Sortieren mit TUSTEP (Teil 1)

Daten zu sortieren ist ein häufiges Problem, das in vielen Anwenderberatungen auftaucht. TUSTEP stellt dafür ein flexibles Lösungsangebot zur Verfügung. Viele TUSTEP-Benutzer wagen sich aber an solch »komplizierte« Programme nicht heran. Für sie sollen in dieser BI, mit Fortsetzung in der nächsten Nummer, die Grundprobleme des Sortierens erklärt und einige Beispiele zur Lösung der Standardsortieraufgaben mit TUSTEP gezeigt werden.

Grundprobleme des Sortierens

Sortierte Daten begegnen einem überall: Namen im Telefonbuch, Kärtchen einer Kartei, Listen und Register. All die Sortierungen haben den Zweck, die Daten in eine festgelegte Reihenfolge zu bringen, so daß jeder einzelne Eintrag innerhalb der Liste einen eindeutigen Platz bekommt.

Die mit der Sortierung verbundenen Probleme beginnen aber nicht erst bei der Frage, wie Umlaute und ß zu behandeln sind, sondern schon vorher bei der Abgrenzung der Einträge, die von der Sortierung betroffen sind (Sortiereinheit), bei der Festlegung des Teils eines Eintrags, der für die Sortierung relevant ist (Sortiertext), und erst dann ist die Frage nach den Alphabetisierungsregeln zu stellen (Sortierschlüssel).

Um eine flexible Lösung dieser Probleme zu gewährleisten, schaltet man beim Arbeiten mit TUSTEP vor das eigentliche Sortieren ein eigenes Programm SORTIER-VORBEREITE, das die Texte für die Sortierung aufbereitet.

1. Die Sortiereinheit

Die Sortiereinheit ist jeweils der ganze Eintrag, der zusammengehörende Text, der als Ganzes von der Sortierung betroffen ist. Im Telefonbuch ist eine solche Einheit Name, Vorname, ggf. Beruf, Firma, Adresse und die Telefonnummer. In einer Kartei ist die Sortiereinheit eine Karteikarte.

Diese selbstverständlichen Erfahrungen mit Sortiereinheiten sind auch bei Sortierungen mit Computer-Programmen zu beachten. Sortiert man Daten mit TUSTEP, so muß man schauen, ob das, was man inhaltlich als eine Sortiereinheit betrachten will, schon genau in einem TUSTEP-Satz steht; der TUSTEP-Satz ist die Sortiereinheit, die TUSTEP von sich aus annimmt. Meist wird sich die Sortiereinheit über mehrere Sätze erstrecken. In diesem Fall muß man angeben, welches Kennzeichen (eindeutige Zeichenfolge) in den Quelldaten den Beginn einer neuen Sortiereinheit kennzeichnet. Im vorliegenden Beispiel (Abb. 1) markiert die Zeichenfolge **&a** jeweils den Beginn einer Sortiereinheit, die man über Parameter (AA) dem Programm mitteilt, so daß für die weitere Verarbeitung der Text zwischen zwei solchen Kennungen als Sortiereinheit gilt, die als ganzes durch die Sortierung ihren bestimmten Platz im Ergebnis erhält.

2. Der Sortiertext

Der Benutzer einer Bibliothek bestellt sein Buch in der Regel anhand eines (alphabetischen) Autorenkatalogs; der Bibliothekar benötigt zusätzlich einen Standortkatalog seiner Bibliothek. Bei beiden Katalogen handelt es sich um die gleichen Karteikärtchen (Sortiereinheiten), die jeweils unter anderer Hinsicht sortiert sind. Im Standortkatalog gilt als Kriterium fürs Sortieren die Signatur, im Autorenkatalog der Autorenname. D. h. in den beiden Sortierungen besteht ein Unterschied im *Sortiertext*. Die Wahl des Sortiertextes bestimmt die Hinsicht, unter der die Sortierung erfolgen soll, denn in gedruckten Listen kann man nicht mehrere Kriterien ineinander mischen.

Durch Auswahl mehrerer Textteile als Sortiertext kann man eine Hierarchie der Sortierkriterien festlegen, wie z. B. Autorenname und Jahreszahl.

