



syska EURO FIBU – Finanzsoftware für alle, die ihren Weg machen.

syska EURO FIBU 2022

Der flexible ASCII-Import - Beschreibung

**Inhaltsverzeichnis**

<b>1</b>	<b>Einleitung .....</b>	<b>3</b>
<b>2</b>	<b>Bedienung .....</b>	<b>3</b>
<b>3</b>	<b>Definitionsdatei bearbeiten .....</b>	<b>4</b>
3.1	Kontextmenü und Schaltflächen .....	4
3.1.1	Neu .....	4
3.1.2	Eigenschaften .....	5
3.1.3	Ausschneiden .....	5
3.1.4	Kopieren .....	5
3.1.5	Einfügen .....	5
3.1.6	Löschen .....	5
3.1.7	Einrücken.....	5
3.1.8	Ausrücken.....	5
3.1.9	Speichern .....	5
3.1.10	Speichern als.....	5
<b>4</b>	<b>Bestandteile einer Importdefinition.....</b>	<b>6</b>
4.1	Import .....	6
4.1.1	Trennzeichen .....	7
4.1.2	Variablen .....	7
4.2	Datei .....	8
4.3	Datensatz .....	9
4.4	Datenfeld .....	9
4.4.1	Spezialfelder in Datensätzen .....	10
4.5	Datum .....	10
4.6	Kommentar .....	11
4.7	Konstante .....	11
4.8	Strukturierung .....	12
4.9	Zahl.....	13
4.10	Zeichenkette .....	14
4.10.1	Zeichenmengeneigenschaften.....	16
<b>5</b>	<b>Beispiel.....</b>	<b>16</b>

# 1 Einleitung

Mit dem flexiblen ASCII-Import von Stamm- und Bewegungsdaten stellt Ihnen die syska EURO FIBU ein Modul zur Verfügung, mit dessen Hilfe Sie problemlos Daten aus Alt- oder Fremdprogrammen reproduzierbar übernehmen können. Die Definition der Datenformate für den flexiblen Import ist allerdings relativ komplex - sie sollte nur von erfahrenen Anwendern vorgenommen werden. Wenden Sie sich im Zweifelsfall bitte an Ihren Fachhändler.

Flexibel bedeutet, dass das Datenformat (=Beschreibung des Datensatzes) der Importdatei (=Datenart) frei definiert werden kann. Die syska EURO FIBU verwendet dafür eigene Beschreibungsdateien, die Sie an der Dateierweiterung „.FID“ erkennen. Innerhalb einer Beschreibungsdatei - wie z.B. der BUBE.FID - wird der genaue Aufbau desjenigen Datensatzes beschrieben, der importiert werden soll. Flexibel bedeutet aber auch, dass Sie problemlos Datensätze wahlweise mit Trennzeichen oder mit einer festen Länge importieren können. Haben Sie die Beschreibungsdatei für eine bestimmte Datenart einmal erstellt, dann kann diese immer wieder für einen Import herangezogen werden.

Um einen Datenimport durchzuführen, wählen Sie durch Anklicken des entsprechenden Schalters zuerst die Datenart aus, für welche der Import erfolgen soll. Als nächstes wählen Sie im Bereich „Importdatei“ die Datei mit den zu importierenden Daten. Anschließend ordnen Sie im Bereich „Definitionsdatei“ die passende Beschreibungsdatei für den flexiblen Import zu. Ein Mausklick auf die Schaltfläche OK startet den Importlauf.

Beim Import von Buchungen haben Sie zusätzlich die Möglichkeit, über die Optionsfelder „Buchungsbelege“ und „Stapelbuchungen“ Buchungen entweder direkt zu verbuchen oder in einen Buchungsstapel importieren zu lassen. Im Gegensatz zum ASCII-Import erzeugt der flexible Import die Steuerumbuchung bei EG-Buchungen und Buchungen mit dem Verrechnungssatz automatisch, so dass diese nicht mit übergeben werden muss.

**Wichtig:** Eine der wichtigsten Voraussetzung für den Datenimport ist das Vorliegen der Daten in einem lesbaren Format, d.h., sie dürfen nicht binär codiert oder ähnliches sein. Ob Daten im ASCII-Format lesbar sind, sehen Sie am besten daran, dass diese z.B. mit einem Editor eingesehen werden können.

**Tip:** Sie finden in Ihrem EUROFIBU-Verzeichnis bereits Definitionsdateien zum ASCII-Import/Export-Format der syska EURO FIBU. Diese können Sie in eine andere Datei speichern und als Vorlage zur Definition Ihres eigenen Importformates verwenden. Sie dienen aber auch als Beispiel, um die doch recht komplexen Möglichkeiten zur Erzeugung einer Definitionsdatei zu illustrieren.

## 2 Bedienung

Sie erreichen den flexiblen Import über das Menü (Datei → Import → Flexibler Import).

- Zunächst wählen Sie die Art der zu importierenden Daten.
- Als nächstes ist die Datei mit den entsprechenden zu importierenden Daten zu wählen. Hier wird ein Standardname vorgeschlagen, abhängig von der Datenart.
- Zuletzt wählen Sie die Definitionsdatei für die gewählte Datenart. Auch hier wird ein Standardname in Abhängigkeit der gewählten Datenart

vorgeschlagen. Diese Definitionsdatei bestimmt, wie die Daten interpretiert werden. Sie können eine eigene Datei anlegen oder vorhandene Dateien bearbeiten.

### 3 Definitionsdatei bearbeiten

Dieser Dialog erscheint, sobald Sie im Dialog zum flexiblen Import die Schaltfläche NEU oder BEARBEITEN angeklickt haben. Er besteht hauptsächlich aus einer großen Strukturansicht, worin die Beschreibung des Formates der zu importierenden Daten vorgenommen wird. Die weiteren Elemente sind eine Reihe Schaltflächen am oberen und am unteren Rand. Die Bearbeitung der Struktur kann über das Kontextmenü oder die Schaltflächen am oberen und unteren Rand, die den Menüpunkten des Kontextmenüs entsprechen, erfolgen.

Wichtig: Um das Format der zu importierenden Daten beschreiben zu können, benötigen Sie zunächst eine möglichst präzise Beschreibung des Formates. Wenden Sie sich dazu bitte an den Hersteller bzw. Erzeuger des Formates, um eine solche zu erhalten.

Das Format der zu importierenden Daten wird durch Aufzählung der Einzelbestandteile und deren Eigenschaften innerhalb der Importdatei(en) beschrieben. Diese Aufzählung erfolgt mittels einer Baumstruktur, die das Importformat vom Stamm aus immer feiner beschreibt. In der Strukturansicht findet sich daher diese als Baumstruktur in einer Baumanzeige wieder, die die einzelnen Bestandteile mit kleinen Bildern und Texten abstrakt darstellt. Jede Zeile der Baumanzeige entspricht genau einem Bestandteil des Beschreibungsformates. Die Darstellung dieses Bestandteiles besteht aus drei Teilen:

- Ganz links ist entweder ein Plus-, Minus- oder Rautensymbol zu finden. Falls ein Plus dargestellt wird, so existieren Kinderbestandteile, die gerade nicht angezeigt werden. Durch Klicken auf das Plus werden diese Bestandteile unterhalb des aktuellen Teiles - des sog. Elternbestandteils - etwas nach rechts eingerückt dargestellt. Das Minus-Symbol dagegen zeigt an, dass gerade alle Kinderbestandteile angezeigt werden und Klicken auf das Minus bewirkt das Verstecken dieser Teile. Wird eine Raute angezeigt, so existieren keine Kinderbestandteile.
- In der Mitte befindet sich ein weiteres Symbol, das das Element oder den Typ des Elementes abstrahiert.
- Ganz rechts ist eine Kurzbeschreibung des Elementes, zum Teil sogar mit den entsprechenden Daten zu sehen.

