TRADEBYTE

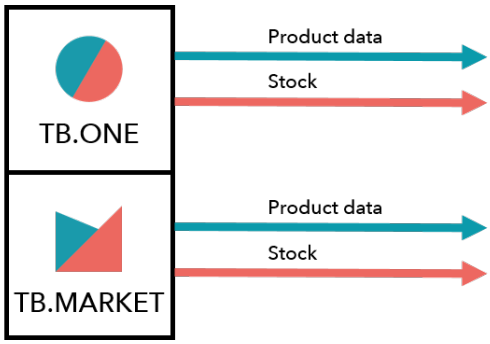
CHANNEL BASICS PART 2

PRODUCT DATA EXPORT VIA TB.CAT 1.3

TABLE OF CONTENTS

1. WHAT YOU CAN EXPECT IN THIS MANUAL

This document is intended for operators of marketplaces and platforms. It describes the product data export of TB.One or TB.Market to the channel.



THE DOCUMENTATION FOR THE CONNECTION OF CHANNELS INCLUDES 3 PARTS:

- **Part 1: Definitions and Process Overview**

This document defines the terminology used at Tradebyte and in TB.One.

- **Part 2: Product Data Exchange via TB.Cat**

Naming conventions and field descriptions for XML files in the Tradebyte standard "TB.Cat".

- **Part 3: Order Data Exchange via TB.Order**

Naming conventions for XML files in the Tradebyte standard "TB.Order", field descriptions of the TB.Order, and a guideline for message exchange (delivery, returns, cancellations).

2. DATA EXCHANGE WITH TB.ONE

The product data exchange between TB.One/TB.Market and the channel is done via XML in the Tradebyte standard TB.Cat. Via REST interface, you retrieve product data catalogs and inventories directly from a TB.Market or TB.One account. Generally, you will receive a part of the TB.CAT.xml as a result which you can continue processing.

Please agree upon the data exchange with your Tradebyte contact. They will also provide you with your login details.

Please note that extensions of the interface are reserved. We always try to provide the best scope. Therefore, enhancement of the interface may be necessary. Higher releases are generally compatible with former releases.

In general, time stamps follow the time stamp format "Unix timestamp", i.e. seconds since 01.01.1970 00:00 UTC.

The currently valid schema file can be found here:

```
https://api.core.tradebyte.com/schema/all_in_one/tb-cat_1_3_all_in_one.xsd
```

For a pure stock update, the TB.Stock scheme is available. The valid scheme file is:

```
https://api.core.tradebyte.com/schema/all_in_one/tb-stock_all_in_one.xsd
```

2.1 CALLING THE REST INTERFACE

You can receive product catalogs and inventory changes via the TB.One REST interface.

Basic structure of a REST-URL:

```
https://rest.core.tradebyte.com/ACCOUNT-NR/TARGET/?channel=CHANNEL_ID[&filter=criterion]
```

The call of the REST interface requires the authentication by user name and password as well as the release of this user for the requested channel in TB.One resp. TB.Market. To execute REST calls, create the API-user in the called system (TB.One/TB.Market) first. Also, assign the rights for requesting data from the appropriate channel.

REQUIRED FIELDS

TAB	DESCRIPTION
USR/PWD	User name and password for REST access
ACCOUNT NO	Merchant number
TARGET	Specifies the resource group which the call is for. Here: products or article portfolio (articles/stock).
CHANNEL_ID	The channel ID is required and will be provided by your Tradebyte contact. In the example calls the ID is referenced with CHID. If the API user is not assigned to this channel-ID, a call is not possible.

RETURN VALUES

Basically, a REST call returns a message via XML. You either get the requested result or a status message which is complemented by a status code in the HTTP-header to indicate the successful or unsuccessful processing.

In addition to the relevant content you have to consider the reported http-status-code (header):

- 200 (all codes beginning with 2xx): call/processing successful.
- 400 (all codes starting with 4xx): call not successful or request incorrect.
- 500 (all codes starting with 5xx): processing error.