Ausgewählt wird der Sortiertext aus der Sortiereinheit durch die Angabe von eindeutigen Zeichenfolgen (Kennungen), die in den Quelldaten stehen und durch Parameter dem Programm mitgeteilt werden. Im Beispiel von Abb. 1 beginnen die Autoren mit der Zeichenfolge **&a** und enden vor dem nächsten **&**, das Erscheinungsjahr beginnt mit **&j** und endet vor dem nächsten **&**.

Das Auswählen von Textteilen aus den Quelldaten für bestimmte Zwecke setzt eine gewisse Aufbereitung der Daten voraus. Auf die Aufbereitung der Quelldaten wird im zweiten Teil in der nächsten BI eingegangen.

3. Der Sortierschlüssel

Sind die Sortiereinheiten festgelegt und die Sortiertexte daraus isoliert, so stellt sich die Frage nach den Alphabetisierungsregeln: welches Alphabet soll gelten, wie sind Umlaute zu behandlen, was soll mit Akzenten und Sonderzeichen geschehen. Diese Probleme werden mit Hilfe des Sortierschlüssels gelöst, der aus dem Sortiertext aufgebaut wird. Er ist dann allein für die eigentliche Sortierung mit dem Kommando SORTIERE verantwortlich.

Der Sortierschlüssel hat eine doppelte Funktion. Zunächst eine inhaltliche: bei der Erstellung des Sortierschlüssels aus dem Sortiertext kann der Benutzer all die im folgenden genannten Probleme und Regeln der Alphabetisierung berücksichtigen. Zum anderen eine technische: der Sortierschlüssel hat eine Form, die es dem eigentlichen Sortierprogramm erlaubt, mit schnell arbeitenden Algorithmen die Sortierung vorzunehmen. Sortiert werden nicht die Zeichen des Sortiertextes, sondern jedem dieser Zeichen wird für den Sortierschlüssel ein Wert zugewiesen, der dann die Grundlage für die Sortierung ist.

Beim Aufbauen des Sortierschlüssels kann man u. a. ein ganz neues Sortieralphabet angeben, d. h. die Reihenfolge der Zeichen gänzlich neu bestimmen. Mit dem Parameter A1 (Alphabet für den **1.** Sortierschlüssel) läßt sich z. B. das hebräische Sortieralphabet **abgdhuzxjklmnsypcqrwt** festlegen.

Die Umlaute **ä, ö, ü** werden nach DIN 5007 **ae, oe, ue** gleichgeachtet und wie diese doppelten Buchstaben hinter **ad, od, ud** eingeordnet (*Hadek - Häberle - Haffner*). Das gleiche gilt für den Buchstaben **ß**, der wie **ss** behandelt wird (*Masrich - Maßen - Mast*). Die gewünschte Anordnung kann dadurch erreicht werden, daß man die Zeichen **ä, ö, ü, ß** des Sortiertextes für den Sortierschlüssel in die beiden entsprechenden Zeichen austauscht, so daß im Sortierschlüssel statt *Häberle* die Schreibweise *Haeberle*, und *Massen* statt *Maßen* steht. Damit ergibt sich bei alphabetischer Sortierung die richtige Reihenfolge. Dem Programm SORTIER-VORBEREITE schreibt man das Austauschen durch Parameter XS1 (EXchange im Sortierschlüssel **1**) vor (XS1 /ä/ae/; XS1 /ö/oe/ etc.)

Probleme treten auf, wenn es - wie in dem Beispiel unten - neben einem *Jäger* noch einen *Jaeger* gibt, der sich mit *ae* schreibt. Nach dem Austauschen der Umlaute lauten im Sortierschlüssel beide Namen gleich: *Jaeger*. Sie werden beide an die richtige Stelle nach *Jader* einsortiert, können aber in sich nicht unterschieden werden. Als Abhilfe bietet sich an, aus dem Sortiertext zusätzlich einen weiteren Sortierschlüssel aufzubauen, der zur Unterscheidung von **ae** von **ä** dient. Die gewünschte Reihenfolge ist gemäß DIN 5007 **ä** nach **ae**, **ö** nach **oe** etc. In SORTIER-VORBEREITE tauscht man mit Parameter XS2 (die 2 steht für den zweiten Sortierschlüssel) die Umlaute **ä, ö, ü** aus in **az, oz, uz**, entsprechend **ß** in **sz**. Diese Veränderung ist für die meisten Wörter irrelevant, da sie sich bereits im ersten Sortierschlüssel unterscheiden und damit alle weiteren Zeichen (zweiter Sortierschlüssel) keine Rolle mehr spielen (vergegenwärtigen Sie sich: *Azalee* steht vor *Baum*, denn nachdem der erste Buchstabe die Reihenfolge festlegt, ist vollkommen uninteressant, welcher als nächstes folgt). Bei *Jaeger* und *Jäger* aber, die sich im ersten Sortierschlüssel nicht unterscheiden, wird der zweite Sortierschlüssel wichtig: es wird *Jaeger* (entstanden aus *Jaeger*) mit *Jazger* (entstanden aus *Jäger*) verglichen, womit sich im Ergebnis die richtige Reihenfolge ergibt.

