary file is not needed for anything (not that I can tell at least). You can destroy it.

Note that if you don't have a Dictionary file in that directory, then Websters is using this CompactDictionary (which has less info) so you wouldn't want to delete it.

For those that are curious about all these "snapshots", an assortment of these snapshots (as well as the script that I use to generate the snapshots) are available via anonymous FTP in the directory */NeXT/lis\_output* off the host *eclipse.its.rpi.edu*. The snapshots are just a bunch of "du -s" and "find -ls" commands done from root, so you get a picture of what files are where, and how much disk space is being taken up by various things. Each snapshot is a directory of files.

I'm pretty sure I reported this to NeXT for release 3.0, but I haven't reported that the oversight is still there in release 3.1. It's more noticeable in NS-3.1, as the German and French packages have grown by about 6 meg each.



**Herausgeber:** NiCE – NeXT User Group

**PowerKey** ist das Magazin der NiCE und erscheint 5 – 6 mal jährlich. Ein Abonnement ist in der Mitgliedschaft bei der NiCE enthalten. **PowerKey** wird vollständig auf NeXT-Computern mit *Write-Now* erstellt; diese Ausgabe zusätzlich mit *PasteUp 2.1* (na ja...).

Auflage: 250 Exemplare • Einzelverkaufspreis: Fr. 7.–

## Redaktion:

*Verantwortlicher Redaktor:* Adriano Gabaglio

*Mitarbeiter dieser Ausgabe:* N. Franklin, D. Moser.

“Wir bemühen uns, sowohl die männliche als auch die weibliche Schreibform zu verwenden. Wo wir dies zugunsten einer besseren Lesbarkeit nicht tun, beziehen sich sämtliche Aussagen auf Männer und Frauen.”

## Redaktionsadresse:

Adriano Gabaglio, Brunnmattstr. 22a, 6010 Kriens

Anfragen und Inserate von Mitgliedern bitte nur schriftlich!

## Verlag und Inseratenaufgabe:

Matthias Heubi, Postfach 106, 8621 Wetzikon 4

Adressänderungen bitte an den Aktuar!

## Anzeigenpreise:

1 Seite A4 Fr. 300.– • 16 × 10 / 8 × 20 cm Fr. 175.– • 8 × 10 cm Fr. 100.– • 8 × 5 cm Fr. 60.– • Mengenrabatt bereits ab 2 Ausgaben!

Einmalige, nicht gewerbmässige Inserate von Mitgliedern gratis.

## Copyright:

Copyright aller Artikel bei NiCE, ausgenommen Artikel vom Internet (bezeichnet) bei den entsprechenden Autoren. Die gewerbliche Nutzung, insbesondere der Programme, Schaltpläne, gedruckten Schaltungen und Adressen von Mitgliedern, ist nur mit schriftlicher Genehmigung des Herausgebers zulässig. Nachdruck, auch auszugsweise, nur mit schriftlicher Genehmigung des Herausgebers. Warennamen werden ohne Gewährleistung der freien Verwendbarkeit benutzt. Für unverlangt eingesandte Manuskripte übernimmt die Redaktion keine Haftung. © 1993 NiCE – NeXT User Group.

## Haftung:

Der Herausgeber lehnt jegliche Haftung für direkte und indirekte Schäden oder Folgeschäden ab. Für abgedruckte Tips und Anleitungen kann keine Garantie übernommen werden. Eine Haftung für die Richtigkeit der Veröffentlichungen kann trotz sorgfältiger Prüfung durch die Redaktion vom Herausgeber nicht übernommen werden.

## NiCE – NeXT User Group

**Vereinsadresse:** NiCE – NeXT User Group  
Rechenzentrum IFW A44  
ETH Zentrum  
8092 Zürich

**Bankverbindung:** Migrosbank, 8023 Zürich  
PC 80–533–6  
Konto-Nr. 16 719.247.3/06

## Vorstand

**Präsident:** Neil Franklin  
Morgenweg 8, 8404 Winterthur

**Vizepräsident:** Tom Heynemann  
Unt. Briggerstr. 61, 8406 Winterthur

**Aktuar:** Albin Mächler  
Jonas Furrer-Str. 97, 8400 Winterthur

**Kassier:** André Muff  
Dorfstr. 26, 6005 Luzern

**Redaktor:** Adriano Gabaglio  
Brunnmattstr. 22a, 6010 Kriens

**Verleger:** Matthias Heubi  
Motorenstr. 131, 8621 Wetzikon

**Beisitzer:** Patrik Lori  
Schumacherweg 44, 8046 Zürich

**Beisitzer:** Thomas Röttig  
Forchstr. 194, 8032 Zürich

**e-mail Vorstand** [vorstand@nice.usergroup.ethz.ch](mailto:vorstand@nice.usergroup.ethz.ch)  
**oder einzeln:** [<vorname>@nice.usergroup.ethz.ch](mailto:<vorname>@nice.usergroup.ethz.ch)

**Mailbox:** Tel. 01 251 20 02  
# **call nice**  
nice login: **mailbox**

**PowerKey 6/93 erscheint Ende November 1993**

Redaktions- und Anzeigenschluss: 31. Oktober 1993