

**Neuerungen in TUSTEP Version 2013**  
gegenüber Version 2012 (Übersicht)  
Stand: 13.03.2013

Band-Dateien

{25}

Band-Dateien können jetzt auch mit dem Kommando #DATEI eingerichtet, mit #RETTE beschrieben, mit #HOLE gelesen und mit #LOESCHE gelöscht werden.  
Bei den Kommandos #MBA, #MBE, #MBI, #MBK, #MBL, #MBT ist die Voreinstellung zu GERAET jetzt "-".

INI-Datei

{67}

Enthält die INI-Datei ein Segment mit dem Namen EDIT, so darf es nur Editor-Definitionen enthalten; sie werden beim ersten Aufruf des Editors in einer Sitzung automatisch interpretiert; außerdem werden sie interpretiert, wenn im Aufruf des Editors die Spezifikation DEFINITIONEN=+ angegeben wird.

#DATEI

{104}

TYP = TAPE                    Die einzurichtende Datei  
                                 ist eine Banddatei.

#DEFINIERE

DATEINAMEN = name:-    Definition des Dateinamens name    {109}  
                                 löschen.

#EDIERE

{122}

DEFINITIONEN = -STD-  
                                 Alle Definitionen löschen und die  
                                 voreingestellten Definitionen  
                                 wieder übernehmen.  
                                 = +    Alle Definitionen löschen und die  
                                 voreingestellten sowie die im  
                                 Segment EDIT der INI-Datei  
                                 stehenden Definitionen wieder  
                                 übernehmen.

Aufruf des TUSTEP-Editors im Windows-Explorer

Mit der rechten Maustaste auf den Dateinamen klicken, "Öffnen mit dem TUSTEP-Editor" auswählen. Falls zuvor mehrere Dateien markiert wurden, wird sofort für jede Datei eine eigene Sitzung begonnen und die jeweilige Datei angezeigt.

Mit der rechten Maustaste auf den Dateinamen klicken, "Senden an" auswählen, dann "TUSTEP-Editor" auswählen. Falls zuvor mehrere Dateien markiert wurden, wird trotzdem nur eine Sitzung begonnen und die erste Datei angezeigt; nach dem Beenden des Editors wird jeweils die nächste Datei angezeigt.



MUSTER = ... Falls Namen der in einer Band-Datei stehenden Dateien aufgelistet werden, nur die Dateinamen in die Liste aufnehmen, die ein angegebenes Muster enthalten.

#LOESCHE {152}

DATEN = name Ist name der Name einer Band-Datei, so werden die darin enthaltenen Daten gelöscht; die Band-Datei wird neu gelabelt.

DATEI = name Ist name der Name einer Band-Datei, so wird diese gelöscht.

#MBAUSGABE, #MBEINGABE, #MBINFORMIERE, #MBLABEL, #MBTEST {157}

GERAET = - \* Band-Datei ist angemeldet.

#MBKOPIERE {163}

QGERAET = - \* QUELL-Band-Datei ist angemeldet.

ZGERAET = - \* ZIEL-Band-Datei ist angemeldet.

#RETTE {185}

MODUS = -STD- \* Falls ZIEL-Datei eine Band-Datei ist: Daten der QUELL-Datei als Datei mit dem zur Spezifikation QUELLE angegebenen Namen ans Ende der Band-Datei kopieren.

= name Falls ZIEL-Datei eine Band-Datei ist: Daten der QUELL-Datei als Datei mit dem angegebenen Namen ans Ende der Band-Datei kopieren.

= n Falls ZIEL-Datei eine Band-Datei ist: Daten der QUELL-Datei als Datei mit dem zur Spezifikation QUELLE angegebenen Namen als n-te Datei in die Band-Datei kopieren.

= n:name Falls ZIEL-Datei eine Band-Datei ist: Daten der QUELL-Datei als Datei mit dem angegebenen Namen als n-te Datei in die Band-Datei kopieren.

#UMWANDLE {207}

MODUS = TQ Die Textdaten aus der QUELL-Datei in Quoted-Printable-Kodierung in die ZIEL-Datei schreiben.

