

Starten von TUSTEP:

Vor dem Aufruf kann die Systemvariable TUSTEP\_INI definiert werden. Sie kann den Dateinamen oder (wenn es keine TUSTEP-Datei ist) Dateiname/code der Datei enthalten, die statt der Datei tustep.ini als INI-Datei benutzt werden soll. "code" ist UTF8 oder ISO8859/1. Die Datei muss in dem Verzeichnis stehen, in dem die Datei TUSTEP.INI erwartet wird.

INI-Datei:

Segmente EXIT und TERM mit Kommandomakros, die beim Unterbrechen bzw. Beenden einer Sitzung aufgerufen werden

Kommandos

#EDIERE:

fragt ggf. nach, ob Datei repariert werden soll

kann auch Dateien mit Sätzen bis zu 64000 Zeichen anzeigen (ändern nur in Sätzen bis 8000 Zeichen)

Änderungsdatum --> Last-Modified auf Systemebene

#INFORMIERE:

zeigt Daten der verwendeten TUSTEP-Version und der aktuellen Sitzung an, wenn ohne Spezifikationen aufgerufen

#LISTE:

MODUS = TITEL  
CODE = ... (für das Anzeigen von SDF-Dateien)  
MUSTER = (Dateien auflisten, die angegebenes Muster als Titel oder als Text enthalten, bzw. Sätze auflisten, die angegebenes Muster enthalten)  
OPTIONEN = (für Recherchiertabelle in MUSTER)

#KOPIERE:

10 Merktexthe; s17 / s18 / s19 gibt an, welcher gesetzt / ausgetauscht / geholt werden soll.

Initialisierung:

Vorbelegen der Merktexthe mit MTV

Programmteil 1:

ZFS (Zeichenfolgen im AT suchen, Zahl --> S8 für VSP; wird vor KEN / KSP ausgeführt)

Programmteil 3:

MIV / MDN (Zahlenwerte in MLD)

Programmteil 7:

AMT, EMT, (MT, )MT, MTI, XMT

Programmteil 9:

MTP (und MTH), MTA, MTE

#NUMMERIERE:

LNS (Schrittweite der lfd. Nr.)

#RAUFBEREITE:

RFF 0 | 1 | 2 (Zusammenfassen: nein / unter sich /  
auch mit "VON-BIS"-Paaren aus #RV)

#UMWANDLE:

SATZLAENGE = nnn (bzw. n-m):

Bei Blanks in tags wird möglichst nicht aufgeteilt

NUMMERIERUNG = -std- (Textmodus)  
= n Schrittweite für die Nummerierung

SEITENNUMMER = - Normalfall  
= ... Zeichenfolge, die zusammen mit der  
Seitennummer als eigener Satz in  
SDF-Datei ausgegeben bzw. beim  
Einlesen von SDF in Seitennummer  
für nachf. Sätze verwandelt wird.  
= <var> Die in <var> angegebene Zeichenfolge  
kann "#" enthalten, das durch  
Seitennummer ersetzt wird.

OPTIONEN = - Normalfall  
= ZEILEN Sätze mit gleicher Zeilennummer  
zusammenfassen  
= ZIRKUMFLEX | BOM | RLM

PROTOKOLL = + | - | -STD- | datei (für Hex-Codes)

#VERGLEICHE:

SZW szn wn Stellenangaben in Korrekturanweisungen  
szn = 0: gleiche Seiten- u. Zeilennummern weg  
= 1: immer ausgeben (nicht verkürzen)  
wn = 0: weglassen, falls ganze Zeilen  
= 1: immer ausgeben

Editor

Emulation von Clipboard (Windows-Zwischenablage) unter Unix

d!,... (keine Anfrage bei noch nicht geretteten Daten)

tt (Dateititel abfragen/einstellen/löschen)

gtt (gibt aktuellen Titeltext)  
 gtsl (letzte Anweisung tsl / Def. leere Tags)

y,name\*=feld1L'feld1M'feld1R:anzeigel,fld2:anzeige2,...  
 y,?name=feld1L'feld1M'feld1R:anzeigel,fld2:anzeige2,...  
 (für jedes Feld 3 Namen für linke, mittlere, rechte Maustaste;  
 Anzeige anders als Makroname)

