

Literarische und Dokumentarische Datenverarbeitung

Neuerungen im TUSTEP-Editor ab Version Februar 1992

Ab der Version Februar 1992 bietet TUSTEP neue Verbesserungen. Sie betreffen die Darstellung der Zeichen im Editor, Groß- und Kleinschreibung in Austauschweisungen und Parametern, die Einstellung der Zeilenlänge im Editor und neue Datei-Informationen.

Neue Modi P und T im Editor

Im Editor gibt es nur noch die Modi T und P; Modus M und Modus O wurden abgeschafft. Der einzige Unterschied der beiden neuen Modi ist die Numerierung der Sätze. Die Darstellung der Zeichen und Sonderzeichen (einschließlich der mit ^ codierten Zeichen), sowie die möglichen Angaben in Zeige- und Austauschweisungen sind nun gleich. Der neue Modus P entspricht dem alten Modus M.

Da die Codierung der Zeichen gleich ist, hängt die Wahl des geeigneten Modus nur noch davon ab, welche Möglichkeiten die unterschiedlichen Satznummern bieten.

Charakteristika von Modus P:

- Satznumerierung: nur Zeilennummern, bis zu zweistellige Unterscheidungsnummern
- Zwischen Sätzen mit der Standardnumerierung lassen sich 99 neue Sätze einschieben.
- Abspeichern in Segmentdatei möglich
- Mehr Platz für den Text in der Bildschirmzeile als bei Modus T
- Satznummern lassen sich als Bereichsangabe im TUE-Kommando verwenden, um nur einen Teil der Kommandofolge auszuführen. Für Programme (Kommandofolgen) ist der Modus P vorzuziehen.

Anmerkung:

Die Codierung der Zeichen erlaubt problemlos das Schreiben von Texten im Modus P. Modus P ist notwendig, wenn man Texte in Segmentdateien abspeichern will.

Charakteristika von Modus T:

- Satznumerierung: Seitennummer und Zeilennummer, bis zu dreistellige Unterscheidungsnummern
- Zwischen Sätzen mit der Standardnumerierung lassen sich 999 neue Sätze einschieben.
- Daten lassen sich seitenweise ansprechen (in Anweisungen, Editor-Makros, Parametern)
- Satznummern können in Kommandofolgen

für die Auswahl der Daten über Parameter direkt verwendet werden.

Zeigen der Daten mit und ohne Satznummer

Bei Modus T und P läßt sich durch ein vorangestelltes + oder - bei der Modusangabe wählen, ob die Daten mit oder ohne Satznummer gezeigt werden.

Modi +T, +P:

Die Daten erscheinen auf dem Bildschirm wie gewohnt mit Satznummern. Dies ist die Voreinstellung.

Modi -T, -P:

Die Daten erscheinen auf dem Bildschirm ohne Satznummern: man hat damit die volle Bildschirmbreite für den Text zur Verfügung. Den Text kann man anschauen, durchblättern, durchsuchen und mit Anweisungen bearbeiten. Es sind jedoch *keine Korrekturen* der Daten direkt am Bildschirm möglich.

Modus -T wird z. B. beim Edieren des Zweitprotokolls mit dem Kommando #prot,ed eingestellt.

Darstellung der Akzente

Wenn mit DEFINIERE (vgl. unten) ein Code eingestellt ist, der Akzentbuchstaben beinhaltet, läßt sich bei allen Modi durch ein nachgestelltes + oder - bei der Modusangabe (+T+, -T+, +T-, -T-, bzw. +P+, -P+, +P-, -P-) die Darstellung der Akzentbuchstaben (und einiger Sonderzeichen) wählen. Dies betrifft nicht die nicht-lateinischen Schriften; für sie wird weiterhin der FILTER benötigt.

Modi +T+, -T+, bzw. +P+, -P+:

Wird bei der Modusangabe hinter dem Buchstaben ein + angegeben, so werden Akzent-

buchstaben (und einige Sonderzeichen) auf dem Bildschirm – soweit möglich – graphisch dargestellt.

Die Eingabe der Sonderzeichen erfolgt – soweit vorhanden – über entsprechende Tasten (z. B. §) oder Tastenkombinationen (z. B. für die häufigsten Akzentbuchstaben). Für die übrigen Sonderzeichen erfolgt die Eingabe wie bisher über die TUSTEP-Codierung. Bei der nächsten Zeige-Anweisung erscheinen diese Zeichen in ihrer graphischen Form.