= QT Die Daten, die in Quoted-Printable-Kodierung in der QUELL-Datei stehen, als Textdaten in die ZIEL-Datei schreiben.

Editor

Steuerbefehle: Cursor

LAST_LINE	Skip to last line of text	{ 307 }
	<ul style="list-style-type: none"> <li>- bei Dateneingabe (z.B. nach der Anweisung ee): Cursor springt ans Ende des Textes in der letzten Zeile mit Text. Falls kein Text vorhanden ist, springt der Cursor an den Anfang der oberste Zeile.</li> <li>- sonst: (noch) nicht definiert.</li> </ul>	

Makros

Zugriff auf System-Variablen: { 368 }

DEFINE/SYSTEM	System-Variablen können jetzt auch unter Linux mit dieser Makro-Anweisung definiert werden.
---------------	---------------------------------------------------------------------------------------------

Tabellen prüfen:

<pre>check s_table s_err + s_mssg = *   (in s_err steht lfd. Nr. des fehlerhaften    Zeichens)   (in s_mssg steht der Text der Fehlermeldung)</pre>	{ 384 }
-----------------------------------------------------------------------------------------------------------------------------------------------------	---------

Zeichenvorrat

Mit #(name) codierte Sonderzeichen

#(TM)	™ Trademark	{ 681 }
-------	-------------	---------

##\*SATZ

Druckfarben:

Im Satz und Export können 22 Farbtöne entsprechend S. 18 der Beschreibng des Standardmakros EXPORT verwendet werden.

Im Modus Edieren können in einer Datei nur die Fußnoten angezeigt werden.

Näheres siehe ##\*SATZ,,NEUERUNGEN; ausführliche Beschreibung mit ##\*ZEBE,SATZMAKRO

Parametergesteuerte Programme

In allen parametergesteuerten Programmen gibt es einen neuen Parameter:

PAR	<>	Für die Interpretation der folgenden Parameter gilt der Parameter-Modus <>
	{ }	Für die Interpretation der folgenden Parameter gilt der Parameter-Modus { }

#SATZ

## Parameter

## Schriften

SCH	Für sehr steil stehende Kursivschriften kann in der Klammer hinter der Schriftnummer (\\) statt (\) angegeben werden, um die horizontale Verschiebung von übergesetzten Buchstaben und von Akzenten unter dem Buchstaben anzupassen.	{1064}
-----	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------

## Satzspiegel

HOE	NAUSGL	Hinter dem zweiten Wert (NAUSGL = Angaben zum Spaltenhöhenausgleich) kann (ohne Spatium und ohne die Gänsefüßchen) ";2" oder ";8" angegeben werden. Damit wird die Schrittweite (1/2 bzw. 1/8 Punkt) festgelegt, mit der ggf. der Durchschuss verändert wird, wenn die variablen Vorschübe nicht für den Ausgleich der Spaltenhöhe ausreichen. Wird nichts angegeben, so ist dies gleichbedeutend mit ";2" (= bisherige Regelung).	{1073}
-----	--------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------

## Zeilenumbruch und Silbentrennung

AUS	LAUSGZ = n	Auf zu kurze Ausgangszeilen von Abschnitten kann mit einer Fehlermeldung im Protokoll hingewiesen werden. Dazu gibt man als 8. Zahlenwert (LAUSGZ) an, wie viele Prozent der augenblicklich geltenden Zeilenlänge eine Ausgangszeile einnehmen muss. Wird diese Länge unterschritten, so wird ins Protokoll die Fehlermeldung "Kurze Ausgangszeile" ausgegeben.	{1079}
-----	------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------

## Marginalien

ZLN	Als 7. Wert kann angegeben werden:	{1082}
IZNPR	0 = keine Angabe: keine Meldung über Zeilennummern ins Protokoll ausgeben. 1 = Meldung über die am Satzspiegelrand ausgegebenen Zeilennummern ins Protokoll. 2 = Meldung über die Zeilennummern	

ins Protokoll, auch wenn sie nicht ausgegeben werden.

- 3 = wie 2, jedoch auch für die Zeilen, für die die Ausgabe der Zeilennummern durch &!r- bzw. &!r. unterdrückt wird.