INS:"text" (unabhängig von Replace- oder Insert-Modus)

REP:"text" " " " " " " "

SAVE\_MRK, REST\_MRK, EXCH\_MRK  
 (Zwischenspeicher <--> gemerkter Text)

X\_MRK:"zfatab"  
 (Zeichenfolgen im Zwischenspeicher austauschen)

SET\_MRK:"text", CLR\_MRK  
 (Zwischenspeicher füllen /leeren)

MRK\_TST:"zfstb"  
 (markierten Text prüfen)

TEST\_MRK:"zfstb"  
 (Zwischenspeicher prüfen)

MRK\_STR:"zfstb"  
 (an Cursorposition stehende und in zfstb  
 enthaltene Zeichenfolge markieren)

IF\_EMPTY:name, NOT\_EMPTY:name, DO\_EMPTY:name  
 (Editor-Datei leer / nicht leer)

IF\_ZERO:name, NOT\_ZERO:name, DO\_ZERO:name  
 (akt. Satzposition 0 / nicht 0)

IF\_MATCH:name, NO\_MATCH:name, DO\_MATCH:name  
 (Match-Bedingung; darin neu: Ergebnis von  
 NEXT\_REC, MRK\_STR, MRK\_TST, TEST\_MRK,  
 SET\_MATCH, CLR\_MATCH)

### Standardmakros:

```

#*IMPORT, QUELLE, ZIEL, MODUS=-STD-, LOESCHEN=-, LISTE=+,
  IGNORIEREN=-, ERSETZEN=-, SATZLAENGE=-STD-,
  KENNUNG=-STD-

  Import von XML-Dateien, die in WinWord erzeugt wurden

#*PARSER, QUELLE, ZIEL=-STD-, MODUS=-STD-, LOESCHEN=-,
  EXE=PARSER_EXE, ARG=PARSER_ARG, CLICK=M_MC

  TUSTEP-Dateien mit XML-Daten durch externen Parser
  prüfen lassen, Fehlermeldungen (im unteren Text-
  fenster) und zugehörige Textstelle (im oberen Text-
  fenster) im Editor anzeigen / zur Korrektur anbieten
  