Die Baumstruktur kann über das Kontextmenü geändert werden; dieses erscheint, sobald Sie über einem markierten Bestandteil der Baumanzeige die rechte Maustaste drücken.

#### 3.1 Kontextmenü und Schaltflächen

##### 3.1.1 Neu

Damit wird ein neuer Bestandteil in die Baumstruktur eingefügt, und zwar nach dem letzten markierten Bestandteil auf der gleichen Hierarchieebene. Vor dem Einfügen muss der Typ des Bestandteiles durch Auswahl innerhalb der angezeigten Liste der Typen ausgewählt werden.

In den beiden obersten Ebenen stehen nur die Bestandteile „Datei“, „Kommentar“, „Strukturierung“ zur Verfügung. Erst in der dritten Ebene kommen weitere Bestandteile hinzu. Dort stehen dann zur Verfügung: „Datei“,

„Datenfeld“, „Datensatz“, „Datum“, „Kommentar“, „Konstante“, „Strukturierung“, „Zahl“, „Zeichenkette“. Eine Beschreibung dieser Bestandteile erfolgt bei der Beschreibung der Bestandteile der Importdefinition.

Die Ebene eines Elementes können Sie durch Einrücken oder Ausrücken verändern, alle Kindelemente werden dabei ebenfalls um eine Ebene mitverrückt

### **3.1.2 Eigenschaften**

Damit kann der Eigenschaftsdialog zu einem ausgewählten Bestandteil geöffnet und aktiviert werden. Innerhalb dieses Dialoges werden die genauen Vorgaben für den Bestandteil eingestellt. Der Inhalt des Eigenschaftsdialoges hängt vom Typ des ausgewählten Bestandteiles ab. Den Eigenschaftsdialog können Sie auch direkt über die Schaltfläche EIGENSCHAFTEN unterhalb des Listfeldes aufrufen.

### **3.1.3 Ausschneiden**

Damit werden die ausgewählten Bestandteile ausgeschnitten und aus der Baumstruktur entfernt. Die Bestandteile bleiben bis zum nächsten Speichern oder neuem Ausschneiden in der Windows-Zwischenablage hinterlegt und können an anderer Stelle wieder eingefügt werden.

### **3.1.4 Kopieren**

Damit werden die ausgewählten Bestandteile in der Windows-Zwischenablage hinterlegt und können an anderer Stelle als Kopie wieder eingefügt werden.

### **3.1.5 Einfügen**

Damit werden vorher ausgeschnittene oder kopierte Bestandteile hinter dem letzten ausgewählten Bestandteil eingefügt.

### **3.1.6 Löschen**

Damit werden die markierten Bestandteile aus der Baumstruktur entfernt, allerdings bleibt keine Kopie davon erhalten.

### **3.1.7 Einrücken**

Damit ist es möglich, die markierten Bestandteile um eine Ebene innerhalb des Baumes zu versetzen. Das Versetzen bezieht sich dabei auf den auf gleicher Ebene direkt vor dem ersten markierten Teil in der Struktur stehenden Bestandteil, dieser erhält dann die einzurückenden Bestandteile als Kinder und mutiert also zu deren Elternbestandteil. Die markierten Bestandteile werden in der Baumanzeige eingerückt.

### **3.1.8 Ausrücken**

Damit ist es möglich, das Einrücken rückgängig zu machen. Dabei bekommen die markierten Kinderbestandteile einen neuen Elternbestandteil, nämlich genau den des bisherigen Elternbestandteiles; sie mutieren somit zu Geschwistern des ehemaligen Elternbestandteiles. In der Baumanzeige werden die markierten Bestandteile um eine Stufe nach links gerückt.

### **3.1.9 Speichern**

Damit wird der aktuelle Baum mit allen Eigenschaften gespeichert.

### **3.1.10 Speichern als**

Damit kann der aktuelle Baum unter anderem Namen gespeichert werden. Es erscheint ein Dateidialog, in dem der neue Name eingegeben werden kann.

Die bereits angesprochene Gliederung des Importes in Eltern- und Kinderbestandteile und die daraus resultierende Baumstruktur teilt beim Importieren der syska EURO FIBU mit, wo innerhalb einer Datei ein Datensatz (z. B. eine Buchung) beginnt und wo innerhalb dieses Datensatzes die Datenfelder stehen (z. B. der Buchungstext), wobei für jedes Datenfeld wieder wichtig ist, in welcher Form es vorliegt (z. B. Reihenfolge von Tag, Monat, Jahr in einem Datumswert). Diese Gliederung in Bestandteile wird durch die Baumstruktur erreicht: Die Eltern-Kinder-Beziehung ist so zu verstehen, dass die Eltern aus den Kinderbestandteilen bestehen. Also werden Datenfelder sinnvollerweise Kinder von Datensätzen sein und Datensätze wiederum Kinder einer Importdatei (denn diese besteht ja aus den einzelnen Datensätzen).

Ein Bestandteil gilt nur dann als erkannt, wenn er alle an ihn gestellten Bedingungen erfüllt. Tritt beispielsweise eine Zahl auf, die 4 Stellen hat, und fordert ein Zahlenbestandteil eine Mindestlänge von 5 Stellen, so gilt dieser Zahlenbestandteil als nicht erkannt. Nicht erkannte Bestandteile führen nicht unbedingt zu Fehlern, sondern können gezielt eingesetzt werden, um Fälle zu unterscheiden.

## 4 Bestandteile einer Importdefinition

### 4.1 Import

Äußerster und alles enthaltender Bestandteil der Importbeschreibung. Enthält alle allgemeinen Eigenschaften des Imports wie Titel, Fehlerdateiname sowie Variablen- und Trennerverwaltung (s. u.).

#### Eingabefeld "Titel"

In das Eingabefeld "Titel" kann ein Name für den Import eingegeben werden. Der Titel ist unerheblich, dient aber der Übersichtlichkeit. Der Titel wird in der Baumanzeige angezeigt.

#### Liste "Variablen"

Diese Liste enthält alle definierten Variablen. Variable dienen dazu, Werte zwischenzuspeichern und an anderer Stelle wiederzuverwenden. Mehr dazu bei der Beschreibung des Bestandteiles Variable.

#### Liste "Trennzeichen"

Diese Liste enthält alle definierten Trennzeichen. Trennzeichen sind frei definierbare Folgen von konstanten Zeichen, die zur Trennung von Daten dienen. Beispielsweise kann ein Komma, dem Leerzeichen vorausgehen und folgen können, als Trennzeichen definiert werden. Mehr dazu bei der Beschreibung des Bestandteiles Trennzeichen.

#### Schaltflächen "Bearbeiten"

Zu jeder der Listen "Trenner" und "Variablen" existiert eine Schaltfläche "Bearbeiten". Ist eine Variable oder Trenner ausgewählt, so wird mit der Schaltfläche "Bearbeiten" der Eigenschaftsdialog des gewählten Bestandteiles aufgerufen.