#### Umdefinition von Zeichen und Dikten

- DIC Zur Änderung des voreingestellten Dichtenwertes {1102}  
von 1000 für ™ ist beim Parameter DIC zu PA statt der Primäradresse die Zeichenfolge "tm" oder "TM" (ohne Anführungszeichen) anzugeben.
- SP Sperrung am Wortanfang und am Wortende <0,1> {1103}
- ISPWA 0 = Bei "Sperrung Anfang" nach einem Spatium wird erst nach dem ersten Zeichen dahinter mit der Sperrung begonnen.  
1 = Bei "Sperrung Anfang" nach einem Spatium wird sofort mit der Sperrung begonnen.
- ISPWE 0 = Bei "Sperrung-Ende" unmittelbar vor einem Spatium endet die Sperrung schon vor dem Spatium.  
1 = Bei "Sperrung-Ende" unmittelbar vor einem Spatium wird auch hinter dem "Sperrung Ende" noch ein entsprechender Zwischenraum erzeugt (analog zu "Sperrung Ende" innerhalb eines Wortes).

#### Steueranweisungen

##### Verändern der Satzbreite {1118}

- &!S(1,mmm;n) wie &!S(1,mmm), jedoch gilt die Änderung der Spaltenbreite für max. n Zeilen, falls sie nicht vorher durch &!s{ beendet wird.
- &!S(1,Pnn;n) Die Änderung gilt für max. n Zeilen.
- &!S(1,Pnn+m;n) Die Änderung gilt für max. n Zeilen.
- &!S(1,Pnn-m;n) Die Änderung gilt für max. n Zeilen.

##### Ändern von Spaltenhöhe und Durchschuss {1132}

- &!D/(nn) Verringerter Durchschuss von nn/8 Punkt (wie &!D/n, jedoch auch mehrstellige Angabe möglich).
- &!D\ (nn) Verringerter Durchschuss von nn/8 Punkt (wie &!D\n, jedoch auch mehrstellige Angabe möglich).
- &!D\ (nn!) Verringerter Durchschuss von nn/8 Punkt nach dieser und allen folgenden Zeilen bis &!D/0.

&!D=n	Zusätzlicher Durchschuss von n/8 Punkt nach (nicht: vor) der Zeile, in der diese Anweisung steht.
&!D=(nn)	Zusätzlicher Durchschuss von n/8 Punkt nach (nicht: vor) der Zeile, in der diese Anweisung steht.
&!D=(nn!)	Zusätzlicher Durchschuss von n/8 Punkt nach dieser und allen folgenden Zeilen bis &!D/0.

Der in Achtelpunkt-Schritten veränderte Durchschuss wird bei der Berechnung der Seitenhöhe nicht mit berücksichtigt, jedoch vor der Einsteuerung der Fußnoten bzw. vor der unten stehenden Seitenzahl wieder ausgeglichen.

#### Vertikale Merkstellen

{1152}

Wird auf einer Seite / in einer Spalte mit &!P(Vn) oder &!P(Wn) oder &!P(Hn) auf eine vertikale Position Bezug genommen, die auf dieser Seite / in dieser Spalte noch nicht definiert ist, so wird - zusätzlich zur Fehlermeldung in der Protokolldatei - für die Satzausgabe eine Seite mit einer passenden Fehlermeldung erzeugt. Beim Versuch, die zugehörige Satzausgabe mit \*psaus in eine PostScript-Datei zu schreiben, wird in die .ps-Datei nur die Seite mit dieser Fehlermeldung übernommen.

#### Zeilenzähler; Marginalien

{1154}

&!R(nn)	<p>schaltet von seitenweiser Nummerierung der Zeilen am Satzspiegel-Rand auf fortlaufende Nummerierung um; für die aktuelle Zeile wird die Nummer nn festgesetzt.</p> <p>Ist schon auf fortlaufende Nummerierung umgeschaltet, so wird nur die für die aktuelle Zeile geltende Nummer neu festgesetzt.</p> <p>Es werden z.Zt. nur die letzten 4 Ziffern der Zeilennummern ausgegeben.</p>
&!R(U)	unterbricht bei fortlaufender Nummerierung die Zählung der Zeilen.
&!R(W)	nimmt bei fortlaufender Nummerierung die Zählung der Zeilen wieder auf.
&!R(-1)	(oder anderer negativer Wert): schaltet von fortlaufender Nummerierung auf seitenweise Nummerierung um.

