Description standard for data media transfer
Version 1.1 of 1 August 2002

Explanations concerning the storage and description of data within the scope of the "Rules for Data Access and Verifiability of Digital Documentation (GDPdU)"

Description standard – Version 1.1 of 1 August 2002

Change	Date	Author(s)
Description standard – Version1.0	6/6/2002	Jon Chick
Description standard – Version 1.1	1/8/2002	Jon Chick

Document version 1.1 of 1 August 2002

Change	Date	Author(s)
Document version 1.0	6/6/2002	Jon Chick, Tankred Giese, Martin Otto
Document version 1.1 Explanations regarding references Expansions version 1.1	1/8/2002	Martin Otto, Jon Chick, Tankred Giese, Michael Schleupen

Table of contents

Target audience for this paper

- Users of software instructed to provide a data medium pursuant to GDPdU.
- Developers of software providing tax relevant data for data media transfer pursuant to GDPdU.

Summary/objective

Pursuant to §147 par. 6 AO, internal revenue is entitled to audit the data of electronic accounting systems "digitally" either through the transfer of data media and/or through indirect or alternatively direct access.

For the purposes of transferring data media, the data must be provided in a "machine-analysable form" by the business subject to taxation (or by the commissioned tax advisor, accounting firm/subcontractor etc.) on suitable data media.

The term "machine analysability" from the perspective of internal revenue refers to optional access to all stored data, including the master data and links with sorting and filtering functions.

In order to achieve such analysability and usability, it is necessary to define and standardise the file formats for data media transfer.

This document presents a procedure to facilitate the defined and standardised data media transfer and to outline a standard for the software industry.

Should you have any queries, please write to mailto:GDPdU@audicon.net, or obtain information from http://www.audicon.net.

Text of §147 par. 6 AO

In the version of the act to lower tax rates and to reform business taxation (Tax Reduction Act – StSenkG) of 23/10/2000, BGBl. (Federal Gazette) I 2000, p. 1433]

(6) In the event that documentation has been prepared pursuant to par. 1 with the aid of a data processing system, internal revenue, within the scope of an external audit, has the right to view stored data and to use the data processing system to audit such documentation. Within the scope of an external audit, internal revenue can also require that the data be analysed automatically as specified by it, or that the stored data and copies be made available to it on a machine-utilisable data medium. The costs involved are borne by the party subject to taxation.

Data media transfer

a possible scenario for a tax audit involving a data media transfer:

An auditor is commissioned to audit a business. For specific auditing fields and auditing periods, the auditor decides whether to analyse the data with the aid of auditing software. Accordingly, the business concerned will be provided with notice of the audit and will be notified of the areas on which the audit will concentrate. He or she requests the business to make the tax relevant data available on a machine-analysable data medium as specified by the "Rules for Data Access and Verifiability of Digital Documentation" (Federal Ministry of Finance communication of 16/07/2001).

Upon commencement of the audit, the auditor receives a CD ROM as a data medium.

In line with the focus of auditing, the auditor searches for the relevant tables and then imports these. The CD ROM contains tax relevant data and descriptive data to allow the data to be imported without further clarifications. As soon as the data have been imported into the analysis software, the auditor carries out various analyses (in part also automated by means of macros).

[Abb.]

1. Requests data CD ROM
2. Imports from CD ROM
3. Carries out analysis or runs analysis macros

Auditor

From the perspective of the business, the scenario described above involves the following work steps:

An employee initiates the extraction of data or exporting from the production database, during which process data are potentially (automatically) transformed in the sense of changing the way the data are organised, the structure of the tables or the structure of the objects.

Similarly, in the case of compressed or encrypted data, the compression or encryption must be removed. It is then additionally necessary to create or copy a description in the form of a machine-analysable file to accompany the tax relevant data. A file of this type must be created during the process of provision in a case where the software manufacturer is unable to foresee which data are to be provided during the audit. It is possible to provide a static file if the scope and structure of the data are certain beforehand.