2.2 REST CALLS

PRODUCT LISTS

Method	GET	Retrieves
Target	products	a complete product list (Full Load)
Example	.../products/?channel=CHID	
Response	<TBCATALOG channel="chxx" creation="1359543069"> <SUPPLIER> <NAME>Dokuaccount</NAME> </SUPPLIER> <CLASSIFICATION></CLASSIFICATION> <PRODUCTDATA type="full"> <PRODUCT></PRODUCT> <PRODUCT></PRODUCT> ... </PRODUCTDATA> </TBCATALOG>	

Method	GET	Retrieves
Target	products	a list of products
Parameter	delta	from a recent change in delta (delta load)
Example	.../products/?channel=CHID&delta=1359543069	
Response	<TBCATALOG channel="chxx" creation="1359543069"> <SUPPLIER></SUPPLIER> <PRODUCTDATA> <PRODUCT></PRODUCT> <PRODUCT></PRODUCT> ... </PRODUCTDATA> </TBCATALOG>	

INVENTORY LISTS

For channels with a main and several sub- channels, you have several channel-IDs available:

- If you use the ID of a sub-channel as CHID, you receive the stock of all articles activated for the specified sub-channel.
- Choose the ID of the main channel to get a stock list of activated articles for all sub-channels together.

Method	GET	Retrieves
Target	stock	an inventory list for the specified channel (Full Load)
Example	.../stock/?channel=CHID	
Response	<TBSTOCK changedate="1359705447"> <ARTICLE> <A_NR>4688-pi-42</A_NR> <A_STOCK>10</A_STOCK> </ARTICLE> ... </TBSTOCK>	

Method	GET	Retrieves
Ziel	stock	an inventory list for the specified channel
Parameter	delta	starting from a last change delta (delta load)
Example	.../stock/?channel=CHID&delta=1359705447	
Response	<TBSTOCK changedate="1359706621"> <ARTICLE> <A_NR>4689olxxl</A_NR> <A_STOCK>5</A_STOCK> </ARTICLE> </TBSTOCK>	

PRICE LISTS

Prices may not only be provided by means of a full product data export but you as a channel may retrieve changed price data also by just calling a price feed (full or delta).

Method	GET	Retrieves
Target	prices	an inventory list for the specified channel (Full Load)
Example	.../prices/?channel=CHID	
Response	<TBPRICES changedate="1359705447"> <ARTICLE> <A_NR>4688-pi-42</A_NR> <A_PRICEDATA> <A_PRICE channel="CHID" currency="EUR"> <A_VK>69.97</A_VK> <A_VK_OLD>89.95</A_VK_OLD> <A_UVP>99.95</A_UVP> <A_MWST>2</A_MWST> <A_EK>38.32</A_EK> </A_PRICE> </A_PRICEDATA> </ARTICLE> ... </TBPRICES>	

Method	GET	Retrieves
Target	prices	an inventory list for the specified channel
Parameter	delta	starting from a last change delta (delta load)
Example	.../stock/?channel=CHID&delta=1359705447	
Response		

```
<TBPRICES changedate="1359706621">
        <ARTICLE>
                <A_NR>4688-pi-42</A_NR>
                <A_PRICEDATA>
                        <A_PRICE channel="CHID" currency="EUR">
                                <A_VK>69.97</A_VK>
                                <A_VK_OLD>89.95</A_VK_OLD>
                                <A_UVP>99.95</A_UVP>
                                <A_MWST>2</A_MWST>
                                <A_EK>38.32</A_EK>
                        </A_PRICE>
                </A_PRICEDATA>
        </ARTICLE>
        ...
</TBPRICES>
```

3. CONTENT OF FULL AND DELTA LOADS

Full and delta loads are possible for both product and inventory data:

- Full loads contain the complete inventory of data that is valid at the time of the retrieval.
- Delta loads contain the changes made since the last call.

A **product data catalog** can contain all valid products, including all applicable articles ("full") or only modified products and articles ("delta"). A delta load provides the complete data set per product (including all possible variants), but only those articles being changed since the last time stamp. Not modified products are not included in a delta load and don't need to be processed further.

An **inventory data catalog** contains either the stock of all products valid at the time of retrieval and included in the last product catalog ("full") or only the inventory changes since the last polling time ("delta").

A full load of product data is allowed **once a day**. A full load of inventory data is permitted only **once an hour** in the standard.