Sperrung {1157}

- #s\*            wie #s+, jedoch wird der erste zusätzliche Abstand auch bei Sperrung in der Wortmitte erst **h i n t e r** dem nächsten Zeichen (bei #s+ schon davor) eingefügt.
- #s=            wie #s-, jedoch wird an der Stelle, an der die Anweisung steht, kein zusätzlicher Abstand mehr eingefügt (z.B. vor Punkt oder Anführungszeichen nach einem gesperrten Wort).

## Hardware-nahe Anweisungen

- &!(+125=)    wie &!(+125), aber nur auf linken Seiten    {1206}  
bzw. Spalten mit gerader Spaltennummer.
- &!(+125/)    wie &!(+125), aber nur auf rechten Seiten  
bzw. Spalten mit ungerader Spaltennummer.
- &!(++125=)   wie &!(++125), aber nur auf linken Seiten  
bzw. Spalten mit gerader Spaltennummer.
- &!(++125/)   wie &!(++125), aber nur auf rechten Seiten  
bzw. Spalten mit ungerader Spaltennummer.
- &!(-125=)    wie &!(-125), aber nur auf linken Seiten  
bzw. Spalten mit gerader Spaltennummer.
- &!(-125/)    wie &!(-125), aber nur auf rechten Seiten  
bzw. Spalten mit ungerader Spaltennummer.
- &!(--125=)   wie &!(--125), aber nur auf linken Seiten  
bzw. Spalten mit gerader Spaltennummer.
- &!(--125/)   wie &!(--125), aber nur auf rechten Seiten  
bzw. Spalten mit ungerader Spaltennummer.

Makros für die Satzumgebung

## #\*GRF2EPS

Stellt aus einer mit \*GRAFIK für TUSTEP aufbereiteten Grafik eine EPS-Datei her, die mit GhostView betrachtet werden kann.

Das Makro hat folgende Spezifikationen:

- QUELLE    = name    Name der Grafikdatei, in der die umzuwandelnde Grafik steht. Wurden die Grafikdaten beim Aufruf von \*GRAFIK auf zwei Zieldateien aufgeteilt, so kann hier der Name der ersten oder der zweiten dort angegebenen Zieldatei angegeben werden.



NUMMER	= n	Nummer der Grafik, die umgewandelt werden soll.
ZIEL	= name	Name der EPS-Datei, die erzeugt werden soll.
LOESCHEN	= +	Ist die ZIEL-Datei nicht leer, so muss LOESCHEN = + angegeben werden.
ZEIGEN	= nein	(Voreinstellung)
	= ja	erzeugte EPS-Datei anzeigen.

#### ##\*PRECOMPOSED

Mit dem Makro \*PRECOMPOSED werden die mit \*PSAUS erzeugten Akzentbuchstaben in precomposed characters umgewandelt, soweit diese im ISOLatin1 Encoding Vector von Adobe vorgesehen sind.

Das Makro hat folgende Spezifikationen:

QUELLE	= datei	Name einer von SATZ und *PSAUS erzeugten PostScript-Datei mit fliegenden Akzenten.
ZIEL	= datei	Name einer FDF-Datei (zweckmäßigerweise mit der extension .ps), in die die umgewandelten Postscript-Daten geschrieben werden.
LOESCHEN	= +	Ist die ZIEL-Datei nicht leer, so muss LOESCHEN = + angegeben werden.

#### Hinweis:

Das TUSTEP-Satzprogramm arbeitet mit fliegenden Akzenten. Wörter, die Akzentbuchstaben enthalten, werden deshalb bei der Suche in PDF-Dateien nicht gefunden, wenn diese mit SATZ und \*PSAUS vorbereitet wurden. Dies kann vermieden werden, wenn die PS-Datei vor dem Umwandeln mit dem Makro \*PRECOMPOSED in eine PDF-Datei umkopiert wird.

Es fehlt noch die Prüfung, ob in den verwendeten Fonts die entsprechenden Zeichen vorhanden sind. Das Makro sollte deshalb nur verwendet werden, wenn dies gewährleistet ist.