#### Schaltflächen "+"

Zu jeder der Listen "Trenner" und "Variablen" existiert eine Schaltfläche "+", mit der eine neue Variable bzw. ein neuer Trenner angelegt werden können.

#### Gruppe "Fehlerdatei"

Mit den Einzelfeldern der Gruppe "Fehlerdatei" kann der Titel der evtl. erzeugten Fehlerdatei für den Import festgelegt werden. Ausgehend von der vorgegebenen zu importierenden Datei kann der Fehlerdateiname

zusammengestellt werden. Um abweichende Fehlerdateinamen zusammenzustellen, wird der jeweils abweichende Teil des Dateinamens in die Eingabefelder "Verzeichnis", "Datei" und "Extension" eingegeben. Soll also die Fehlerdatei genauso heißen wie die vorgegebene Importdatei, jedoch mit '.err' enden, so wird in das Eingabefeld "Extension" '.err' eingegeben.

### 4.1.1 Trennzeichen

Trennzeichen sind Bestandteile, die nicht innerhalb der Baumstruktur angezeigt werden. Sie sind global bezüglich des gesamten Importlaufes, und mit ihrer Hilfe können häufig vorkommende konstante Zeichensequenzen global definiert werden. Sind beispielsweise alle Daten mit Kommas getrennt, denen Leerzeichen vorausgehen und folgen dürfen, so kann dies durch geeignete Wahl eines Trennzeichens, auf den im Import immer wieder Bezug genommen wird, ausgedrückt werden. Sie werden im Eigenschaftenmenü des Imports verwaltet.

#### Eingabefeld "Eindeutiger Name"

Im Eingabefeld "Eindeutiger Name" wird der Name des Trenners eingetragen. Dieser muss eindeutig bezüglich aller anderen Trennernamen sein. Unter diesem Namen wird der Trenner später angesprochen.

#### Zeichenfeld "Zeichen"

Das Zeichenfeld "Zeichen" enthält alle Zeichen, die als Trennzeichen interpretiert werden. Im oben genannten Beispiel würde man das Komma als Trennzeichen definieren. Dies geschieht über die Schaltfläche "...", die hinter dem Zeichenfeld steht. Damit wird ein Zeichenmengendialog aufgerufen, indem die erlaubten Trennzeichen definiert werden können.

#### Escape

Mit Hilfe dieser Liste kann ein zuvor definiertes Trennzeichen als Escape-Zeichen ausgewählt werden. Dies hat zur Folge, dass ein Trennzeichen, dem ein Escape-Zeichen unmittelbar vorausgeht als normaler Text übernommen wird und nicht trennt. Mit dieser Option ist es nicht mehr nötig, bei jeder Zeichenkette das Escape-Zeichen auszuwählen.

#### Optionsschalter "Dateiende trennt ebenfalls"

Ist der Optionsschalter "Dateiende trennt ebenfalls" markiert, so gilt der Trenner als erkannt, auch wenn die Datei kein Zeichen mehr enthält.

#### Zeichenfeld "evtl. vorher"

Das Zeichenfeld "evtl. vorher" enthält alle Zeichen, die vor dem Trennzeichen beliebig oft auftreten dürfen. Damit können unnötige Füllzeichen erkannt und überlesen werden. Die Auswahl der Füllzeichen geschieht über die Schaltfläche "...", die hinter dem Zeichenfeld steht.

#### Zeichenfeld "evtl. nachher"

Das Zeichenfeld "evtl. nachher" enthält alle Zeichen, die nach dem Trennzeichen beliebig oft auftreten dürfen. Damit können unnötige Füllzeichen erkannt und überlesen werden. Die Auswahl der Füllzeichen geschieht über die Schaltfläche "...", die hinter dem Zeichenfeld steht.

### 4.1.2 Variablen

Variablen sind Bestandteile, die nicht innerhalb der Baumstruktur angezeigt werden. Sie sind global bezüglich des gesamten Importlaufes, und mit ihrer Hilfe können Werte gespeichert werden, die an anderer Stelle benötigt werden. Ein Wert wird einer Variablen über einen Datentypbestandteil wie

beispielsweise Konstante, Datum, Zeichenkette oder Zahl zugewiesen. Die Übergabe eines Variablenwertes wird in einem Datenfeldbestandteil wie z. B. Buchung oder Belegnummer vollzogen, wobei der Inhalt der Variablen als Initialwert für das Datenfeld benutzt werden kann. Sie werden im Eigenschaftenmenü des Imports verwaltet.

#### **Eingabefeld "Eindeutiger Name"**

Im Eingabefeld "Eindeutiger Name" wird der Variablenname der Variablen eingetragen. Dieser muss eindeutig bezüglich aller anderen Variablennamen sein. Unter diesem Namen wird die Variable später angesprochen.

#### **Eingabefeld "Standardwert"**

Im Eingabefeld "Standardwert" kann ein Startwert für die Variable definiert werden. Mit diesem wird die Variable zu Beginn des Importlaufes belegt. Zeichenketten werden in Anführungszeichen gestellt, Datumswerte enthalten zwei Dezimalpunkte (z. B. 17.4.87), wobei 0-Werte als nicht definiert gelten und entsprechend missachtet werden. Die konstante Zeichenkette <Systemdatum> bezeichnet das aktuelle Datum des Rechners; die konstante Zeichenkette <Geschäftsjahr> anstelle eines Jahreswertes steht für das aktuelle Geschäftsjahr des Mandanten. Zahlenwerte werden normal eingegeben (evtl. Vorzeichen, Vorkommastellen, evtl. Dezimalpunkt und Nachkommastellen).

## **4.2 Datei**

Bestandteil, der genau eine Importdatei beschreibt. Der Aufbau der Importdatei wird über die Kinderbestandteile beschrieben. Der Dateibestandteil beschreibt die Charakteristika der Importdatei wie Format und Name.

#### **Eingabefeld "Titel"**

In das Eingabefeld "Titel" kann ein beliebiger Dateititel eingegeben werden, der nur der Übersichtlichkeit dient. Der Titel der Datei wird in der Baumanzeige ausgegeben.

#### **Gruppe "Dateiname"**

Mit den Einzelfeldern der Gruppe "Dateiname" kann der Name der Importdatei festgelegt werden. Normalerweise entspricht dieser genau dem ausgewählten Importdateinamen, allerdings kann hier auch ein fester Dateiname gefordert werden. Ist beispielsweise der Name der Importdatei immer 'BUCH.TXT', so kann im Eingabefeld "Datei" 'buch' und im Eingabefeld "Extension" '.txt' eingetragen werden. Das bedeutet, dass stets im aktuellen Verzeichnis die Datei BUCH.TXT geöffnet wird. Entspricht der Importdateiname stets dem gewählten Dateinamen, so bleiben die Eingabefelder dieser Gruppe leer.

#### **Optionsschalter "Zeichensatz"**

Die Auswahlshalter definieren, in welchem Zeichensatz die vorgegebene Datei geschrieben wurde. Wird aus einer DOS-Anwendung importiert, so sollte der Optionsschalter "DOS" markiert sein, wird aus einer Windows-Anwendung importiert, so sollte der Schalter "Windows" markiert sein. Damit wird eine korrekte Übernahme von Zeichenketten insbes. von Umlauten garantiert.