The data are subsequently saved automatically or manually to a portable data medium (CD ROM, DVD...).

[Abb.]

Organisation of data during data media transfer

The Federal Ministry of Finance communication of 16/07/2001 as a matter of principle provides audited companies substantial leeway for the technical organisation of the data medium to be transferred.

After the tax authorities of the federation and federal states have procured analysis software in common use for auditing (IDEA 2002), the auditors are in a position to read and process data mechanically (automated or manually). The financial authority does not define the data fields and contents saved onto the data medium during data media transfer. It is therefore necessary that parties subject to taxation and/or manufacturers of software who process tax relevant data decide themselves which data are to be provided during the transfer of data media. Depending on the individual EDP system, software structures, business structure and size, a variety of data contents may here be sensible and necessary.

In order to minimise queries by auditors regarding the content of data and formats or to render these unnecessary, a procedure is described here which describes and comments on the data, data structures and links.

In order to be able to process the various data structures, the party subject to taxation or his/her advisor makes the tax-relevant data available in a largely denormalised form. Additionally, he/she provides a machine-analysable description of the data, data formats and links. The file format for the tax relevant data has been standardised to common default formats. Both tax relevant data and description data are made available on a common data medium.

[Abb.]

The supported file formats for tax relevant data are:

- VariableLength
- FixedLength

In this way, it is possible for the software systems to use pre-existing export filters or storage mechanisms.

Internal revenue does not permit any installation of special software in order to decrypt or uncompress data on its systems. However, during conventional mailing of data media, at minimum, the encryption of data is required for data protection reasons. The description standard therefore, for example, provides for programmes for decryption or uncompressing to be started directly from the data medium without these installing themselves on the internal revenue system.

The computers of the auditors are equipped with access protection software and where applicable also with anti-virus software. The functionality of these directly-accessed programmes should therefore be clarified by the data provider in advance. Testing whether these routines can be executed does not form part of this description standard.

The role of the software manufacturer

In what follows, a "software manufacturer" – from the perspective of the business subject to taxation – is assumed to refer to a supplier of software or a data processing system, including programmes, or an internal department that makes relevant programmes or services available.

In line with the outlined scenarios for auditing, amongst others, a machine-analysable description of data and links needs to be made available during auditing.

The description file is read by a programme that, in line with this description, controls the import function of the analysis programme of the tax auditor. No operations are carried out on databases of the productive systems.

The manufacturer of the programmes or procedures by means of which the business subject to taxation processes tax relevant data (software manufacturer) must in addition to extraction functionality ensure the formatting or provision of the description functionality.

XML-based description standard

Overview

XML stands for "eXtensible Markup Language". XML only prescribes a generic set of language elements, so that an individual language needs to be defined based on each individual instance of application. The in each instance newly-formulated language can be stored in a machine-readable form by means of structure information and element definitions in a DTD (Document Type Definition). In this way, an XML parser can check whether an XML document corresponds to the agreed language (validation). A distinction is therefore drawn between the language description in the DTD and the contents in the XML instance.

Within the DTD itself, no descriptive data are therefore stored for the data media transfer, but instead in the XML file called INDEX.XML. The DTD describes the structure of the INDEX.XML file.

Additional information regarding XML is available at www.w3c.org

General assumptions or alternatively requirements for the DTD

In the DTD, it is assumed that the tax relevant data are organised into files. Per table, exactly one file is assumed, so that organisation according to file and table coincide physically.

The descriptive data for the data media transfer are saved into a file called INDEX.XML.

The DTD needs to be located in the same directory on the data medium as the INDEX.XML file.

In the event that more than one data medium (media) is provided, the INDEX.XML must only be located on the first data medium.

Technical sequence of the import process

The import process is controlled by a component that interprets the INDEX.XML file and transfers the export tables from the individual data media to the analysis software. This component checks the INDEX.XML file for correctness and serves as a table of contents for the delivered data for the auditor.

[Abb.]

The XML DTD

[Abb.]

[Abb.]