#### ##\*SCHNEIDEN

Mit dem Makro SCHNEIDEN werden Schneidemarken in die Daten einer PS-Datei eingefügt. Zu diesem Zweck werden die Daten um den dafür benötigten Raum nach oben und nach rechts verschoben.

Das Makro hat folgende Spezifikationen:

QUELLE	= datei	Name einer von SATZ und *PSAUS erzeugten PostScript-Datei mit fliegenden Akzenten
ZIEL	= datei	Name einer FDF-Datei (zweckmäßigerweise mit der extension .ps), in die die um die

Schneidemarken ergänzten Postscript-Daten geschrieben werden.

LOESCHEN = + Ist die ZIEL-Datei nicht leer, so muss LOESCHEN = + angegeben werden.

UEBERSTAND = 0 (Voreinstellung) Die in der BoundingBox in der Quelldatei genannten Maße entsprechen dem Format der beschnittenen Seite.

= mm Die Schneidemarken sollen auf allen vier Seiten um mm Millimeter innerhalb der in der QUELL-Datei angegebenen BoundingBox gesetzt werden.

#### Hinweise:

Die Voreinstellung "0" zu UEBERSTAND geht davon aus, dass die in der QUELL-Datei angegebene BoundingBox dem beschnittenen Seitenformat entspricht und evtl. bis zur Beschnittkante reichende Teile (z.B. über den Satzspiegel hinausragende Abbildungen oder Griffregister) ausreichend weit über diese BoundingBox hinausragen.

Umfasst die BoundingBox in der Quelldatei schon die über das beschnittene Seitenformat hinausragenden Teile, so wird zu UEBERSTAND angegeben, um wie viele mm die Schneidemarken innerhalb der BoundingBox gesetzt werden sollen.

Sollen die Schneidemarken außerhalb der in der QUELL-Datei enthaltenen BoundingBox gesetzt werden, so kann dies durch negatives mm angegeben werden.

#### #\*PSAUS

{1266}

SEITEN = 0(999990) Verhindert Abbruch wegen fehlender vertikaler Merkstellen auf der Seite, auf der sie angesprochen werden (nur zum Testen, nicht für Produktion geeignet).

= 2-4'999990 Werden einzelne Seiten ausgewählt, so wird der Abbruch verhindert, wenn als letzte die Seite 999990 angegeben wird.

SATZPROT = p1'p2'fpl'!

Die Zugehörigkeit der Dateien zum selben Satzlauf wird anhand des letzten Schreibzugriffs überprüft: die beiden ersten Dateien und die Quelldatei (=Satz-Ausgabe) müssen gleichzeitig erstellt sein, die dritte Datei darf höchstens eine Woche älter sein. Soll die letztere Prüfung unterbleiben, so kann an vierter Stelle ein "!" angegeben werden. "!!" als vierte Angabe schaltet alle Prüfungen aus.

#\*PSMONT

{1289}

MODUS = page Zusammenmontieren von Seiten mit der  
selben Seitennummer (z.B. zweispaltig  
gesetzter Text, einspaltige Apparate  
oder umgekehrt).

\* \* \* \* \*

Corrigenda zum gedruckten Handbuch 2008

Seite 205, letzte Zeile:

Statt "ggf. nach den Daten" muss es heißen:  
"ggf. vor den Daten"

Seite 336, Zeile 20:

Statt "werden die Modi VARIABLE und DATA (s.o.)"  
muss es heißen:  
"werden die Modi VARIABLE und STATEMENT (s.o.)"

Seite 425, Zeile 9:

Statt "\$\$ SET neu = EXTRACT (alt, ....  
muss es heißen:  
"\$\$ SET neu = EXTRACT (hlf, ....