In der Datei wird in jedem Fall die TUSTEP-Codierung abgespeichert. So bleibt die 100%ige Kompatibilität der Daten erhalten, unabhängig davon, ob ein Bildschirm die Sonderzeichen darstellen kann und ob die Tastatur deren Eingabe erlaubt.

Welche Zeichen der Bildschirm darstellen kann, ist abhängig vom Gerät und von der Einstellung (vgl. unten »Definieren des Codes«).

Beispiel:

Sie tippen á (' gefolgt von a) und sehen sofort á; in der Datei wird %/a abgespeichert.

Sie tippen %?n und sehen zunächst die TUSTEP-Codierung auf dem Bildschirm. Nach dem nächsten Zeigen sehen Sie ñ. In der Datei wird %?n abgespeichert.

Anmerkung:

Beim Suchen von Zeichenfolgen werden wie bisher die in der Datei gespeicherten Zeichen gesucht, die ggf. in dieser Form auf dem Bildschirm erscheinen.

Sie suchen a, finden u. a. %/a und bekommen es in dieser Form auf dem Bildschirm gezeigt.

Modi +T-, -T-, bzw. +P-, -P-:

Wird bei der Modusangabe hinter dem Buchstaben ein - angegeben, erscheinen die Akzentbuchstaben und Sonderzeichen auf dem Bildschirm in der Form, wie sie in der Datei abgespeichert sind.

Einstellen des Modus

Der Modus kann mit dem Kommando EDIERE als Angabe zur Spezifikation MODUS oder mit der Anweisung M eingestellt werden.

Neue Voreinstellung:

Bei leeren Dateien ist die Angabe des Modus im Kommando EDIERE nicht mehr obligat. Voreinstellung für leere Dateien ist: MODUS=+P+.

Wird innerhalb des Editors mit Anweisung D in eine leere Datei gewechselt, wird ebenfalls Modus +P+ eingestellt.

Die Angabe des Modus muß nicht vollständig mit allen + und - erfolgen.

Schreibt man nur den Buchstaben T oder P, so werden für + und - die bisherigen Angaben beibehalten, bzw. es wird die Voreinstellung +T+ bzw. +P+ genommen.

Soll nur die führende Angabe für die Darstellung der Satznummern gewechselt werden, braucht das abschließende + bzw. - nicht geschrieben zu werden. Bei der Editor-Anweisung genügt es in diesem Fall M, + bzw. M, - zu schreiben. Für die fehlenden Angaben werden die bisherigen Einstellungen übernommen.

Soll die hintere Angabe für die Sonderzeichen gewechselt werden, muß der Modus vollständig (z. B. +P-) angegeben werden.

Definieren des Codes zur Zeichendarstellung am Bildschirm

Beim Kommando DEFINIERE gibt es zur Spezifikation CODE neue Spezifikationswerte, um die Codierung der Zeichen für Ein- und Ausgabe mit Tastatur und Bildschirm zu regeln.

Die folgende Beschreibung bezieht sich auf TUSTEP unter MS-DOS. Wird ein PC als Terminal verwendet, vergleiche dazu den nachstehenden Artikel.

#DEF, CODE=IBMPC

Umlaute und ß können direkt eingegeben werden. Diese Zeichen werden auf dem Bildschirm dargestellt. Akzentbuchstaben und Sonderzeichen sind nur in der TUSTEP-Codierung eingeb- und darstellbar. CODE=IBMPC ist gegenüber früher unverändert.

#DEF, CODE=CP437

Bei der Eingabe und Bildschirmdarstellung werden die Zeichen (Akzentbuchstaben, Sonderzeichen) der Code Page 437 unterstützt. Im Betriebssystem MS-DOS muß die Code Page 437 eingestellt sein (vgl. MS-DOS-Befehle *chcp*, *mode*, *country*).

#DEF, CODE=CP850

Bei der Eingabe und Bildschirmdarstellung werden die Zeichen (Akzentbuchstaben, Sonderzeichen) der Code Page 850 unterstützt. Im Betriebssystem MS-DOS muß die Code Page 850 eingestellt sein.

Die Wahl der Code Pages in TUSTEP ist von der Systemeinstellung abhängig. Die richtige Ein-

stellung Code Page in TUSTEP läßt sich am Bildschirmprotokoll des Kommandos DEFINIERE kontrollieren.

Stellt man im System die Code Page neu ein, ist 850 zu empfehlen, da sie mehr Zeichen enthält, die üblicherweise in der TUSTEP-Umgebung gebraucht werden.