Ähnlich kann man mit den Akzenten beim Aufbauen des Sortierschlüssels verfahren. In erster Linie sollen die Akzente beim Sortieren nicht berücksichtigt werden, d. h. **é** ist wie **e** hinter **d** einzuordnen. In TUSTEP sind die Akzente als fliegende Akzente vor dem jeweiligen Buchstaben codiert (z. B. **%/e**). Für den ersten Sortierschlüssel ist diese Codierung einfach ersatzlos zu streichen. Das geschieht in SORTIER-VORBEREITE mit Parameter XS1 (XS1 \%/\\). Ähnlich wie bei den Umlauten soll aber **é** und **e** bei ansonsten vollkommen gleichen Wörtern durchaus unterschieden werden; die gewünschte Reihenfolge ist *Denoix* vor *Dénoix*. Dies erreicht man wieder durch den zweiten Sortierschlüssel: die Codierung des Akzentes wird ausgetauscht in ein **z**. Sollen verschiedene Akzente in sich noch unterschieden werden, so kann man sie im zweiten Sortierschlüssel durch eine Zahl, die dem **z** folgt, unterscheiden. Man tauscht im zweiten Sortierschlüssel **é, è, ê, ë** aus in **z1e, z2e, z3e, z4e**. Verglichen wird im zweiten Sortierschlüssel *Denoix* mit *Dz1enoix* (entstanden aus *Dénoix*) und *Dz3enoix* (entstanden aus *Dênoix*), wodurch sich die gewünschte Reihenfolge ergibt. Im ersten Sortierschlüssel lauten alle einheitlich *Denoix* und unterscheiden sich von *Dubois*.

Sortieren nach Autoren und Jahreszahl (Beispiel)

Das erste Beispiel bietet die Standardlösung für das Sortieren einer Bibliographie. Weitere Beispiele zu spezielleren Problemen folgen in der nächsten BI.

Die Bibliographie (Abb. 1) soll nach Namen und Vornamen der Autoren sortiert werden; mehrere Werke eines Autors werden nach der Jahreszahl angeordnet. Das Problem wird zur besseren Nachvollziehbarkeit in drei Versuchen gelöst. Für einige Anwendungen bringen bereits die Teillösungen ein befriedigendes Ergebnis.

1. Sortierung nach dem gegebenen Wortlaut

Die einfachste Art zu sortieren ist nach dem Wortlaut der Sortiereinheiten. D. h. jeder Bibliographieeintrag wird nach seinem Beginn alphabetisch sortiert. Wenn die Daten bereits eine einheitliche Form haben, bringt das bereits ein sinnvolles Ergebnis.

Wichtig ist aber auch hierbei die Festlegung der Sortiereinheiten. Schaut man in die Quelldatei (Abb. 1), stellt man fest, daß sich die Bibliographieeinträge jeweils über mehrere Sätze erstrecken. Nur ein Eintrag steht bereits in genau einem Satz (6.1) und würde von TUSTEP ohne weitere Angaben als Sortiereinheit richtig erkannt. Man muß dem Programm den Beginn einer Sortiereinheit mitteilen; in den Beispieldaten beginnt eine Sortiereinheit jeweils mit der Kennung für den Autor &a.

Liegt die Sortiereinheit fest, braucht für eine Sortierung nach dem Wortlaut nur noch angegeben zu werden, wieviele Zeichen im Sortierschlüssel berücksichtigt werden sollen. Für die anschließende Sortierung ist eine genau definierte Anzahl von Zeichen relevant. Wird für den Sortierschlüssel ausschließlich mit Parameter SSL die Länge angegeben - im vorliegenden Fall kann man die Länge z. B. auf 50 festlegen - dann heißt dies, daß aus den ersten Zeichen der Sortiereinheit nach dem Standardsortieralphabet ein 50 Zeichen langer Sortierschlüssel aufgebaut wird und in der Zieldatei von SORTIER-VORBEREITE vor dem Text der Sortiereinheit auf den Positionen 1-50 abgespeichert wird.