#### **Gruppe "Zeichenersatz"**

Mit Hilfe des Zeichenersatzes kann ein ASCII-Zeichen durch ein anderes ASCII-Zeichen ersetzt oder verworfen werden. Im ersten Eingabefeld ist das zu ersetzende Zeichen anzugeben, im zweiten Feld der Ersatz oder nichts, falls das Zeichen verworfen werden soll. Die Zeichen werden als ASCII-Werte eingetragen (also z.B. 65 für 'a').

### **Optionsschalter "Existenz optional"**

Ist der Optionsschalter "Existenz optional" markiert, so wird kein Fehler ausgegeben, sobald die Datei nicht existiert, ihre Existenz also nicht notwendig ist. Die Auswahl dieses Schalters ist zum Beispiel bei Dateien sinnvoll, die evtl. noch zusätzliche Information enthalten, z. B. eine Vorlaufdatei, die den Zeitraum für Daten enthält.

## **4.3 Datensatz**

Bestandteil, mit dem ein Datensatz eingegrenzt wird. Damit ist es möglich, die vorkommenden Einzeldatenfelder eines Datensatzes zu bündeln, also als zusammengehörend zu markieren. Die Bestandteile des Datensatzes werden durch die Kinder beschrieben.

### **Auswahlliste "Typ"**

In der Auswahlliste "Typ" kann gewählt werden, welcher Datensatztyp durch die Kinder beschrieben wird. Der Importlauf startet stets mit Vorauswahl des zu importierenden Datensatztyps. Es werden nur die mit dem gewählten Typ verträglichen Datensätze importiert. So ist es möglich, aus einer Importdatei, die verschiedene Arten von Datensätzen (z. B. Sachkonten und Kreditoren) enthält, nur die Sachkonten oder nur die Kreditoren zu importieren. Wurde beispielsweise für den Importlauf der Datensatz Sachkonten gewählt und beginnt der Datensatz Kreditoren, so wird dieser zwar erkannt und bearbeitet, aber nicht importiert. Beginnt jedoch der Datensatz Sachkonten, so wird dieser erkannt, bearbeitet und danach importiert. Der Datensatztyp "Personenkonto" steht sowohl für Kreditoren als auch für Debitoren, d. h. je nach gewähltem Importdatensatztyp wird der Datensatz als Kreditor oder als Debitor betrachtet.

## **4.4 Datenfeld**

Bestandteil, mit dem ein Datenfeld eines Datensatzes eingegrenzt wird. Damit ist es möglich, die Grenzen eines Datensatzbestandteiles festzulegen. Die eigentlichen Daten des Datenfeldbestandteiles werden durch die Kinder beschrieben und Wert von den Kinderbestandteilen zugewiesen.

### **Auswahlliste "Datensatz"**

In der Auswahlliste "Datensatz" wird der Datensatz für dieses Feld ausgewählt. Ein Datenfeld wird nur dann beachtet, wenn der Typ des Datensatzes zum umliegenden Datensatzbestandteil passt.

### **Auswahlliste "Datenfeld"**

In der Auswahlliste "Datenfeld" wird das Datenfeld des Datensatzes ausgewählt, der durch die Kinder beschrieben wird, der also als nächstes auftreten soll.

### **Eingabeliste "Standardwert"**

In der Eingabeliste Standardwert kann ein Startwert für das Datenfeld angegeben werden. Damit kann z. B. ein Datenfeld, das in der Importdatei überhaupt nicht vorkommt, mit einem Standardwert belegt werden (in diesem Fall hat der Datenfeldteil normalerweise keine Kinderbestandteile).

Zeichenketten werden in Anführungszeichen gestellt, Datumswerte enthalten zwei Dezimalpunkte (z. B. 17.4.87), wobei 0-Werte als nicht definiert gelten und entsprechend missachtet werden. Die konstante Zeichenkette <Systemdatum> bezeichnet das aktuelle Datum des Rechners, die konstante Zeichenkette <Geschäftsjahr> anstelle eines Jahreswertes steht für das aktuelle Geschäftsjahr des Mandanten. Zahlenwerte werden normal eingegeben (evtl. Vorzeichen, Vorkommastellen, evtl. Dezimalpunkt und Nachkommastellen).

Falls das Datenfeld numerisch ist und sinnvoll vordefinierte konstante Werte besitzt, so befinden sich in der Auswahlliste die Konstanten in lesbarer Form.

#### **Auswahlliste "durch Variable"**

In der Eingabeliste "durch Variable" kann ein Startwert aus einer Variablen übernommen werden. Wurde also eine Variable mit einem Wert belegt, so wird dieser Wert als Standardwert genommen. Überschrieben wird dieser Wert jedoch, falls ein Datentypfeld, das Kind dieses Datenfeldbestandteils ist, einen Wert zuweist.

### **4.4.1 Spezialfelder in Datensätzen**

Dieses Kapitel beschreibt einige Besonderheiten bei bestimmten Datensätzen. Da diese Besonderheiten sich in Datenfeldern niederschlagen steht ihre Beschreibung hier.

#### **"Buchung.Soll-Haben-Kennung"**

Mit diesem Feld kann die Soll/Habenstellung der beteiligten Konten anhand eines Kriteriums (z.B. negativer Betrag, oder ein 'H' an irgendeiner Stelle im Datensatz) vertauscht werden. Mit Hilfe dieses Flags ist es also möglich, Buchungen die in Konto/Gegenkonto-Logik in der Importdatei aufgeführt sind, zu erkennen. Falls "Soll- u. Habenkonto vertauscht" (1) gewählt ist, wird die Soll/Habenstellung getauscht, ansonsten wird die Soll/Habenstellung als "normal" (0) beibehalten.

#### **"Buchung.Automatische Mischsteuersatzbestimmung"**

Durch Zuweisung eines Wertes aus "ja" (1) oder "nein" (0) kann festgelegt werden, ob bei Import einer Buchung mit Mischsteuersatz (d.h. einem sonstigen Steuersatz) dieser automatisch aus den beteiligten Konten ermittelt wird, falls er nicht gegeben ist. Wird "ja" gewählt, so wird bei Mischsteuerbuchungen und nicht gegebenem Steuersatz oder -betrag automatisch der im Konto hinterlegte Steuersatz angenommen und die Steuer gebucht, ansonsten wird ohne Steuer verbucht. Dies wird benötigt, falls die Steuer bei einer Buchung mit einem sonstigen Steuersatz in einer eigenen Buchung in der Importdatei steht. Dann muss dieses Feld auf "nein" gestellt werden, um die Steuerautomatik der EURO-FIBU außer Kraft zu setzen, damit die Steuer nicht zweimal gebucht wird.

## **4.5 Datum**

Bestandteil, mit dem Datumswerte erkannt und extrahiert werden können.