```

Makros:

```

DEFINE/TUSTEP var = "..."/>
DEFINE/SYSTEM var = "..."/>
FETCH/TUSTEP var = "..."/>
FETCH/SYSTEM var = "..."/>
REMOVE/TUSTEP var
REMOVE/SYSTEM var

SHOW "...", var, ...
PRINT[/WARNING | /ERROR | /FILE | /DATA] "...", var, ...
ERROR "Zeichenfolge", <var>, ...
    Falls Inhalt der letzten Variablen "OK":
    keine Fehlermeldung, kein Fehlerflag
ERROR/RESET (löscht Fehlerflag)

WINDOW ... winvar[, aktfeld]
FIELD name: DUMMY
    wird nicht angezeigt, Cursor bleibt unsichtbar
FIELD name: INPUT [/SHIFT][/SCROLL] xx txt, num, pos
FIELD name: OUTPUT[/SHIFT][/SCROLL] xx txt, num, pos
FIELD name: OUTPUT[/WRAP] xx txt, num, pos
    /SCROLL: vertikal
    /WRAP: langen Text umbrechen
FIELD name: FLAGS /MULTIPLE/CLICK ...
FIELD name: SELECT/MULTIPLE/CLICK ...
FIELD name: FLAGS /SINGLE/MULTIPLE/CLICK ...
FIELD name: SELECT/SINGLE/MULTIPLE/CLICK ...
    /CLICK: Eingabe wird abgeschlossen, wenn ein
    Schalter mit der Maus angeklickt wird
FIELD name: typ[/ON | /OFF] xx-xx-xx-xx ...
    /ON: Feld wird angezeigt (Voreinstellung)
    /OFF: Feld wird nicht angezeigt
    xx-xx-xx-xx bis zu 4 Farben
    (1: Text, 2: Hervorhebung
    3: Text, 4: Hervorhebung für OFF-Felder)
MODIFY FIELD name ON | OFF
MODIFY FIELD name COLOR xx-xx-xx-xx
    während der Anzeige Status eines Feldes
    ändern
IF (MODIFIED) ...
    bei der zuletzt ausgeführten EDIT-Anweisung
    wurde Feldinhalt verändert
SET zeile = CURSOR_ROW (var, pos)
    Zeilennummer für den in Sternvariable var
    stehenden Text an der Stelle pos
SET spalte = CURSOR_COLUMN (var, pos)
    Zeichennummer (in der Zeile) ...
SET pos = CURSOR_POSITION (var, zeile[, spalte])
    Cursor-Position für Zeichen in Zeile zeile
    an Zeichennummer spalte in Sternvariable var
    stehenden Text
SET var = COLOR (zeile:spalte, xx)
    var = Hex-Code der ausgewählten Farbe
    oder "ENTER | CANCEL | Fn"

Prüfen von Zeichenfolgen; neue Operatoren
.ID. identisch (einschl. Groß/Kleinschreibung)
.NI. nicht identisch (einschl. Groß/Kleinschreibung)
.SW. beginnt mit

```

```

.SI. beginnt identisch mit
.EW. endet mit
.EI. endet identisch mit
      (bei .EQ. und .NE. sind Groß- u. Kleinbuchstaben
       gleichwertig)
.NC. enthält keine Zeichenfolge aus Suchtabelle(n)
.CT. enthält Zeichenfolge(en) aus Suchtabelle(n)
.MA. stimmt mit Zeichenfolge(en) aus Recherchiertabelle(n)
      überein
.NM. stimmt mit keiner Zeichenfolge aus Recherchier-
      tabelle(n) überein
      (direkt angeg. Recherchiertabelle(n): Option TEXT)

IF (#sn .GT. m) (Zahlenwert, nicht Ziffer / alphab.)

SET var1 = var2 = var3 ...

SET var = TIME_4 () (Uhrzeit xx:xx:xx.xxx)

LOOP/n n = (Grenze f. Meldung "Endlosschleife")

SET var = REDUCE (liste[, raus])
      reduziert aufeinanderfolgende gleiche Teil-
      zeichenfolgen bzw. Zeilen
SET var = COUNTER (var, level)
      var = Zähler (alpha oder numer.),
      level: für Dez-Klassifikationen
SET var = CONCAT (var, var1, var2, ...)
      statt Variablen auch "... "
SET var = JOIN (vart[,vare | "..."])
SET var = SPLIT (text, [|]stab[|], zeilenlaenge)
      | vor bzw. hinter Trenner: Trenner wird vor bzw.
      hinter Textteil übernommen
SET var = INSERT_INDEX (liste, var)
      Index für's alphabetische Einfügen von
      (var) in liste
SET var = REMOVE (liste, auswahl[, raus])
      raus = nach Aufruf: entfernte Zeichenfolgen
SET var = EXTRACT (var, [|]apos[|], [|]epos[|], num, anz, erg)
SET var = ELIMINATE (var, [|]apos[|], [|]epos[|], num, anz, erg)
SET var = IDENTIFY (var, stab, [|]apos[|], [|]epos[|], npos)
SET var = SEARCH (var, stab, [|]apos[|], [|]epos[|], n, npos)
SET var = COUNT (var, stab, [|]apos[|], [|]epos[|])
SET var = VERIFY (var, stab, [|]apos[|], [|]epos[|])
SET var = EXCHANGE (var, xtab, [|]apos[|], [|]epos[|],
      xnum, xanz, num, anz, xtabx)
set var = ASSIGN (var, abk, [|]stab[|], var1, var2, ...)
      | vor bzw. hinter Trenner: Trenner wird vor bzw.
      hinter Textteil übernommen
SET var = MIXED_SORT (var)
      Ziffernfolgen werden dem Zahlenwert nach sortiert
SET var = DECODE (var, modus)
      neu zu modus: ANSI | ENTITIES |
      ANSI/ENTITIES | ISO8859/ENTITIES

SET var = CHECK (name, art, typ)
      typ = TUSTEP | SYSTEM
SET var = PROJECT_NAME (name)
SET var = FILE_NAME (name)
SET var = FULL_NAME (typ, name)

