

Textverarbeitung mit Shells: LINGUIX

Das Betriebssystem UNIX ist bekannt für die Fülle seiner Dienstprogramme, worunter sich auch mehrere Systeme zur Textverarbeitung finden. Auf dem ST beherrschen interaktive, graphisch orientierte Textsysteme den Markt. Der Münchner Hans Kaufmann nahm sich UNIX zum Vorbild und schrieb die Programmsammlung LINGUIX, die als Public-Domain und - mit erweitertem Leistungsumfang - kommerziell erhältlich ist. Für Liebhaber kommando-orientierter Shells wie Guläm bietet sich damit eine Alternative zur "konventionellen" ST-Textverarbeitung.

Im Gegensatz zu den ST-Compute 7-8/90 vorgestellten Systemen ist LINGUIX kein eigenständiges Programm - es handelt sich um eine Utility-Sammlung, die in Zusammenarbeit mit einer Shell und Editoren ihre Wirkung entfaltet. Um Funktionen auf Texten auszuführen muß man die jeweiligen Programme mit Optionen versehen aufrufen. LINGUIX bildet also ein kommando-orientiertes Textsystem.

Kommandeur statt Mäuschen

Hat man in einer interaktiven Textverarbeitung ein Programm laufen, dessen einzelne Funktionen man durch Befehle respektive Mausclicks anwählt, schickt man seinen Text in einer kommando-orientierten Umgebung durch mehrere Programme, die jeweils einen neuen, entsprechend verarbeiteten Text erzeugen. Durch geschickte Kombination der einzelnen Programme und entsprechende Shell-Skripts erreicht man eine Flexibilität, die eine interessante Alternative zur interaktiven Arbeit bildet. Sind die verwendeten Textfilter mit hinreichend vielen Optionen ausgestattet, kann ein kommando-orientiertes System durchaus einem Texteditor ohne Makrosprache - wie WordPlus - überlegen sein.

Allerdings ist ein solches Textsystem völlig ungeeignet für Anwender, die nach dem What-You-See-Is-What-You-Get-Prinzip WYSIWYG - also der Darstellung des druckfähig formatierten Dokuments schon während der Texteingabe - arbeiten wollen, überhaupt nicht geeignet. Man muß wissen, welches Aussehen das Dokument nach Ausführung eines Filterprogramms haben wird und man muß natürlich auch wissen, welche Optionen nötig sind, um die gewünschte Formatierung zu erzeugen. LINGUIX stellt also höhere Anforderungen und erfordert eine längere Einarbeitung als ein sofort zu benutzendes System wie WordPlus.

Schließlich ist natürlich zu berücksichtigen, daß in einem interaktiven Textsystem die Kommandos auf dem schon im Speicher liegenden Text arbeiten. LINGUIX arbeitet dagegen auf Dateien, die jedesmal neu gelesen und geschrieben werden müssen. Logisch, daß die Verarbeitungsgeschwindigkeit dabei entscheidend von der Geschwindigkeit des Laufwerks abhängt. Mit einem reinen Diskettensystem wird

die Arbeitsleistung eher unbefriedigend sein.

Die Installation beschränkt sich auf das Hochkopieren der Utilities auf Festplatte (oder ein Arbeitslaufwerk). Sollen die LINGUIX-Programme in einem extra Ordner stehen, muß noch der Suchpfad der verwendeten Shell gesetzt werden. Beim Guläm ist dies die Variable PATH. Insgesamt werden circa 330 KB Plattenplatz benötigt.

Mit Filter

Einfache Anzeigen von ASCII-Dateien ist mit *head* und *cat* möglich. Zum Suchen gibt es *lgrep*, das allerdings leider nicht mit den von UNIX bekannten Regular-Expressions arbeitet. Will man die Umgebung des gefundenen Worts sehen, kann man sich mit *kwic* und *twic* auch Zeilen vor und nach der Fundstelle anzeigen lassen.

Mehrere Filter erlauben Ersetzungen in einer Textdatei. So wandelt *tabtosp* Tabulator-Zeichen in Leerzeichen und mit *wysilix* lassen sich Schriftattribute in einem WordPlus-Dokument verändern. Letzteres dient übrigens eigentlich der Erzeugung eines Registers: Alle Textteile mit einer bestimmten Attributkombination - also beispielsweise hell und unterstrichen - werden mit einer Seitenangabe in eine Datei geschrieben und im Ursprungstext die Attribute gelöscht.

Mit *wpsix* kann man in ähnlicher Weise ein Register aus ASCII- oder WordPlus-Dateien vorbereiten. Dabei kommt eine zweite Datei zum Einsatz, in der die Worte vermerkt sind, die in den Index aufgenommen werden oder - als Ausnahmeliste - dort eben nicht auftauchen sollen.

Die erzeugten Registerdateien werden anschließend mit *forminx* endgültig aufbereitet, also die Seiteneinträge für ein Stichwort zusammengefaßt. Sehr schön ist die automatische Erzeugung von "12f" aus den Seitenangaben 12 und 13, sowie "12-14", falls für das Stichwort die Seiteneinträge 12, 13 und 14 vorkamen. Und hervorragend ist die Fähigkeit von *forminx*, Stichworte, die sich nur in der Endung unterscheiden automatisch zusammenzufassen. Gibt es Einträge "Gazelle 12" und "Gazellen 14", macht diese "Flexionskontrolle" daraus eine Indexzeile "Gazelle 12,14".