Seite 464, Seitenmitte, erste Zeile nach "Beispiel:"

Statt |%<| muss es heißen: |%<<|  
also (die ganze Zeile):  
|<%|0|%/|1|%\|2|%<<|3|%:|4|

Seite 1045, 10 Zeile:

Statt "Wie \$\$-nnn\$\${" muss es heißen:  
"Wie \$\$nnn\$\${"

**Parameter-Modi seit TUSTEP Version 2012**

Seit der TUSTEP-Version 2012 steht zur Kennzeichnung von Zeichen- und Stringgruppen, Häufigkeitsangaben, Verweisen usw. in Anweisungen bzw. Parametern zur Mustererkennung eine weitere (dritte) Konvention zur Verfügung, bei der geschweifte Klammern statt wie bisher spitze Klammern verwendet werden.

Sie war notwendig geworden, um Parameter oder Kommandomakros, die solche Angaben enthalten, auch mit einem XML-Editor schreiben oder ändern zu können.

Sie empfiehlt sich unabhängig davon vor allem für neue Projekte, weil die entsprechenden Codierungen leichter zu merken und leichter zu lesen sind.

Nach welcher Konvention die Parameter und die Editor-Anweisungen interpretiert werden, kann mit dem Kommando #PARAMETER eingestellt oder innerhalb von Parametern mit dem Parameter par festgelegt werden. In Kommandomakros wird der Interpretationsmodus mit der Anweisung \$\$MODE eingestellt.

In Kommandofolgen / Editor:

```
#parameter, modus=<>
```

(gleichbedeutend mit #parameter,modus=neu):

Verwendung von Spitzklammern zur Kennzeichnung von Zeichen- und Stringgruppen, Häufigkeitsangaben, Verweisen usw.

```
#parameter, modus={}
```

Verwendung von geschweiften Klammern zur Kennzeichnung von Zeichen- und Stringgruppen, Häufigkeitsangaben, Verweisen usw.

Innerhalb von Parametern:

```
par      <>  Nachfolgende Parameter nach der
           <>-Parameter-Konvention interpretieren.
```

```
par      {}  Nachfolgende Parameter nach der
           {}-Parameter-Konvention interpretieren.
```

Die Einstellung mit dem Parameter PAR hat Vorrang vor der mit dem Kommando #PARAMETER gewählten Einstellung; sie gilt jeweils nur für die nachfolgenden Parameter bis zum nächsten Parameter PAR bzw. bis zum Ende der Parameter des jeweiligen Programms.

In Kommandomakros:

```
$$ MODE <>  Verwendung von spitzen Klammern zur Kennzeichnung
           von Zeichen- und Stringgruppen, Häufigkeitsangaben,
           Verweisen usw.
```

```
$$ MODE {}  Verwendung von geschweiften Klammern zur
           Kennzeichnung von Zeichen- und Stringgruppen,
           Häufigkeitsangaben, Verweisen usw.
```

Aus Kompatibilitätsgründen ist <> voreingestellt.

Mit dieser Übersicht soll eine kompakte Umstellungshilfe von #parameter,modus=<> auf #parameter,modus={ } bzw. von \$\$ MODE <> auf \$\$MODE { } geboten werden. Sie soll die ausführliche Beschreibung im Handbuch ergänzen.

Modus	Modus
<>	{ }

#### Vordefinierte Zeichengruppen

<%	?	ein beliebiges Zeichen
>%	{!}	ASCII-Zeichen
	{;}	TUSTEP-Zeichen außer ASCII-Zeichen
<@	{@}	Zeichen außer Buchstaben und Ziffern
>@	{%}	Zeichen hinter % zur Akzentcodierung
>*	{\a}	Kleinbuchstaben
<*	{\A}	Großbuchstaben
</	{&a}	Kleinbuchstaben & Großbuchstaben
	{\0}	normale Ziffern
>/	{&0}	normale Ziffern & hochgestellte Ziffern
><<>%	*	null bis beliebig viele beliebige Zeichen

#### Zeichen- und Stringgruppen

- Definition im Editor:

```
>[vo]=aeiou C:vo=aeiou
                Z:vo=aeiou
<[di]=|ei| S:di=|ei|
```

- Definition in Parametersätzen (Sp. 1-3):

>xy	>xy	Definition einer Zeichengruppe
>lz		Definition einer Zeichengruppe
<lz		Definition einer Zeichengruppe
<xy	<xy	Definition einer Stringgruppe
>ls		Definition einer Stringgruppe
<ls		Definition einer Stringgruppe
><	{-}	nachfolgende Zeichen aus der Gruppe entfernen
<>	{+}	nachfolgende Zeichen in die Gruppe aufnehmen