Die in Abb. 2 wiedergegebene Kommandofolge ist ein erster Versuch für die gewünschte Sortierung. Die Quelldatei für SORTIER-VORBEREITE ist die Datei BIB (vgl. Abb. 1),

Die anschließende Sortierung mit dem Kommando SORTIERE erfolgt genau nach den eben beschriebenen 50 Zeichen (Spezifikation *Sortierfeld*), die nach der Sortierung ihren Zweck erfüllt haben und wieder getilgt werden können (Spezifikation *Tilgen*), so daß die Sortiereinheiten in der Zieldatei von SORTIERE in ihrem Wortlaut gegenüber der ursprünglichen Quelle unverändert sind. Der anschließende Aufruf des Kommandos KOPIERE dient dazu, die Satznummern aufsteigend zu vergeben, damit die Datei im EDITOR bearbeitet werden kann (denn durch SORTIER-VORBEREITE und SORTIERE blieben die Sortiereinheiten unverändert, d. h. auch die ursprünglichen Satznummern hatten sich noch nicht geändert).

Das Ergebnis steht nachher in der Datei BIBSO, wobei die Daten in BIB weiterhin unverändert zugänglich sind. Die Zwischendatei -STD- braucht den Benutzer nicht weiter zu interessieren.

In der Ergebnisdatei BIBSO stehen die Bibliographieeinträge in folgender Reihenfolge (angegeben wird hier zur Verständigung die laufende Nummer des Eintrags in der Quelldatei, sie entspricht der Seitennummer in der Quelldatei): Nr. 3 - 7 - 1 - 5 - 8 - 4 - 6 - 2. *Auer* (Nr. 3) ist erwartungsgemäß an erster Stelle; die beiden *Denoix* sind durch die Codierung des Akzentes unterschieden: % (Nr. 7) kommt vor **e** (Nr. 1). Da keine Akzentbehandlung angegeben wurde, ist nach der Standard-Sortierfolge sortiert worden. Nr. 5 ist *Jaeger* mit **ae**. Er steht zuerst, da hier das **e** mit dem **g** der übrigen Jäger verglichen wird, weil **ä** und **a** als gleichwertig behandelt werden. *Jäger, Gustav* führt die *Jäger* mit **ä** wegen seines Vornamens an. Die beiden *Zenger* unterscheiden sich durch den Titel, der auf den Autorennamen folgt: *Hessen* (Nr. 6) kommt vor *Statistische* (Nr. 2).

2. Sortierung nach Autorennamen mit Berücksichtigung der Umlaute

Der erste Versuch einer Sortierung brachte zwar ein Ergebnis, das eine erste alphabetische Orientierung gibt, doch ist es noch nicht befriedigend, was die Behandlung der Umlaute und Akzente betrifft.

Um eine in diesem Punkt korrekte Sortierung zu zeigen, sollen im nächsten Versuch nur die Namen der Autoren berücksichtigt werden. Die Beschränkung des Sortiertextes

auf die Autoren erfolgt über die Kennungen in den Quelldaten. Der Sortiertext beginnt hinter **&a** (Parameter AK1) und endet vor der Kennung für die nächste Kategorie **&** (Parameter EK1).

Bei der Sortierung der Umlaute und Akzente verfährt man so wie oben geschildert. Man braucht dafür zwei Sortierschlüssel, wobei die Umlaute in den beiden Sortierschlüsseln jeweils anders ausgetauscht werden (z. B. **ä** zu **ae** bzw. **az**).

Für beide Sortierschlüssel ist ihre Länge anzugeben. Das für die Sortierung wichtige Sortierfeld setzt sich dann aus beiden Sortierschlüsseln zusammen.

Abb. 3 zeigt das Programm, das die Datei BIB (vgl. Abb. 1) nun mit einer bezüglich der Namen korrekten Sortierung in die Datei BIBSO1 sortiert.