Auswirkungen der Codeeinstellung für das Kommando UMWANDLE

Beim Umwandeln von SDF- in TUSTEP-Dateien und umgekehrt mit dem Kommando UMWANDLE erfolgt die Umcodierung der Zeichen normalerweise entsprechend dem mit DEFINIERE eingestellten Code. Für die drei oben genannten Codes gibt es davon eine Abweichung.

Wird beim Kommando UMWANDLE zur Spezifikation CODE nichts angegeben, erfolgt

die Umcodierung wie bei CODE=IBMPC, unabhängig davon, welcher der drei genannten Codes eingestellt ist. Beim Umwandeln einer SDF- in eine TUSTEP-Datei werden Akzentbuchstaben durch ihren Hexadezimal-Code in der Form #[xx] dargestellt. Beim Umwandeln einer TUSTEP- in eine SDF-Datei bleibt für die Akzentbuchstaben die TUSTEP-Codierung (z. B. %?n) erhalten. Umlaute werden in beiden Fällen umgewandelt.

Sollen Akzentbuchstaben in beiden Richtungen entsprechend der eingestellten Code Page (CP437 oder CP850) umgewandelt werden, so ist beim Kommando UMWANDLE die Spezifikation CODE=+ anzugeben. Beim Umwandeln einer SDF- in eine TUSTEP-Datei wird dann aus dem Akzentbuchstaben die entsprechende TUSTEP-Codierung, beim Umwandeln einer TUSTEP- in eine SDF-Datei wird aus der TUSTEP-Codierung der entsprechende Akzentbuchstabe.

Groß- und Kleinschreibung in Editor-Anweisungen

Bei Editor-Anweisungen werden Groß- und Kleinbuchstaben in den Ersatzzeichenfolgen einer Zeichenfolgen-Austausch-Tabelle auch ohne Umschaltzeichen unterschieden. Groß- und Kleinbuchstaben werden in der Datei in der Form eingesetzt, in der sie in der Anweisung angegeben sind. Das Codieren der Groß- und Kleinbuchstaben in der einzusetzenden Zeichenfolge mit den Umschaltzeichen < und > ist noch möglich aber nicht mehr nötig.

Die neue Regelung betrifft Austausch-Anweisung A und Suchparameter XVm und XZm in Datenbankanweisungen.

In den Suchzeichenfolgen ändert sich nichts. Wie bisher meint ein Buchstaben ohne Umschaltzeichen Groß- und Kleinbuchstaben, ein mit spitzer Klammer gekennzeichnet nur Groß- bzw. Kleinbuchstabe.

Beispiel:

```
a,,,/x/U/
```

Mit dieser Anweisung wird ein kleines oder großes x gesucht, das durch ein großes U ersetzt wird.

Neue Codierung der Zeichen in Parametern von Kommandofolgen

Mit dem Kommando PARAMETER läßt sich wählen, ob Parameter von Kommandofolgen in neuer Zeichencodierung oder noch in der alten angegeben sind.

```
#PARAMETER, MODUS=NEU
```

Groß- und Kleinbuchstaben werden in Parametern wie in den Editor-Anweisungen unterschieden. Von der neuen Regelung sind folgende Parameter betroffen:

- Parameterart II »Textteile« (z. B. ERG, KT)
- Parameterart IV »Textteile-Austausch-Tabelle« für die Ersatz-Textteile (z. B. TTT)
- Parameterart X »Zeichenfolgen-Austausch-Tabelle« für die Ersatz-Zeichenfolge (z. B. XX, XV, XTT).

Die Groß- und Kleinschreibung wird in diesen Zeichenfolgen und Textteilen in der angegebenen Form berücksichtigt. Das Codieren der Groß- und Kleinbuchstaben in der einzusetzenden Zeichenfolge mit den Umschaltzeichen < und > ist noch möglich aber nicht mehr nötig.

Für alle Parameter gilt: Zeichen, die mit Dach (^) codiert sind, können direkt oder mit ^ codiert angegeben werden. ^a meint das Zeichen ä in der Datei. Soll das Zeichen ^ selbst gesucht werden, ist im Parameter ^^ (= 1 Zeichen in der Datei) zu schreiben.

```
#PARAMETER, MODUS=ALT
```

Die alte Regelung für die Parameter soll beibehalten werden. Dies ist - für eine

Übergangszeit – die Voreinstellung. Groß- und Kleinbuchstaben lassen sich ausschließlich durch die Umschaltzeichen < und > unterscheiden; Groß- und Kleinschreibung allein wirkt sich nicht aus.