#### **Eingabefeld "Format"**

Im Eingabefeld "Format" kann festgelegt werden, in welcher Form das Datum vorliegt. Die Eigenschaften werden in Form eines Formatstrings definiert. Dabei werden mit den Buchstaben 't', 'm' und 'j' die Stellung der Datumsbestandteile und über Groß- und Kleinschreibung dieser Buchstaben evtl. feste oder variable Längen definiert. Ist einer dieser Buchstaben großgeschrieben, so steht dieser Buchstabe für eine feste Stellenanzahl des Wertes, d. h. "MM" würde bedeuten, dass der Monat zwei feste Stellen besitzt, hingegen "mm" würde eine Maximallänge von zwei Zeichen bedeuten. Ein amerikanisches Datum variabler Länge würde folgendermaßen definiert: "mM/tT/jJ" (jeweils mindestens ein Zeichen lange Bestandteile)

#### **Optionsschalter "Zuweisung an Datenfeld"**

Der Schalter "Zuweisung an Datenfeld" hat nur einen Sinn, wenn der Datumsbestandteil Kind oder Kindeskind eines Datenfeldbestandteils ist. Dieser

Schalter wird markiert, falls der Datumswert den Inhalt des Datenfeldbestandteiles festlegt. Der Schalter ist also wichtig für die eigentliche Datenübergabe an das Datenfeld.

Bei Datumsbestandteilen gibt es eine Besonderheit im Zusammenhang mit der Zuweisung insofern, als es möglich ist, unvollständige Daten zuzuweisen, d. h. Daten, die beispielsweise nur aus Tag und Monat bestehen. In diesem Fall werden nur die definierten Einzelteile (hier also nicht das Jahr) zugewiesen. Damit wäre es denkbar, das Datenfeld mit einem Jahreswert zu initialisieren, wobei dann Tag und Monat explizit verändert würden.

#### **Eingabefeld "String-Format"**

Im Eingabefeld "String-Format" kann ein Ausgabeformat für den erkannten Datumswert definiert werden. Dieses Ausgabeformat kommt nur dann zum Tragen, wenn der Schalter "Zuweisung an Datenfeld" markiert ist und das umgebende Datenfeld Texte (Strings) aufnimmt. In diesem Fall muss der Datumswert in eine Zeichenkette umgewandelt werden, wobei diese Umwandlung durch den Formatstring im Eingabefeld "String-Format" beeinflusst wird. Das Formatstring ist genauso definiert wie der für das Eingabefeld "Format". Wird beispielsweise "TT.MM.JJJJ" eingegeben, so wird die erkannte Zahl in ein langes Datum mit zweistelligem Tages- und Monatswert sowie vierstelligem Jahreswert umgewandelt, die jeweils durch Punkte getrennt sind.

#### **Auswahlliste "Variable"**

In der Auswahlliste "Variable" kann eine Variable angegeben werden, welcher der erkannte Zahlenwert zugewiesen werden soll. Diese Variable speichert den erkannten Text und kann an anderer Stelle ein Datenfeld initialisieren.

## **4.6 Kommentar**

Ein an beliebiger Stelle einfügbarer Bestandteil, der nur der Kommentierung der Beschreibungsdatei dient. Dieser Bestandteil wird beim Abarbeiten des Importformates nicht beachtet, sondern dient nur der besseren Übersichtlichkeit beim Bearbeiten des Importdateiformates. Sämtliche Kinderbestandteile eines Kommentarfeldes werden ignoriert.

#### **Eingabefeld "Text"**

In das Eingabefeld "Text" kann ein beliebiger Text eingegeben werden, der in der Baumanzeige ausgegeben wird.

## **4.7 Konstante**

Bestandteil, mit dem konstante Zeichenketten erkannt und extrahiert werden können.

#### **Eingabefeld "Zeichenkette"**

In das Eingabefeld "Zeichenkette" kann eine konstante Zeichenkette eingegeben werden, die genau in dieser Form in der Importdatei vorkommen muss. Die Zeichenkette kann auch nicht druckbare Zeichen enthalten: diese werden mit '\ ' und dem entsprechenden Zahlenwert des Zeichens dargestellt. Eine konstante Zeichenkette aus den Zeichen 31 und 't' würde lauten: "\31t".

#### **Auswahlliste "Trenner"**

In der Auswahlliste "Trenner" kann ein Trenner ausgewählt werden, der erkannt werden soll. Dieser wird nur dann beachtet, wenn im Eingabefeld "Zeichenkette" keine Eingabe gemacht wurde. Ist ein Trenner ausgewählt, so gilt die Zeichenkette genau dann erkannt, wenn die charakteristischen Zeichen

des Trenners gelesen werden konnten. Mit Auswahl eines konstanten Trenners werden also i.d.R. Trennzeichen wie Kommas oder Zeilenende überlesen.

#### **Eingabefeld "Wert"**

Im Eingabefeld "Wert" kann ein Wert ausgewählt werden, der bei Erkennung des Konstanten-Bestandteils an ein evtl. umgebendes Datenfeld zugewiesen wird. Zeichenketten werden in Anführungszeichen gestellt, Datumswerte enthalten zwei Dezimalpunkte (z. B. 17.4.87), wobei 0-Werte als nicht definiert gelten und entsprechend missachtet werden. Die konstante Zeichenkette <Systemdatum> bezeichnet das aktuelle Datum des Rechners, die konstante Zeichenkette <Geschäftsjahr> anstelle eines Jahreswertes steht für das aktuelle Geschäftsjahr des Mandanten. Zahlenwerte werden normal eingegeben (evtl. Vorzeichen, Vorkommastellen, evtl. Dezimalpunkt und Nachkommastellen). Falls das Datenfeld numerisch ist und sinnvolle vordefinierte konstante Werte besitzt, so befinden sich in der Auswahlliste die Konstanten in lesbarer Form.

#### **Optionsschalter "Zuweisung an Datenfeld"**

Der Schalter "Zuweisung an Datenfeld" hat nur einen Sinn, wenn dieser Konstantenbestandteil Kind oder Kindeskind eines Datenfeldbestandteils ist. Dieser Schalter wird markiert, falls der im Eingabefeld "Wert" angegebene konstante Wert den Inhalt des Datenfeldbestandteiles festlegt. Der Schalter ist also wichtig für die Zuweisung eines konstanten Wertes an das Datenfeld.

#### **Auswahlliste "Variable"**

In der Auswahlliste "Variable" kann eine Variable angegeben werden, welcher im Eingabefeld "Wert" angegebene Wert zugewiesen werden soll. Diese Variable speichert den Wert und kann an anderer Stelle ein Datenfeld initialisieren.

#### **Optionsschalter "Keine DOS-Konvertierung bei DOS-Dateien"**

Mit diesem Schalter kann eine Konvertierung des gelesenen Zeichens verhindert werden. Auf diese Weise kann bei DOS-Dateien jeder ASCII-Wert gelesen werden. Diese Option hat nur dann eine Auswirkung, wenn in der dazugehörigen Datei der Zeichensatz auf DOS eingestellt ist. Bei Dateien mit Windows-Zeichensatz werden generell keine Zeichen konvertiert.

## **4.8 Strukturierung**

Bestandteil, mit dem mehrfach vorkommende oder nicht unbedingt vorkommende Bestandteile beschrieben werden können. Eine Strukturierung bezieht sich stets auf eines oder mehrere Kinder. Damit ist es möglich, mehrfach vorkommende Bestandteile (z. B. Buchungssätze) zu charakterisieren. Außerdem kann auf optional vorkommende Bestandteile reagiert werden.

#### **Auswahlliste "Typ"**

Mit der Auswahl eines Eintrages in der Auswahlliste "Typ" wird festgelegt, in welcher Form die Kinder dieses Bestandteils auftreten. Die einzelnen Typen bedeuten (<Anzahl> steht für die Zahl im Eingabefeld "Anzahl"):

- 'genau einmal: alle nacheinander': die Kinder treten in der gegebenen Reihenfolge genau einmal nacheinander auf.
- 'genau <Anzahl> mal: alle nacheinander': die Kinder treten genau in der angegebenen Anzahl nacheinander auf.