In der Ergebnisdatei stehen die Bibliographieeinträge in der Reihenfolge: Nr. 3 - 1 - 7 - 8 - 5 - 4 - 2 - 6. Die beiden *Denoix* (Nr. 1 und 7) stehen richtig: die Schreibweise mit Akzent folgt der ohne Akzent. Bei den *Jäger*s sieht man, daß **ae** und **ä** in der gewünschten Reihenfolge (**ä** hinter **ae**, wenn der Rest identisch ist) angeordnet werden. *Jäger, Gustav* wird nach DIN korrekt, vor *Jakob*, eingeordnet. Die Anordnung kam aufgrund folgender Sortierschlüssel zustande:

jaeger, gustav jazger, gustav
jaeger, jakob jaeger, jakob
jaeger, jakob jazger, jakob

Bevor die Zeichen des zweiten Sortierschlüssels, der für die Unterschiedung **ä** zu **ae** verantwortlich ist, verglichen werden, wird der Vorname relevant. Daher wird *Gustav* vor *Jakob* sortiert, unabhängig davon, daß er gemäß dem zweiten Sortierschlüssel dahinter kommen sollte.

Ein Problem hat der aktuelle Lösungsversuch noch nicht bewältigt: er unterscheidet nicht innerhalb der Werke eines Autors. Nr. 2 und Nr. 6, die beiden Werke von *Zenger*, stehen in der Reihenfolge der Quelldatei. Als Kriterium zu ihrer Unterscheidung soll laut Aufgabenstellung die Jahreszahl dienen.

Und eine Ergänzung: so wie die Kommandofolge jetzt eingerichtet ist, steht in der Zieldatei eine Sortiereinheit in einem Satz (das ist die einzige Veränderung der Daten gegenüber der Quelldatei). U. u. ist es übersichtlicher, wenn wie in der Quelldatei jeweils eine Kategorie eines Bibliographieeintrages in einer eigenen Zeile steht, und ein Bibliographieeintrag jeweils die gleiche Seitennummer erhält. Auch das soll im dritten Versuch berücksichtigt werden.

3. Sortierung nach Name, Vorname und Jahreszahl

Als Sortiertext wird der Autor (**&a** bis **&**) isoliert; zusätzlich wird als Sortiertext die Jahreszahl (**&j** bis **&**) genommen. Erstes Sortierkriterium sind die Namen, wobei die Umlaute in die entsprechenden doppelten Buchstaben ausgetauscht werden. Als zweites Kriterium dient der Name, wobei die Umlaute in die entsprechende Kombination mit **z** ausgetauscht werden. Erst dann werden die Jahreszahlen berücksichtigt.

Angestrebt wird damit folgender Aufbau des Sortierschlüssels:

jaeger, gustav jazger, gustav 1987
jaeger, jakob jaeger, jakob 1957
jaeger, jakob jazger, jakob 1953
zenger, zacharias zenger, zacharias 1970
zenger, zacharias zenger, zacharias 1975

Die Jahreszahl wird erst bei vollkommener Identität der Namen (*Zenger*) als Unterscheidungskriterium herangezogen. *Jäger, Jakob* und *Jaeger, Jakob* werden unabhängig von der Jahreszahl weiterhin durch ihren Umlaut unterschieden.

Erreichen kann man die Anordnung, indem man für den Sortierschlüssel aus dem Sortiertext mit den Parametern AS1, ES1, AS2, ES2 und AS3, ES3 (Anfang bzw. Ende des Sortierschlüssels **1-3**) bestimmte Teile auswählt. Die Auswahl geht so vor sich:

Aus der gesamten Sortiereinheit wird der Sortiertext erstellt, z. B.:

Jäger, Jakob 1985

Daraus erstellt man die Sortierschlüssel. Der erste Sortierschlüssel wird auf den Namen beschränkt (Ende ist das Blank vor der ersten Ziffer). Hier werden die Umlaute in **ae**, **oe**, etc. ausgetauscht. Erster Sortierschlüssel ist:

jaeger, jakob

Als zweiter Sortierschlüssel dient der gesamte Sortiertext. Hier werden die Umlaute in **az**, **oz**, etc. ausgetauscht, was die Jahreszahl nicht stört. Zweiter Sortierschlüssel ist:

jazger, jakob 1985

Die beiden Sortierschlüssel bilden dann zusammen das für die Sortierung relevante Sortierfeld. Es enthält folgenden Text:

jaeger, jakob jazger, jakob 1985

Durch Ausgabe des Testprotokolls bei SORTIERVORBEREIE läßt sich das Aufbauen des Sortierschlüssel dokumentieren und damit für die Fehlersuche die Bearbeitung jeder Sortiereinheit nachvollziehen.