[Abb.]

Organisation of the description standard

[Abb.]

The basic element of the description standard is the DataSet element. This contains the elements of Version, DataSupplier, Command and Media.

- Version contains the version number of the data media provision.
- DataSupplier contains details concerning the origin of data.
- The media containers contain details concerning the provided data media and tables stored on these. Here, for each data medium (CD ROM) provided, an individual media entry is carried out. It is therefore also possible to combine a number of data media (CD ROMs) within a DataSet.

[Abb.]

[Abb.]

[Abb.]

Notes regarding XML

- Note on operators common in XML:

Operator	Element type may occur
?	0 times or 1 time
*	0 times or as often as desired
+	1 time or as often as desired

- #PCDATA stands for parsed character data and means that the element is provided for a text entry.

- In the case of enumerations, the possible values are separated by means of vertical lines ("\|").

- XML is case sensitive.

- Please do not use any of the following characters in the descriptive fields and/or the table or column names:

" & < >

Instead use names or numbers from the list below:

Character	Name	Number
"	&quot;	&#34;
&	&amp;	&#38;
<	&lt;	&#60;
>	&gt;	&#62;

Description of elements

What follows is a description of the individual ELEMENT declarations.

What generally applies is that the texts stored in the relevant "description" attributes are displayed to the auditor as commentary. The length of the text in the description attributes should not exceed 255 characters.

<!ELEMENT Version (#PCDATA)>
Contains the version number of the data media provision. This element has no technical effect but serves as a description.

Used in container
DataSet

<!ELEMENT Location (#PCDATA)>
Describes the location of the data supplier.

Used in container
DataSupplier

<!ELEMENT Comment (#PCDATA)>
Comment field for additional information concerning the data supplier.

Used in container
DataSupplier

<!ELEMENT Length (#PCDATA)>
contains information concerning the length of attributes or datasets.

Used in container
Range

<!ELEMENT References (#PCDATA)>
contains information concerning links

Used in container	
ForeignKey	referenced table

In this regard also see example on page 39.

<!ELEMENT From (#PCDATA)>
contains information concerning the starting value of a value range.

Used in container	Context-dependent description
Range	Starting value within a value range
Map	Redefinition of data

<!ELEMENT To (#PCDATA)>
contains information concerning the end value of a value range.

Used in container	Context-dependent description
Range	End value within a value range
Map	Redefinition of data

<!ELEMENT MaxLength (#PCDATA)>
contains information concerning the maximum length of an alpha-numeric field in a VariableLength table.

Used in container	Context-dependent description
VariableColumn	Maximum length of the field
VariablePrimaryKey	Maximum length of the field

The specification of MaxLength for the VariableLength file type speeds up the import process, because otherwise the file would be tested before the import process and the maximum lengths of fields would be determined.

<!ELEMENT TextEncapsulator (#PCDATA)>
In the case of VariableLength files, it is possible to encapsulate text fields using an "Encapsulator" character, e.g. in case the field separator occurs in the data.

Standard value: “
Used in container
VariableLength

<!ELEMENT Accuracy (#PCDATA)>
Number of decimal places. Be careful with data the accuracy of which is more precise than specified under Accuracy.

Standard value: 0
Used in container
Numeric

<!ELEMENT ImpliedAccuracy (#PCDATA)>
Number of decimal places, even when no decimal places are specified in the source data.

Example:
Source data: "100; 200; 102; 5000;6587890"
ImpliedAccuracy 3: "0,100; 0,200; 0,102; 5,000; 6587,890"

Used in container
Numeric

<!ELEMENT Format (#PCDATA)>
Details concerning format definitions.