Auch Inhaltsverzeichnisse lassen sich aus WordPlus-Dokumenten erzeugen. *wpinv* erkennt eine Überschrift an ihrem Beginn mit einer Zahl und Unterstreichung. In diesem Fall wird die Zeile in das Inhaltsverzeichnis mit Seitennummer übernommen. Bei der Berechnung der Seitennummern liefert die Verwendung des 1,5-fachen Zeilenabstandes übrigens falsche Werte.

Etwas ausgefallener ist der Filter *ereim*: Er gibt jeweils das letzte Wort einer Textzeile aus, auf Wunsch auch gespiegelt. Anwendungsbereich ist die Verarbeitung von Gedichten - nach einem Lauf von *ereim* hat man alle verwendeten Reimpaare

gesammelt.

Zeilenorientiert sind auch von UNIX bekannte Filter wie *sort*, das eine Datei sortiert - dankenswerterweise wird die deutschsprachige lexikographische Ordnung berücksichtigt (würde man nur nach Zeichencodes sortieren, kämen die Umlaute nach dem Z). *uniq* gibt die Textzeilen der Eingabedatei jeweils nur einmalig aus. Schreibt man die Ausgabe in eine Datei, hat man praktisch alle doppelten Zeilen entfernt. Mit *zdiff* kann man sich die Zeilen ausgeben lassen, die nur in Datei a auftreten, nicht aber in b. Die Ausgabe entspricht damit allen Zeilen, in denen sich a von b unterscheidet. Mit einem Schalter können auch alle Unterschiede von b zu a ausgegeben werden.

Zwischen dem ASCII und WordPlus-Format können Dateien mit *ttow* und *wtot* umgewandelt werden. Bei der Wandlung nach ASCII kann über Schalter der Blocksatz unterdrückt und die Behandlung von Fußnoten gesteuert werden. Normalerweise werden die Fußnoten aus WordPlus am Ende der ASCII-Datei als Endnoten gesammelt ausgegeben.

Da WordPlus-Dokumente schon in ASCII gewandelt werden können, liegt es nahe, daß LINGIUX sie auch ausdrucken kann. *wprint* verwendet dabei die gewohnte Druckerinstallation in PRINTER.CFG. Praktischerweise wird diese Datei auch in den Verzeichnissen gesucht, die man in den Shell-Variablen ETC und PATH angegeben hat. Das Programm beherrscht keinen Grafik-Ausdruck und kann leider auch nicht in eine Datei drucken.

Hat man eine einfache ASCII-Datei erstellt, kann man sie mit *clmns* zu einem Mehrspaltendokument verarbeiten. Dabei ist die Anzahl der Spalten und deren Abstand über Schalter wählbar.

reunit, das zwei sortierte Eingabedateien zu einer wiederum sortierten zusammenmischt, ermöglicht auch eine Tabellenverarbeitung. Dabei enthalten die Eingabedateien eine Text- und ein Wertespalte. Sind in beiden Eingabedateien gleichlautende Einträge in den Textspalten vorhanden, erscheint in der Ausgabedatei die Summation der beiden Wertfelder. Vielleicht sollten sich hier durch einen Schalter noch weitere arithmetische Funktionen abrufen lassen.

Das Paket umfaßt noch eine ganze Reihe weiterer Utilities, die weitere Möglichkeiten zur Textverarbeitung anbieten. Über sämtliche Programme können Sie sich in der Public-Domain-Version informieren.

Da das Inout-System von TURBO-C - mit dem die Programme implementiert wurden - nicht hundertprozentig mit der TOS-Ausgabenumlenkung harmoniert, gibt es bei allen Kommandos den Schalter -o, mit dem die Ausgabe in eine Datei geschrieben wird.

Handbuch

Wie bei Produkten aus dem Shareware-Bereich üblich, gibt es auch für LINGUIX kein gedrucktes Handbuch. Die Anleitung liegt vielmehr als 133 KB großes Dokument auf der Diskette bei. Nach dem Ausdruck hat man einen fast 70-seitigen Text vor sich, der erfreulicherweise mit einem dreiseitigen Register versehen ist.

Nach einer kurzen Einführung und einem Überblick der angebotenen Utilities folgt der Hauptteil der Anleitung. Hier finden sich alle Kommandos in alphabetischer Reihenfolge beschrieben. Jedem Utility ist mindestens eine Seite gewidmet, auf der in UNIX-Manier das Kommando mit Arbeitsweise, Schalterbedeutungen, meistens einem Beispiel und eventuellen Querverweisen auf andere Kommandos beschrieben wird.

Abschließend werden einige Skript-Prozeduren (Batch-Files) für Guläm beschrieben, die beispielsweise die Erzeugung eines Registers auf die Ausführung einer Kommandozeile reduzieren.

Fazit

LINGUIX ist eine interessante Sammlung kommando-orientierter Utilities, die - so man ein solches System bevorzugt - effiziente Textverarbeitung ermöglicht. Mit der Public-Domain-Version hat man eine kostenlose Möglichkeit, den Funktionsumfang zu testen. Entspricht das System der individuellen Arbeitsweise, lohnt sich der geringe Aufwand für die registrierte Version. Alles in allem ein - für ST-Verhältnisse - unkonventionelles, aber funktions- und leistungsfähiges Textsystem; durchaus empfehlenswert.

Sie finden die PD-Version von LINGUIX auf Diskette 248 der MAXON-PD-Sammlung. Die registrierte, erweiterte Version können Sie gegen DM 40,-, formatierter Diskette und frankiertem Rückumschlag beim Autor erhalten:

*Hans Kaufmann
Einsteinstr. 151
8000 München 80*