Das Programm in Abb. 4 liefert das gewünschte Ergebnis.

Abbildung 1

```
1.1   |&a Denoix, Ren%/e
1.2   |&t Geschichte der Stadt Trier
1.3   |&j 1980
1.4   |&o München
1.5   |&s UB: 10 A 6384
1.6   |&k Hier steht beliebig vieler
      Kommentar: Zitate,
      Bearbeitungshinweise, etc.
2.1   |&a Zenger, Zacharias
2.2   |&t Statistik zur Geschichte
2.3   |&z Historische Quartalschrift
2.4   |&b 34
2.5   |&j 1968
2.6   |&o Koblenz
2.7   |&s HI: Ze 1.09
3.1   |&a Auer, Alfons
3.2   |&t Die Geschichte der Ethik
3.3   |&j 1988
3.4   |&o St. Ottilien
3.5   |&s LB: Vi VIII 229
4.1   |&a Jäger, Jakob
4.2   |&t Leben und Werk Karls V.
4.3   |&j 1953
4.4   |&o Göttingen
4.5   |&s LB: Vi VIII 25
5.1   |&a Jaeger, Jakob
5.2   |&t Württemberg und Hohenzollern
5.3   |&z Historische Quartalschrift
5.4   |&b 23
5.5   |&j 1957
5.6   |&o Koblenz
5.7   |&s Historisches Institut: Ze 1.09
5.8   |&k Auch hier steht Kommentar
6.1   |&a Zenger, Zacharias
6.2   |&t Hessen und seine Wirtschaft
6.3   |&j 1970
6.4   |&o Trier
6.5   |&s UB: 9 C 10387
7.1   |&a D%/enoix, Ren%/e
7.2   |&t Einführung in die Geschichte
7.3   |&j 1971
7.4   |&o Köln
7.5   |&s Historisches Institut: Ac 35.16
8.1   |&a Jäger, Gustav
8.2   |&t Mein Tübingen
8.3   |&j 1987
8.4   |&o Reutlingen
8.5   |&s LB: Vi IX 37
```

Abbildung 2

```
#sv,bib,-std-,-,+,*
      Beginn der Sortiereinheit
aa        |&a|
      Länge des Sortierschlüssels
ssl       50
*eof
#so,-std-,-std-,1+50,+,1+50
#ko,-std-,bibso,+,+
```

Abbildung 3

```
#sv, bib, -std-, -, +, *
      Sortiereinheit
aa        |&a|
      Sortiertext ist der Autor
ak1       |&a |
ek1       | &|
      Anfangskennung nicht zum Sortiertext
aei       11
      Austauschen Umlaute/Akzente
        für 1.Sortierschlüssel
xs1       |ä|ae|ö|oe|ü|ue|ß|ss|
xs1       |%/||%\||%>>||%:||
        für 2.Sortierschlüssel
xs2       |ä|az|ö|oz|ü|uz|ß|sz|
xs2       |%/|z1|%\|z2|%>>|z3|%:|z4|
ssl       25  25
*eof
#so, -std-, -std-, 1+50, +, 1+50
#ko, -std-, bibso1, +, +
```

Abbildung 4

```
#sv,bib,-std-,-,+,*
      Sortiereinheit
aa        |&a|
      Sortiertext: Autor (1); Jahr (2)
ak1       |&a |
ek1       | &|
ak2       |&j |
ek2       | &|
      Anfangskennung nicht zum Sortiertext
aei       11  11
      Ende 1. Schlüssel: Blank - Ziffer
es1       | >/|
      Austauschen im 1./2. Sortierschl.
xs1       |ä|ae|ö|oe|ü|ue|ß|ss|
xs1       |%/||%\||%>>||%:||
xs2       |ä|az|ö|oz|ü|uz|ß|sz|
xs2       |%/|z1|%\|z2|%>>|z3|%:|z4|
ssl       25   30
*eof
#so,-std-,-std-,1+55,+,1+55
#ko,-std-,bibso3,+,+,*
      Zeilenanfang bei &
za        |&|
      Neue Seitennummer bei &a
sa        |&a|
*eof
```