Standard value:	DD.MM.YYYY

Used in container	Context-dependent description
Date	This is where the mask for the representation of the date details is stored. The following specifications for the format are valid: o DD for day, o MM for month, o YY or alternatively YYYY for year. • Example (30 September 2001) o MM/DD/YY: 09/30/01 o DD.MM.YY: 30.09.01 o MM/DD/YYYY: 09/30/2001 o YYYY-MM-DD: 2001-09-30 o DDMMYY: 300901 o DDMMYYYY: 30092001
Validity	Masking for the date format used (see Date)

<!ELEMENT DecimalSymbol (#PCDATA)>
Specification concerning the decimal place separator
Standard value: ,
Used in container
Table

<!ELEMENT DigitGroupingSymbol (#PCDATA)>
Defines the separator for thousands.

Standard value: .
Used in container
Table

<!ELEMENT Command (#PCDATA)>
Command defines an operating system command. These can be carried out at the following points:

- prior to the overall import process
- after the overall import process
- prior to an individual data medium being imported
- after an individual data medium being imported

Used in container	Context-dependent description
DataSet	Command for the overall process
Media	Command for an individual data medium

In this regard also see example on page 41.

<!ELEMENT URL (#PCDATA)>
This element contains the physical file name of the file provided.

In this version of the description standard, relative URLs are supported exclusively. The path to the INDEX.XML file serves as the base directory.

Examples of <u>invalid</u> entries:
http://www.somewhere.com/data/Accounts.csv
ftp://ftp.somewhere.com/data/Accounts.csv

Examplesof <u>valid</u> entries:
Accounts.dat
data/Accounts.dat
data/january/Accounts.dat
./Accounts.dat

Used in container
Table

<!ELEMENT Description (#PCDATA)>
Description field

Used in container	Context-dependent description
FixedColumn	Description field for columns in FixedLength files
FixedPrimaryKey	Description field for keys in FixedLength files
VariableColumn	Description field for columns in VariableLength files
VariablePrimaryKey	Description field for keys in VariableLength files
Table	Description field for tables
Map	Description field for redefinition of data

<!ELEMENT Name (#PCDATA)>
Description field.

Standard value:	Specification from the URL element

Used in container	Context-dependent description
FixedColumn	Column name in FixedLength files
FixedPrimaryKey	Key name in FixedLength files
VariableColumn	Column name in VariableLength files
VariablePrimaryKey	Key name in VariableLength files
DataSupplier	Name of the data supplier (company name,...)
Media	Data media name
URL	Physical table name
ForeignKey	Name of the foreign key

<!ELEMENT Epoch (#PCDATA)>
For dates with a two-digit year format, it is possible to specify a standard placeholder that separates 19xx from 20xx. This value is pre-populated with 30.

In some files, the date 25 November 55 can mean either 1955 or 2055. Using Epoch, it is specified up to which two-digit value the year is to be interpreted as referring to the next century. For instance, if the century separation value field contains the value 50, all year indications from 0 to 49 will be interpreted as the years 2000 to 2049, and the year indications 50 to 99 as 1950 to 1999.

Standard value: 30
Used in container
Table

<!ELEMENT ColumnDelimiter(#PCDATA)>
Defines the separator for data fields.

Standard value:	&#59; (semicolon ";")

Used in container	Context-dependent description
VariableLength	Data field separator

Example of conventional separators:

Separators	XML notation
Comma ","	&#44;
Semicolon ";"	&#59;
Tab	&#9;

<!ELEMENT RecordDelimiter (#PCDATA)>
Defines separators for datasets.

Standard value:	&#13;&#10; (CRLF)

Used in container	Context-dependent description
FixedLength	Dataset separator
VariableLength	Dataset separator

Example of conventional separators:

Separators	XML notation
CR Carriage Return	&#13;
CRLF	&#13;&#10;
LF Linefeed	&#10;

<!ELEMENT SkipNumBytes (#PCDATA)>

If a file contains undesirable data at the beginning, e.g. file header, it needs to be specified here how many bytes are to be skipped and not read.

Standard value:	0

Used in container	Context-dependent description
Table	

<!ELEMENT Range (From, (To | Length)?)>

Defines a range or a starting value.
Starting value means: if no *To* or *Length* is specified, this range only defines a starting value (e.g. a dataset position).