- Bezug auf eine Zeichen- oder Stringgruppe:

	[...]	lokale Zeichengruppe, z.B. m[ae][iy]er
>[xy]	{Z:xy}	selbstdefinierte Zeichengruppe xy
	{C:xy}	alternative Schreibweise für {Z:xy}
<[xy]	{S:xy}	selbstdefinierte Stringgruppe xy
>l		selbstdefinierte Zeichen- oder Stringgruppe
<l		selbstdefinierte Zeichen- oder Stringgruppe

#### Häufigkeitsbedingungen in Suchzeichenfolgen

><n	{n}	genau n Elemente
><n<>m	{n-m}	n bis m Elemente, möglichst wenige
<>m><n	{n--m}	n bis m Elemente, möglichst viele

><0	{0}	0 oder 1 Element = {0-1}
<>0	{00}	1 bis unendlich viele Elemente = {1-0}

## Zahlenwertbedingungen in Suchzeichenfolgen

	{#}	Zahl mit beliebigem Wert
>={n-n}	{#n}	Zahl mit dem Wert n
<{n-n}	{!n}	Zahl mit einem Wert ungleich n
>={n-m}	{#n-m}	Zahl mit einem Wert von n bis m
<{n-m}	{!n-m}	Zahl mit einem Wert kleiner n oder größer m

## Verweise in Suchzeichenfolgen

>=nn	{+n=}	n-tes Element von links gezählt a != A
<=nn	{-n=}	n-tes Element von rechts gezählt a != A
>:nn	{+n.}	n-tes Element von links gezählt a == A
<:nn	{-n.}	n-tes Element von rechts gezählt a == A

## Elementbereiche in Suchzeichenfolgen

	{ }	Begrenzungszeichen zwischen Elementbereichen
--	-----	----------------------------------------------

## Verweise in Ersatzzeichenfolgen

>=nn	{+n=}	n-tes Element von links gezählt
<=nn	{-n=}	n-tes Element von rechts gezählt
>=00	{+0=}	alle Elemente der Kernzeichenfolge
<=00	{-0=}	alle Elemente der Kernzeichenfolge
>=(n-m)	{+n-m=}	n-tes bis m-tes Element von links gezählt
>=(n-0)	{+n-0=}	n-tes bis letztes Element von links gezählt
<=(n-m)	{-n-m=}	n-tes bis m-tes Element von rechts gezählt
<=(0-m)	{-0-m=}	erstes bis m-tes Element von rechts gezählt
	{=n=}	alle Elemente des n-ten Elementbereichs
	{=0=}	alle Elemente der Kernzeichenfolge
	{=n-m=}	alle Elemente des n-ten bis m-ten Elementbereichs
>+nn etc	{...+}	... Kleinbuchstaben --> Großbuchstaben
>-nn etc	{...-}	... Großbuchstaben --> Kleinbuchstaben
>:nn etc	{...;}	... a, b, ..., \$, ... --> ä, ^b, ..., ^\$, ...
>:nn etc	{...!}	... ä, ^b, ..., ^\$, ... --> a, b, ..., \$, ...

## Einzelzeichen

?	\?	Fragezeichen
*	\*	Stern
[	\[	eckige Klammer auf
]	\]	eckige Klammer zu
{	\{	geschweifte Klammer auf
}	\}	geschweifte Klammer zu
>a >A	\a	Kleinbuchstabe a
<a <A	\A	Großbuchstabe A
\	\\	Backslash
<<	<	spitze Klammer auf
>>	>	spitze Klammer zu

## Sonstiges

<	{[}	linker Rand
>	{]}	rechter Rand
><	{ }	in Sortieralphabet-Tabellen: Umschalten auf höchste Wertigkeiten

## Im Editor:

CTRL+K + erstes Sonderzeichen, das in den Klammern  
vorgesehen ist, erleichtert die Eingabe von {...} .

CTRL+K + Blank zeigt eine Übersicht über die nach #pa,{}  
geltenden Konventionen zum Schreiben der Parameter  
und der entsprechenden Editor-Anweisungen.