```

```

        typ = TUSTEP | SYSTEM
SET var = VOLUME (name)
        generell:
        SET var = function "dateiname" -->
        SET var = function (name | "dateiname")
SET var = EDIT_INFO (name, datei, segment)
        Datei- und Segment-Name wie bei "d" in #ED
        var = UNKNOWN | MODIFIED | OK

SET var = FILE (name[, auswahl[, anzahl]])
        auswahl: - | "segmentname" | variablenname |
                ? | + | nzeich
        anzahl:  0 | +n | -n | +var | -var

ACCESS d: WRITE[/ERASE] ...
ACCESS d: ... /RECORDS | /LINES | /PAGES ...
        /LINES und /PAGES nur bei TUSTEP-Dateien
ACCESS d: ... /STREAM [/RECORDS] ...
ACCESS d: ... [/ISO | /UTF] ...

READ | FIND | COUNT /EXIT ...
        Option /EXIT erzeugt EXIT (aus einem LOOP)
        wenn keine Texteinheit mehr vorh.
IF (EXIST)
        READ- bzw. CHECK-Anweisung erfolgreich
IF (SOR)
        mit READ zuletzt gelesene Portion war Satzanfang
IF (EOR)
        mit READ zuletzt gelesene Portion war Satzende
ENDACCESS/PRINT daten
        zusätzl. Angaben zu den ein-/ausgegebenen Daten

FILE[/...] "datei" = var | "...
        Optionen: /PROGRAM, /ERASE
FILE/TITLE "datei" = var | "...
        Dateititel

TRACE +
        ohne Var-Name: Protokollierung f. alle Variablen

```

Satz:

Behobene Fehler: Abstand der unteren Seitennummer

Neue Obergrenzen:

Wortlänge: 480 Zeichen.  
Makros: 3000 statt bisher 1000 Makros.  
Zeichen in Makroauflösungen: 32000 statt 16000.  
Satzbreite und Satzhöhe: 2400 statt 1200 Punkt.  
Grafiken mit mehr als 10\*\*6 EPS-Zeilen.

Schriften:

Laufweite ändern mit Parameter SLW (parallel zu  
Angaben in SCH)

Schriftgröße (Höhe und Breite) modifizieren  
mit Parameter SGM (parallel zu Angaben in SCH)

Linksläufiger Satz (derzeit nur für Hebräisch fertig):

HBU 0 (dreht hebräische Texte bei SATZ bzw. \*PSAUS selbst um); Umschaltung auf nicht-hebräischen Font (z.B. {{...{{{) sowie #N+...#N- unterbricht Umdrehen.

Silbentrennungen:

SIL, 7. Wert negativ =  
führt nach der angegebenen Zahl von  
aufeinanderfolgenden Trennungen keine  
weitere aus.  
Zähler wird bei Seitenwechsel auf 0 gesetzt

Makros: TXB Textbeginn; was davor steht, ist Vorspann und  
kommt (ohne Satzausgabe) auf Seite 0

MAZ -2 Tags werden nicht aufgelöst  
-1 Nur Tags, die Zeilenwechsel zwischen erstem  
und letztem Zeichen der Auflösung bewirken,  
werden aufgelöst; Tag-Kennung wird als  
(XML-)Kommentar davor bzw. dahinter  
geschrieben.