Used in container	Context-dependent description
Table	Start dataset
Validity	Validity range

<!ELEMENT FixedRange (From, (To | Length))>

Defines a range specifying *From* and *To* or *From* and *Length*.

Used in container	Context-dependent description
FixedColumn	Field position in FixedLength files
FixedPrimaryKey	Key position in FixedLength files

<!ELEMENT DataSet (Version, DataSupplier?, Command*, Media+, Command*)>

The *DataSet* is the upper element (document element) in the hierarchy of the XML document. The element *DataSet* is the medium for the version, data origin, preceding and subsequent processes and the media containing the tables.

<!ELEMENT AlphaNumeric EMPTY>
Defines the data type as alpha-numeric.

Used in container	Context-dependent description
FixedColumn	Type of field in the FixedLength file
FixedPrimaryKey	Type of key in the FixedLength file
VariableColumn	Type of field in the VariableLength files
VariablePrimaryKey	Type of key in the VariableLength files

<!ELEMENT Date (Format?)>
Defines the data type as date.

Used in container	Context-dependent description
FixedColumn	Type of field in the FixedLength file
FixedPrimaryKey	Type of key in the FixedLength file
VariableColumn	Type of field in VariableLength files
VariablePrimaryKey	Type of key in VariableLength files

<!ELEMENT Numeric ((ImpliedAccuracy | Accuracy)?)>
Defines the data type as numeric.

Used in container	Context-dependent description
FixedColumn	Type of field in FixedLength file
FixedPrimaryKey	Type of key in FixedLength file
VariableColumn	Type of field in VariableLength files
VariablePrimaryKey	Type of key in VariableLength files

<!ELEMENT ANSI EMPTY>
Specifies the use of the ANSI code page.

<!ELEMENT Macintosh EMPTY>
Specifies the use of the Macintosh code page.

<!ELEMENT OEM EMPTY>
Specifies the use of the IBM-PC-ASCII code page.

<!ELEMENT UTF16 EMPTY>
Specifies the use of the UTF16 code page.

<!ELEMENT UTF7 EMPTY>
Specifies the use of the UTF7 code page.

<!ELEMENT UTF8 EMPTY>
Specifies the use of the UTF8 code page.

Standard value:	ANSI

Platform	Character set
Unix computers	7-bit ASCII character set (7-bit ISO code, US variant), only characters with ASCII code below 128, in other words letters, numbers and special characters (full-stop, comma, etc.) without country-specific special characters (umlauts, etc.) and without graphics characters. Line break is LF (Line Feed), character 10.
PC under DOS	8-bit "IBM PC ASCII" character set that in terms of the lower 127 characters corresponds to 7-bit ASCII and from character 128 contains country-specific special characters and (semi-)graphics characters. Line break is the sequence CR and LF, representing characters 12 and 10 (Carriage Return, Line Feed).
PC under Windows	8-bit ANSI character set, which up to 127 corresponds to ASCII and above 127 contains country-specific special characters that do not correspond to those of IBM PC ASCII. Line breaks, as under DOS, the sequence CR, LF. (Special applications are also able to use a 16-bit code, Unicode.)
Apple Macintosh	Macintosh-specific "Mac-ASCII" 8-bit character set, which up to 127 corresponds to ASCII and above 127 contains country-specific special characters. Line break is CR.

The selection of code page does not set a default for the RecordDelimiter. You will always have to specify the RecordDelimiter explicitly if the standard value (CRLF) is not suitable.

<!ELEMENT FixedLength ((Length | RecordDelimiter)?, ((FixedPrimaryKey+, FixedColumn*) | (FixedColumn+)), ForeignKey*)>
Defines a FixedLength type file.

Used in container
Table

<!ELEMENT FixedColumn (Name, Description?, (Numeric | AlphaNumeric | Date), Map*, FixedRange)>
Defines a column (=column) in a FixedLength type file.

Used in container
FixedLength

<!ELEMENT FixedPrimaryKey (Name, Description?, (Numeric | AlphaNumeric | Date), Map*, Range)>
Defines a key field (=key) in a FixedLength type file.