Für die mit Dach codierten Zeichen gilt:

- ^a im Parameter (1 Zeichen in der Parameterdatei) meint ä in der Datei.
- ^^a im Parameter (2 Zeichen in der Parameterdatei) meint ä in der Datei.
- ^^^^ im Parameter (2 Zeichen in der Parameterdatei) meint ^ (am Bildschirm ^^) in der Textdatei.

Um Kommandofolgen mit alten Parametern bezüglich der mit Dach codierten Zeichen auf

die neue Regelung umzustellen, kann man die Programmdatei mit dem Kommando

#UMWANDLE, alt, neu, MODUS=+1 umwandeln.

Die Umstellung der Groß- und Kleinschreibung muß von Hand erfolgen. Sie ist notwendig, falls in Parametern Groß- und Kleinbuchstaben willkürlich verwendet wurden.

Für die Übergangszeit empfiehlt es sich, zu Beginn von Kommandofolgen das Kommando #PAR, ALT oder #PAR, NEU entsprechend einzufügen und ggf. am Ende der Kommandofolge auf einen definierten Zustand zurückzuschalten.

Neue Zeilenlängen-Einstellungen für den Editor mit Ctrl+L

Bei den Längeneinstellungen für den Editor, die man mit Ctrl+L im Editor aufruft, um sie zu verändern, und die man dann mit der Anweisung o in TUSTEP.INI abspeichern kann, gibt es neue Möglichkeiten der Einstellung.

Mit den ersten drei Werten stellt man Höhe und Breite des Bildschirms ein, die dem Editor zur Verfügung gestellt werden. Dies ist z. B. sinnvoll bei Workstations und Graphikterminals mit variabler Fenstergröße, auf denen es andere Bildschirmformate gibt als das aus der Lochkartenzzeit stammende starre Format von 25 Zeilen mit 80 Zeichen Breite, das bisher auf den Bildschirmen üblich war und weiterhin die Voreinstellung ist. Auf dem PC ist eine Änderung dieser Werte z. Zt. nicht unterstützt.

Mit dem vierten und fünften Wert »Line width for data input« läßt sich die Zeilenbreite einstellen, die beim Eintragen von neuen Daten (nach Anweisung ee bzw. e, pos) zur Verfügung gestellt wird. Eine Angabe größer als der zu »Line width for displaying data« angegebene Wert ist nicht sinnvoll. Die Zeilenbreite für die Dateneingabe kann für Modus P und Modus T getrennt angegeben werden.

Die Verringerung der Zeilenbreite für die Dateneingabe ist z. B. sinnvoll, um nur so viel

Daten in eine Zeile schreiben zu können, daß beim anschließenden Zeigen mit Satznummern keine Folgezeilen entstehen.

Mit den letzten beiden Werten »Width of region for wrapping« läßt sich angeben, bis zu wieviel Zeichen vom rechten Rand aus nach einem Blank gesucht wird, um dort bei Sätzen, die nicht in eine Bildschirmzeile passen, eine Folgezeile zu beginnen. Wird im angegebenen Bereich kein Blank gefunden, wird am Zeilenende mitten im Wort eine Folgezeile begonnen. Ein Gravis am Zeilenende zeigt an, daß der Zeilenwechsel kein Blank bedeutet.

Wird 0 angegeben, wird grundsätzlich kein Wortumbruch durchgeführt, sondern erst am Zeilenende – auch mitten im Wort – umbrochen (dies ist der Zustand des alten Modus P).

Wird der Maximalwert angegeben, werden Wörter, die in eine Zeile passen, grundsätzlich nicht umbrochen (dies ist der Zustand des alten Modus T).

Die Voreinstellung 30 ist ein sinnvoller Kompromiß. Der Wortumbruch wird im allgemeinen durchgeführt, er unterbleibt aber bei langen Wörtern, bei denen er zu ungewöhnlichen Zeileneinteilungen führen würde.

Neue Datei-Information mit Anweisung D: »Letzte Schreibzugriffe«

Die Datei-Information im Editor mit Anweisung D enthält eine Meldung über die letzten Schreibzugriffe auf die Datei. Angegeben sind Datum und Uhrzeit des letzten und des vorletzten Schreibzugriffs. Je Tag wird jeweils nur der letzte Schreibzugriff gemerkt.

Bei Segmentdateien werden für jedes Segment die letzten beiden Schreibzugriffe gemerkt. Schreibzugriff auf ein Segment ist der Zeitpunkt, zu dem es gerettet wurde. Wird ein Segment in eine Datei geholt, erscheint in dieser Datei bei der Datei-Information das Datum der Schreibzugriffe für das geholte Segment.