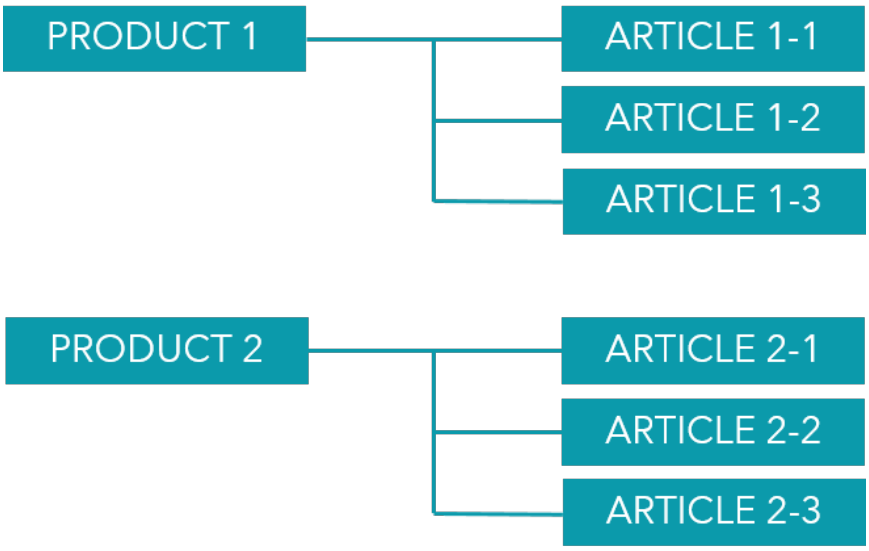
Please agree with your Tradebyte contact upon the time of full load.

4. DEPLOYMENT LOGIC FOR PRODUCT AND ARTICLE DATA

For both deployment types ("full" and "delta"), the provided TB.Cat always includes the fully available information for a product valid at the time of retrieval.

> In the subsequent processing, you are responsible for providing any information not contained or modified and adding new required information.

A product always includes all associated articles and all descriptive information. Here's a schematic view:



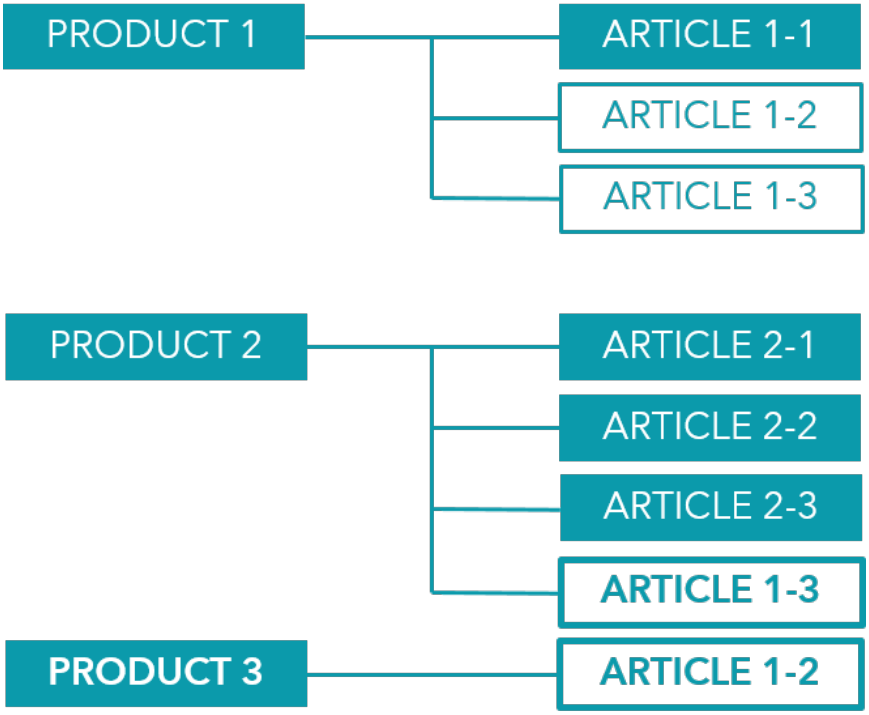
4.1 UNIQUE REFERENCE/IDENTIFICATION OF PRODUCTS AND ARTICLES

You can identify a product by the product ID (P_NR). This is the internal database ID assigned to this product during creation. The database ID is assigned once. In addition, you may optionally receive the product number of the data provider (P_NR_EXTERNAL). It may be modified by the data provider. From the combination of P_NR and P_NR_EXTERNAL, a change can be derived.