- 'mindestens <Anzahl> mal: alle nacheinander': die Kinder treten mindestens in der angegebenen Anzahl nacheinander auf.
- 'höchstens <Anzahl> mal: alle nacheinander': die Kinder treten höchstens in der angegebenen Anzahl nacheinander auf.
- 'genau <Anzahl> aus': Es tritt genau Anzahl mal hintereinander eines der gegebenen Kinder auf. Dabei wird stets vom ersten ausgehend in der gegebenen Reihenfolge geprüft, ob ein Kind auftritt.
- 'mindestens <Anzahl> aus': Es tritt mindestens Anzahl mal hintereinander eines der gegebenen Kinder auf.
- 'höchstens <Anzahl> aus': Es tritt höchstens Anzahl mal hintereinander eines der gegebenen Kinder auf.

Ist die durch den Typ festgelegte Bedingung nicht erfüllt, so gilt dieser Strukturierungsbestandteil als nicht erkannt.

#### **Eingabefeld "Anzahl"**

In das Eingabefeld "Anzahl" kann - abhängig vom Typ der Strukturierung - die Anzahl eingegeben werden, die dann für die Vergleiche herangezogen wird.

#### **Eingabefeld "max. Länge"**

In das Eingabefeld "max. Länge" kann die Anzahl von Zeichen eingegeben werden, die die Kinderbestandteile in einer Importdatei belegen. Überschreitet die Größe der erkannten Kinderbestandteile die maximale Länge, so gilt dieser Strukturierungsbestandteil als nicht erkannt. Ansonsten wird unabhängig von der tatsächlichen Länge der Kinderbestandteile so getan, als hätten diese genau die gegebenen. Maximallänge belegt. Damit ist zum Beispiel möglich, Bestandteile aus einem Block fester Länge zu extrahieren, wobei danach genau am Ende des Blockes fortgesetzt wird.

#### **Optionsschalter "Zurücksetzen auf Beginn"**

Ist der Optionsschalter "Zurücksetzen auf Beginn" markiert, so wird bei erfolgreichem Erkennen der Kinderbestandteile die Position in der Datei so zurückgesetzt, als wäre von den Kindern nichts gelesen worden. Damit ist es möglich, Daten mehrfach zu lesen. VORSICHT: Falsch angewandt kann die Auswahl zu Endlosschleifen führen, da es passieren kann, dass nie etwas gelesen wird!

## **4.9 Zahl**

Bestandteil, mit dem variable Zahlen bestimmter Formate erkannt und extrahiert werden können.

#### **Eingabefeld "Format"**

Im Eingabefeld "Format" kann festgelegt werden, welche Eigenschaften die Zahl innerhalb der Importdatei erfüllen muss, um erkannt zu werden. Die Eigenschaften werden in Form eines Formatstrings wie folgt definiert:

Mit '0' wird eine Mindeststellenzahl vor bzw. nach dem Komma, mit '#' werden Höchststellenzahlen definiert. '.' setzt den Dezimalpunkt fest, kommt ein ',' vor, so wird eine Tausendertrennung erwartet. Beispielsweise bedeutet der Formatstring "###,###,#00.00#", dass die Zahl höchstens 9 Vorkomma- und 3 Nachkommastellen haben darf und mindestens 2 Vorkomma- und 2 Nachkommastellen besitzen muss. Mit dem Buchstaben 'v' wird die Stellung des Vorzeichens geregelt. Falls 'v' hinter dem Dezimalpunkt steht, so wird das Vorzeichen nach der Zahl erwartet. Mit den folgenden Buchstaben und einem nachfolgenden Buchstaben oder Buchstabenwert (angeführt durch '\') können

die einzelnen Zeichen festgelegt werden: 't' legt das Tausendertrennzeichen fest (Standardzeichen ist ','), 'd' legt den Dezimalpunkt fest (Standardzeichen ist '.'), 'm' legt das Minuszeichen fest (Standard ist '-'), 'p' legt das Pluszeichen fest (Standardzeichen ist '\0', gilt als leer) und 'z' legt die Kennung für die 0 fest (Standard ist ebenfalls '\0', also leer). Mit 'f' kann ein Füllzeichen festgelegt werden. Dies ist nur in Kombination mit '@' statt '#' sinnvoll.

So bedeutet beispielsweise "t.d.", dass der Tausendertrenner durch einen Punkt und der Dezimalpunkt durch ein Komma dargestellt wird. "m\0" bedeutet, dass negative Werte kein Vorzeichen haben.

#### **Basis-Variable**

Bei Zuweisung an ein Datenfeld oder eine andere Variable wird der Wert dieser Basis-Variablen dazu addiert. Auf diese Weise ist eine Aufspaltung des zuzuweisenden Wertes in Basis und Offset möglich.

#### **Optionsschalter "Zuweisung an Datenfeld"**

Der Schalter "Zuweisung an Datenfeld" hat nur einen Sinn, falls dieser Zahlbestandteil Kind oder Kindeskind eines Datenfeldbestandteils ist. Dieser Schalter wird markiert, falls der Zahlenwert den Wert des Datenfeldbestandteiles festlegt. Der Schalter ist also wichtig für die eigentliche Datenübergabe an das Datenfeld.

#### **Eingabefeld "String-Format"**

Im Eingabefeld "String-Format" kann ein Ausgabeformat für den erkannten Zahlenwert definiert werden. Dieses Ausgabeformat kommt nur dann zum Tragen, wenn der Schalter "Zuweisung an Datenfeld" markiert ist und das umgebende Datenfeld Texte (Strings) aufnimmt. In diesem Fall muss der Zahlenwert in eine Zeichenkette umgewandelt werden, wobei diese Umwandlung durch den Formatstring im Eingabefeld "String-Format" beeinflusst wird. Das Formatstring ist genauso definiert wie der für das Eingabefeld "Format". Wird beispielsweise "0000" eingegeben, so wird die erkannte Zahl in einen mindestens vierstelligen Zahlenstring umgewandelt, der keine Nachkommastellen besitzt.

#### **Auswahlliste "Variable"**

In der Auswahlliste "Variable" kann eine Variable angegeben werden, welcher der erkannte Zahlenwert zugewiesen werden soll. Diese Variable speichert den Zahlenwert und kann an anderer Stelle ein Datenfeld initialisieren.

#### **Als Zeichenkette**

Mit Hilfe dieser Option ist eine Konvertierung vom Typ Zahl in den Typ Text möglich, der ausgewählten Variable wird dann der formatierte Text zugewiesen. (Hinweis: bei Zuweisung an ein Datenfeld entscheidet dieses über die Art der Konvertierung.)

## **4.10 Zeichenkette**

Bestandteil, mit dem variable Zeichenketten erkannt und extrahiert werden können.

#### **Eingabefeld "feste Länge"**

Im Eingabefeld "feste Länge" kann eine Zeichenkette fester Länge extrahiert werden (Die Länge entspricht der Anzahl Zeichen). Sind andere Bedingungen gegeben, so muss die Länge der Zeichenkette, die diese Bedingungen erfüllt, genau der vorgegebenen Länge entsprechen.