Used in container
FixedLength

In this regard also see example on page 39.

<!ELEMENT VariableLength (ColumnDelimiter?, RecordDelimiter?, TextEncapsulator?, ((VariablePrimaryKey+, VariableColumn*) | (VariableColumn+))?, ForeignKey*)>
Defines a VariableLength type file.

Used in container
Table

The fields (VariableColumn, VariablePrimaryKey) must be defined in the sequence in which they are delivered in the file.

<!ELEMENT VariableColumn (Name, Description?, (Numeric | AlphaNumeric, MaxLength ? | Date), Map*)>
Defines a column (=column) in a variable length type file.

Used in container
VariableLength

The fields (VariableColumn, VariablePrimaryKey) must be defined in the sequence in which they are delivered in the file.

<!ELEMENT VariablePrimaryKey (Name, Description?, (Numeric | AlphaNumeric, MaxLength ? | Date), Map*)>
Defines a key field (=key) in a VariableLength type file.

Used in container
VariableLength

The fields (VariableColumn, VariablePrimaryKey) must be defined in the sequence in which they are delivered in the file.

Concerning links also see example on page 39.

<!ELEMENT DataSupplier (Name, Location, Comment)>
Contains details concerning the data supplier.

Used in container
DataSet supplier (company,…)

<!ELEMENT Media (Name, Command*, Table+, Command*)>

Defines the content of a data medium.

Used in container
DataSet data media transfer

<!ELEMENT Table (URL, Name?, Description?, Validity?, (ANSI | Macintosh | OEM | UTF16 | UTF7 | UTF8)?, (DecimalSymbol, DigitGroupingSymbol)?, SkipNumBytes?, Range?, (VariableLength | FixedLength)?)>

Defines a table.

Used in container
Media table

<!ELEMENT ForeignKey (Name+, References)>

Defines a column as a foreign key.

The column referenced by name must be defined as a FixedColumn or VariableColumn in the same table.

Used in container	Context-dependent description
FixedLength	Field in FixedLength file, referenced table
VariableLength	Field in FixedLength file, referenced table

Kommentar [FP1]: Hier stand im Deutschen tatsächlich „FixedLength-Datei“ - müsste aber doch wohl „VariableLength“ heißen?

In this regard also see example on page 39.

<!ELEMENT Map (Description?, From, To)>

Redefinitions at the data level can be carried out by means of the MAP element.

Example:

From (original data)	To (target data)
1	true
0	false
Hr	Herr (Mister)
Fr	Frau (Mrs/Miss)
01	Married
02	Single
03	Tax category 3

Used in container	Context-dependent description
FixedColumn	Field in FixedLength file
FixedPrimaryKey	Key in FixedLength file
VariableColumn	Field in VariableLength files
VariablePrimaryKey	Key in VariableLength files

Map may only be used for alpha-numerical fields.

You should avoid delivering coded errors as part of the data media transfer and instead convert all codes into textual descriptions. In the event that this conversion is not already carried out by the export programme, Map represents a possibility for converting codes using data description.

<!ELEMENT Validity (Range, Format?)>

Defines the validity period for the data.

Used in container	
Table	Validity period

Example 1 of an INDEX.XML

[Abb.]

[Abb.]

[Abb.]

[Abb.]

Example 2 regarding links

It may make sense in terms of the transfer of data media not to deliver movement data and master data in one broad table, but instead to save these into separate tables:

Example:

[Abb.]

The primary keys are described in the master data:

[Abb.]

The movement data describe the links to the master data via the specification of the foreign key relationship:

[Abb.]

Note: Although the description of a VariablePrimaryKey fully describes the column, the ForeignKey does not define a new column. The column used in the ForeignKey must already have been defined as a VariableColumn or FixedColumn in advance.

Example 3 of an INDEX.XML (compressed data)

Index.xml

[Abb.]

uncompress.bat

[Abb.]