MAC <!-- > wird ggf. ergänzt

Seitenumbruch:

&n&&{ vs. &&n&&{{ (keine Seite mit leerem Text-  
teil bei übergelaufenen Fußnoten und Apparaten)

Einzüge, Einrückungen:

&=(Pm+nnn), &=(Pm-nnn), &=- (Pm+nnn), &=- (Pm-nnn)

Positionieren:

&!P(Hn) = vertikal Mitte zwischen aktueller und  
(davor liegender) mit &!M(Vn) gemerkter Stelle.

Auszeichnungen

Schachtelung möglich:

#/+#f+#g+ gibt griechisch (halb)fett kursiv  
falls ein solcher Font in SCH angegeben ist

\{ und \#?- schaltet um eine Auszeichnungsstufe  
zurück

\\ zwischen Blanks verhindert Ergänzen von  
Auszeichnungen-Ende am Ende des nächsten  
Abschnitts

Farben:

HKS-Farben werden angegeben mit der Anweisung  
&!c(Hnm)  
n = HKS-Farbnummer, m = Material (N = Natur,  
K = Kunstdruck, E = Endlospapier, Z=Zeitung)

Zeichenvorrat:

&!- horizontal spiegeln, Unterkante = Grundlinie  
&!= horizontal spiegeln, Lage wie Original  
&!/ um 180 Grad drehen, Unterkante = Grundlinie  
&!\ um 180 Grad drehen, Lage wie Original

Linien: &!\(a,b,n,p,l,w) gibt bei negativem p statt  
ausgefülltem Dreieck eine offene Pfeilspitze

Arabisch: "--" für kashida (auch in \*CASH berücksichtigt)

### Satzmakros

\*FUNO: %0%0%0 und @f@f beginnen Fußnotenzählung neu  
mit 0 statt 1  
Satzlänge in der ZIEL-Datei bei aufgeteilten  
Abschnitten bis zu 4000 Zeichen

\*PROT2QU, QUELLE, ZIEL, MODUS, LOESCHEN=-, PARAMETER=-,  
PROTOKOLL=+, FUSSNOTEN, SATZPAR

Stellt aus Satzprotokoll Quelldateien für  
Text und Fußnoten sowie die Parameter für  
einen nachfolgenden Satzlauf her.

\*PS4A4, \*PS2A4:

Neue Angaben zu OPTIONEN: D\_LANG und D\_KURZ

\*PSAUS: In die zweite zu DATEI angegebene Datei werden  
neben der PostScript-Fassung der Satzausgabe die  
Meldungen der PROTOKOLL-Datei des Satzlaufs am  
rechten Rand mit ausgegeben. Dazu müssen beide  
PROTOKOLL-Dateien des Satzlaufs zur (neuen, 23.)  
Spezifikation SATZPROT angegeben werden.

\*PSAUSWAHL, QUELLE, ZIEL, LOESCHEN=-, NUMMERN

Stellt eine PostScript-Datei her, die neben dem  
Vorspann nur die ausgewählten Seiten enthält.

\*TAGS, MODUS=k

reduziert fully qualified tags (wie in der zweiten  
Ziel-Datei) wieder zu normalen tags

\*XMLZIEL, QUELLE, ZIEL, MODUS=-STD-, LOESCHEN=-,  
KORREKTUREN=-, SATZQUELLE=-

Stellt wegen "MAZ -1" aufgelöste XML-Tags wieder  
her, prüft die Übereinstimmung mit der zu  
SATZQUELLE angegebenen Datei.

\*XMLATTSOR, QUELLE, ZIEL, LOESCHEN=-, REIHENFOLGE=ALPHA

Sortiert die Attribute innerhalb von XML-Tags  
in die angegebene Reihenfolge

### Dokumentation

aktuelles TUSTEP-Handbuch als verlinkte PDF-Datei im Netz  
und auf Installations-CD