**Zeichenfeld "erlaubte Zeichen"**

Das Zeichenfeld "erlaubte Zeichen" enthält alle Zeichen, die der Text enthalten darf. Die erlaubten Zeichen werden über die Schaltfläche "...", die hinter dem Zeichenfeld steht, ausgewählt.

**Zeichenfeld "Anführ.zeichen"**

Im Zeichenfeld "Anführ.zeichen" kann ein den Text umgebendes Anführungszeichen definiert werden. Dazu wird der Buchstabe für das Anführungszeichen eingegeben oder der mit '\' vorangehende Zahlenwert des Zeichens. Die Anführungszeichen werden nicht in den Text übernommen und nicht zur Länge dazugezählt.

**Auswahlliste "Trenner"**

In der Auswahlliste "Trenner" kann ein Trenner definiert werden, der die Zeichenkette auf jeden Fall beendet. So kann durch Definition und Auswahl eines geeigneten Trenners in diesem Feld eine Zeichenkette variabler Länge eingelesen werden, die durch eine genau definierte Auswahl von Zeichen beendet wird. VORSICHT: der Trenner selbst wird nicht überlesen, sondern dient nur als Ende-Markierung der Zeichenkette. Die nachfolgenden Trennzeichen müssen also evtl. durch ein Konstanten-Bestandteil des Typs Trenner überlesen werden! Trenner können auf der Eigenschaftenseite des Imports definiert und verwaltet werden.

**Escape**

Hier wird ein Escape-Zeichen definiert, d.h. das nachfolgende Zeichen wird als Text übernommen (auch bzw. besonders, wenn es das Trennzeichen oder auch Escape-Zeichen ist). Ist z.B. das Hochkomma ein Trennzeichen, das Fragezeichen ein Escapezeichen, und das Trennzeichen kann aber mit Escapezeichen im Text vorkommen, so würde '?' zu ' ersetzt und kein Trennzeichen erkannt. Die Folge ?? wird so zu ? verkürzt. Das Escapezeichen ist also vergleichbar wie in der Sprache „C“ das Zeichen „\“ im Compiler oder % bei Ausgaben mittels printf().

Escape-Zeichen werden wie Trennzeichen in der Eigenschaftenseite des Imports definiert und stehen nach Übernahme in der entsprechenden Liste zur Auswahl bereit.

**Keine leere Kette**

Durch dieses Kontrollkästchen wird festgelegt, dass die Zeichenkette mindestens ein Zeichen enthalten muss. Auf diese Weise können Pflichtfelder markiert werden.

**Optionsschalter "Zuweisung an Datenfeld"**

Der Schalter "Zuweisung an Datenfeld" hat nur einen Sinn, wenn er Zeichenkettenbestandteil Kind oder Kindeskind eines Datenfeldbestandteils ist. Dieser Schalter wird markiert, falls der Text den Inhalt des Datenfeldbestandteiles festlegt. Der Schalter ist also wichtig für die eigentliche Datenübergabe an das Datenfeld.

**Auswahlliste "Variable"**

In der Auswahlliste "Variable" kann eine Variable angegeben werden, welcher der erkannte Zahlenwert zugewiesen werden soll. Diese Variable speichert den erkannten Text und kann an anderer Stelle ein Datenfeld initialisieren.

**Auswahlliste "Trenner"**

In dieser Auswahlliste kann ein Trenner angegeben werden, an den die Zeichenkette zugewiesen wird. Anschließend sind alle diese Zeichen

Trennzeichen. Zeichen können auch mehrfach vorkommen. Es wird zwischen Groß- und Kleinschreibung unterschieden.

#### Optionsschalter "add. " (additiv)

Mit dieser Option wird festgelegt, dass die zugewiesenen Trennzeichen als zusätzliche Trenner interpretiert werden. Ist diese Option nicht gesetzt, so werden die bisherigen Trennzeichen gelöscht und nur die neuen übernommen.

### 4.10.1 Zeichenmengeneigenschaften

Dieser Dialog wird in der Eigenschaftenseite der Zeichenkette über die Schaltfläche „...“ hinter dem Feld „erlaubte Zeichen“ aufgerufen.

Eine Zeichenmenge definiert eine Gruppe erlaubter Zeichen. Im Zeichenmengeneigenschaftsdialog können Gruppen erlaubter Zeichen ausgewählt werden.

#### Liste "erlaubte Zeichenklassen"

Die Liste "erlaubte Zeichenklassen" enthält eine Sammlung vorgegebener Zeichengruppen, die durch Markierung als erlaubt definiert werden können. Es können mehrere Gruppen von Zeichen gleichzeitig markiert werden, die dann alle als erlaubt gelten.

#### Eingabefeld "zusätzliche Zeichen"

Im Eingabefeld "zusätzliche Zeichen" können beliebige Zeichen eingegeben werden, die als erlaubt angesehen werden. Ist das Zeichen nicht darstellbar, so kann es durch '\n' angegeben werden, wobei n die Buchstabennummer ist. Das Zeilenrücklaufzeichen <CR> könnte also als \13 eingegeben werden, obwohl es auch als vordefinierte Gruppe in der Liste "erlaubte Zeichenklassen" zu finden ist.

## 5 Beispiel

Das folgende Beispiel soll die Beschreibung einer Importdatei als Struktur verstehen helfen. Die Buchungen liegen dabei in einer von einem DOS-Programm erzeugten Datei BUCH.TXT in folgender Form vor:

B: (die Datei beginnt mit einem Prolog)

dann kommen die einzelnen Datensätze (jeweils eingeleitet durch ein 'n'):

n<Belegnr>s<Sollkto.>h<Habenkto.>b<Bruttober.>t<Text>:[p<Steuersatz>].

die Datei endet mit einer anderen Kennung:

e

Also könnte eine Importdatei 'BUCH.TXT' mit 2 Buchungen folgendermaßen aussehen:

B:n129s1200h10000b600.00tText1;p0.00.n130s1200h10001b800.00tText2;.e

Vor dem Start eines Importlaufes müssen die Art der zu importierenden Daten sowie eine Datei zum Importieren ausgewählt werden. Um also die Daten der obigen Datei zu importieren, würde als Importdatei 'bube.txt' gewählt werden und als Datenart 'Buchungsbelege' oder 'Stapelbuchungen'.

Oberster Bestandteil unseres Beschreibungsformates ist der Importbestandteil. In den Eigenschaften des Importbestandteiles legen wir den Titel des Formates sowie den Titel der Fehlerdatei fest. Im Titelfeld des Eigenschaftsdialoges wird also der beliebig zu vergebende Titel eingegeben (z. B. 'Buchungsimport'). In

den Feldern zur Fehlerdatei kann der Name der anzulegenden Fehlerdatei eingegeben werden; Soll die Fehlerdatei 'BUCH.ERR' heißen; und nehmen wir 'BUCH.TXT' als vorgegeben an, so reicht es aus, im Extension-Feld '.ERR' einzutragen.

Alle zu importierenden Daten befinden sich genau in der vorgegebenen Datei 'BUCH.TXT'; unser Import besteht also aus dem Bearbeiten genau einer Datei. Diese Besteht-Aus-Beziehung schlägt sich als Eltern-Kind-Beziehung nieder: Die einzige Importdatei 'BUCH.TXT' ist Kind des Importbestandteiles. In den Eigenschaften der Datei kann ein evtl. abweichender Dateiname definiert werden. Da wir den Dateinamen 'BUCH.TXT' als vorgegeben betrachten, muss hier nichts eingegeben werden. Eine wichtige Eigenschaft dieser Datei ist, dass sie von einem DOS-Programm erzeugt wurde. Also wird als Zeichensatz 'DOS' ausgewählt, damit die Umlaute korrekt übersetzt werden.