%temp% is an operating system environment variable that represents the path to the temporary directory.

The file gdpdu.zip contains the compressed data media provision (including INDEX.XML and GDPDU-01-08-2002.DTD). After decompression the auditor therefore opens the new data provision in the directory %TEMP%\gdpdu.

Note: The <Version> specification does not relate to the version of the description standard but the version of the data media transfer.

Frequently asked questions

- Do we need to deliver data in XML format?
No! Only the description of the data (INDEX.XML) is in XML format. The reference data are in FixedLength or VariableLength format.

- Can your system handle data in DOS/Unix/Mac formats (line breaks)?
Yes, because the line breaks can be specified explicitly.

- How do negative figures need to be indicated (in front of/behind the figure, space character between preceding character and digit permissible)?
Negative figures must be preceded or followed by a minus sign without a space character (e.g. "-1782.90" or "1782,90-").

- When a transferred file is too large for a specific data medium, may it then be split across a number of data media?
This description standard does not provide for the splitting up of a file. However, in an instance like this, you may compress data or select a data medium with a higher capacity (e.g. DVD).

- Which XML parser do you use?
We use Xerces (http://xml.apache.org/xerces-c/index.html).

- Is it an error when the ForeignKey references a table that is not contained in the DataSet, or would the ForeignKey then be ignored?
The description of the links must match the delivered data. Otherwise, your data media transfer will not be consistent. The import process will indicate this inconsistency to the auditor.

- What happens when index.xml describes a table that is, however, subsequently not present?
The description of the files must match the delivered data. Otherwise, your data media transfer will not be consistent. The import process will indicate this inconsistency to the auditor.

- *Are commands compulsory? Do you offer environment variables for commands?*
Commands are not compulsory. When a .BAT or .CMD file is accessed, the conventional operating system environment variables are available. The import process itself does not set any environment variables.

- *How can it finally be checked whether the data of the data medium can be imported by your system?*
The safe method of testing is importing with IDEA for GDPdU (IDEA 2002 + SmartX). Audicon is planning an SDK (Software Development Kit), which amongst others will make available a verification option. A simple option for testing the XML file for its syntactic correctness is to open it with an XML-capable browser (e.g. Microsoft Internet Explorer 5.5 upwards).

- *Since I am not familiar with XML standards, I am not sure how to construct the files INDEX.XML and GDPDU.DTD concretely. I have sent you a VariableLength file. Would you be able to create the corresponding XML or DTD file for me?*
You should acquire some basic XML knowledge. You must (may!) not modify the DTD file. The description of the data (in other words INDEX.XML) should always be carried out from the point that has the data ownership, in other words the software manufacturer or the business subject to taxation. The GDPdU seminars by Audicon, which also address technical details (see www.GDPdU.com) serve as a source of information.

- *In some press releases, apart from VariableLength and FixedLength file formats, various others are mentioned. Which formats apply to data media transfers?*
The file formats referred to in some articles relate to the original import options of the software IDEA. The description standard is, however, product independent and in its current version only supports the two generally applicable formats of VariableLength and FixedLength.

- *Must GDPDU-01-08-2002.DTD also be burnt onto the CD ROM?*
Yes. A CD ROM serving data media transfer purposes must contain GDPDU-01-08-0220.DTD, INDEX.XML and all reference data (e.g. T1.CSV, T2.CSV, T3.CSV, T4.ASC).

- *In a VariableLength file, the field names are stored in the first dataset. How is it possible to peruse the first set?*
Use the range specification in the table element:

[Abb.]

- *By specifying a character set, e.g. UTF7 or Macintosh, is the specification of the line delimiter overridden automatically?*
No! Only by explicitly specifying the line delimiter (RecordDelimiter) is the standard value (CRLF) overridden.

Synopsis of description standard Version 1 and Version 1.1

Renaming is indicated in blue and new or deleted elements in red.

[Abb.]

[Abb.]

[Abb.]

[Abb.]

[Abb.]

[Abb.]