Die Beschreibung des Dateiformates unserer Importdatei wird in Form von Kinderbestandteilen formuliert: Zunächst muss der Prolog beschrieben werden. Da dieser im gegebenen Format stets als konstanter Text 'B:' vorkommt, wird ein Konstanten-Bestandteil als erstes Kind des Dateibestandteiles angelegt: Dieser konstante Bestandteil erhält als Zeichenkette genau den Prolog, also 'B:'.

Direkt nach dem Prolog folgen die Buchungen (unbekannte Anzahl). Da mehrere Buchungen vorkommen können, benötigen wir jetzt einen Bestandteil, das aussagt, dass mindestens einmal, aber evtl. auch öfter hintereinander die gleichen Bestandteile kommen. Diese Aufgabe übernehmen die Strukturierungsbestandteile, die abhängig von ihrem Typ verschiedene Arten von Strukturen darstellen können. Diese Bestandteile sind nur sinnvoll, wenn sie Kinderbestandteile besitzen. In unserem Beispiel sind diese Kinderbestandteile genau die Beschreibungsbestandteile einer Buchung. Der Strukturierungsbestandteil hat den Typ 'mindestens <Anzahl> mal: alle hintereinander' (mit Anzahl = 1), das genau aussagt, dass nun mehrmals hintereinander die Kinderbestandteile in fester Reihenfolge auftreten. Das Ende der Importdatei ist wie der Prolog fest vorgegeben und besteht aus einem 'e'. Genau wie beim Vorspann wird also nach unserem Strukturierungsbestandteil wieder ein konstanter Bestandteil eingefügt, das als Zeichenkette den Text 'e' enthält.

Schauen wir nun die Kinderbestandteile der Importdatei an: es sind genau drei Teile, die unsere Importdatei wie oben beschrieben gliedern (Prolog, Buchungen, Endekennung). Das genaue Aussehen der Buchungen wird mittels der Kinderbestandteile des zweiten Teiles ausgedrückt (wieder die Besteht-Aus-Beziehung in Form der Eltern-Kind-Beziehung).

Um nun eine Buchung von der nächsten unterscheiden zu können, müssen deren Grenzen klar definiert werden. Es muss also irgendwie beschrieben werden können, wo eine Buchung beginnt und wo sie endet. Diese Aufgabe übernimmt der Datensatz-Bestandteil: Damit wird ausgedrückt, dass alle Kinderbestandteile dieses Teils genau einen Datensatz beschreiben (in unserem Fall eine Buchung). Also ist das einzige Kind des oben beschriebenen Strukturierungsbestandteiles ein Datensatzteil, wobei als Datensatz 'Buchung' gewählt wird. Wieder muss beschrieben werden, aus welchen Bestandteilen sich eine Buchung zusammensetzt, was wie immer durch die Eltern-Kind-Beziehung ausgedrückt wird: Die Einzelbestandteile der Buchung werden durch ihre Kinderbestandteile ausgedrückt.

Erster Kinderbestandteil ist das Belegnummern-Datenfeld, das durch die Kennung 'n' eingeleitet wird. Wir modellieren dies dadurch, dass der erste Kinderbestandteil des Datensatzes ein Datenfeldteil wird, wobei als Datensatz 'Buchung' und als Datenfeld 'Belegnummer' ausgewählt wird. Das Format, wie die Belegnummer in unserer Importdatei abgelegt ist, wird durch die Kinderbestandteile des Datenfeldes beschrieben: Erstes Kind ist also ein Konstantenfeld, das als Zeichenkette 'n' enthält. Ihm folgt ein Zahlbestandteil, denn unsere Belegnummer wird als Zahl hinter der Zeichenkette geführt. Dieser Zahlbestandteil beschreibt die Textform, in der eine Zahl in der Dateiform abgelegt ist.

Nach der Belegnummer folgt die Soll-Kontonummer. Dies drücken wir dadurch aus, dass dem Datenfeldbestandteil des Typs Buchung.Belegnummer ein weiterer Datenfeldbestandteil (allerdings des Typs Buchung.Sollkontonummer) folgt - er wird also zweiter Bestandteil des darüberliegenden Datensatzteils 'Buchung'. Diesem zweiten Bestandteil folgt dann der Bestandteil Buchung.Habenkontonummer, Buchung.Bruttobetrag, Buchung.Text und ein Abschnitt zum Steuersatz, der ja optional sein kann.

In diesem Fall kann man nicht einfach analog zu den vorherigen Feldern das Datenfeld Buchung.Steuersatz als Schwester zum vorherigen Datenfeld Buchung.Text einfügen, da in diesem Falle ja die Bedingung, dass alle Felder in der vorgegebenen Reihenfolge in der Importdatei erscheinen ja nur erfüllt wäre, wenn ein Steuersatz angegeben würde (der aber nicht angegeben werden muss!). Abhilfe schafft ein Strukturierungsbestandteil des Typs 'höchstens <Anzahl> mal: nacheinander' mit Anzahl = 1, der als einziges Kind das Datenfeld Buchung.Steuersatz enthält, welches wiederum aus dem Konstantenbestandteil 'p' und Zahlenbestandteil besteht. Tritt nun kein Steuersatz in der Importdatei auf, so kann das Datenfeld Steuersatz nicht erkannt werden, was aber durchaus korrekt ist, da das umgebende Strukturierungsfeld aussagt, dass dies nicht vorkommen muss, sondern kann.

Wieder spiegelt unser Baum nichts anderes wider als die Besteht-Aus-Beziehung:

Der Import setzt sich zusammen aus

der Importdatei 'BUCH.TXT', bestehend aus

Prolog 'B:'

mindestens einer Buchung, bestehend aus

Datenfeld Belegnummer, bestehend aus

'n'

eigentlicher Nummer

Datenfeld Sollkontonummer, bestehend aus

's'

eigentliche Nummer

Datenfeld Habenkontonummer, bestehend aus

'h'

eigentliche Nummer

Datenfeld Bruttobetrag, bestehend aus

'b'

eigentlicher Betrag

Datenfeld Text, bestehend aus

't'

eigentliche Zeichenkette (Trenner: „;“)

'.'

höchstens einmal

Datenfeld Steuersatz, bestehend aus

'p'

Steuersatz

'.' (beendet eine Buchung)

Endekennung 'e'

Im Lieferumfang der EURO-FIBU sind Importdateien für Kreditoren, Debitoren, Sachkonten, Kostenstellen, Buchungsbelege, Stapelbuchungen und Überweisungen standardmäßig enthalten (Sie finden diese im Programmverzeichnis der EURO-FIBU als \*.FID). So können Sie auf vorhandenen Beschreibungen aufbauen statt diese selbst entwickeln zu müssen.

**syska.**

Gesellschaft für betriebliche Datenverarbeitung mbH

Am Sandfeld 15

76149 Karlsruhe

Tel.: +49 (0)721 985 93-0

Fax: +49 (0)721 985 93-60

[info@syska.de](mailto:info@syska.de)

[www.syska.de](http://www.syska.de)