The same logic is used at article level. When creating an article, a unique article ID (A_ID) will be assigned. Additionally, you receive the article number (A_NR) of the data supplier which may theoretically be modified by the data provider.

4.2 MOVING EXISTING ARTICLES

An article can be assigned to only one product at a time. However, it is possible that the data provider shifts existing articles in either a different product or disperses the variants of a product in whole or in part to one or more new products.



4.3 PRODUCT DATA IN FULL LOAD

A product data catalog in "full load" contains all products and their associated articles valid at the time of retrieval. The data must be interpreted according to the "Deployment Logic for Product and Article Data[15]". At a call, you will receive a file with the following scheme:

```
<TBCATALOG channel="chxx" creation="1359554835">          <!-- Creation time stamp -->
    <PRODUCTDATA type="full">                             <!-- Call mode "Full-Load" -->
    <!-- delta request control: 0 -->
        <Product 1>
            <Article 1-1>
            <Article 1-2>
            <Article 1-3>
        <Product 2>
            <Article 2-1>
            <Article 2-2>
            <Article 2-3>
    </PRODUCTDATA>
</TBCATALOG>
```

The creation time (attribute "creation" at the TBCATALOG element) to use as timestamp for the filter attribute "delta" at the next request of a delta load.

Per default, a full load is only permitted once a day. Please agree with your Tradebyte contact upon the time of the full load.

4.4 PRODUCT DATA IN DELTA LOAD

If the product data catalog is retrieved as delta load, only those products are provided that have changed since the specified timestamp "delta". A delta load provides complete products including all currently assigned and valid articles and must be interpreted according to the "Deployment Logic for Product and Article Data [15]".

At a call, you will receive a file with the following scheme:

```
<TBCATALOG channel="chxx" creation="1359555698">          <!-- Creation timestamp -->
        <PRODUCTDATA type="delta">                        <!-- Call mode "Delta-Load" -->
        <!-- delta request control: 1359554835            <!-- requested timestamp  -->
                <Product 1>
                        <Article 1-1>
                        <Article 1-2>
                        <Article 1-3>
                        <Article 1-4>
        </PRODUCTDATA>
</TBCATALOG>
```

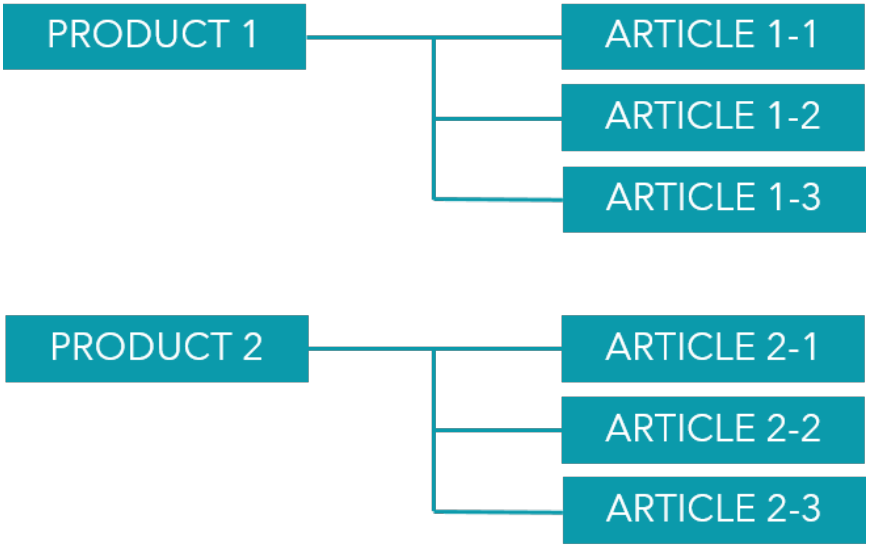
In this example, only the "Product 1" has been changed in the meantime.

Always transfer the creation timestamp of the previous TB.Cat product data retrieval at the next "delta" call. In this case, it is irrelevant whether the last product data retrieval was a full or a delta load.

4.5 EXAMPLE CONSTELLATION FOR PRODUCT DATA IN FULL/DELTA LOAD

The starting point is the example from the chapter "Deployment Logic for Product and Article Data [15]".

Please note: Grey products/articles are activated. White products/articled are disabled.

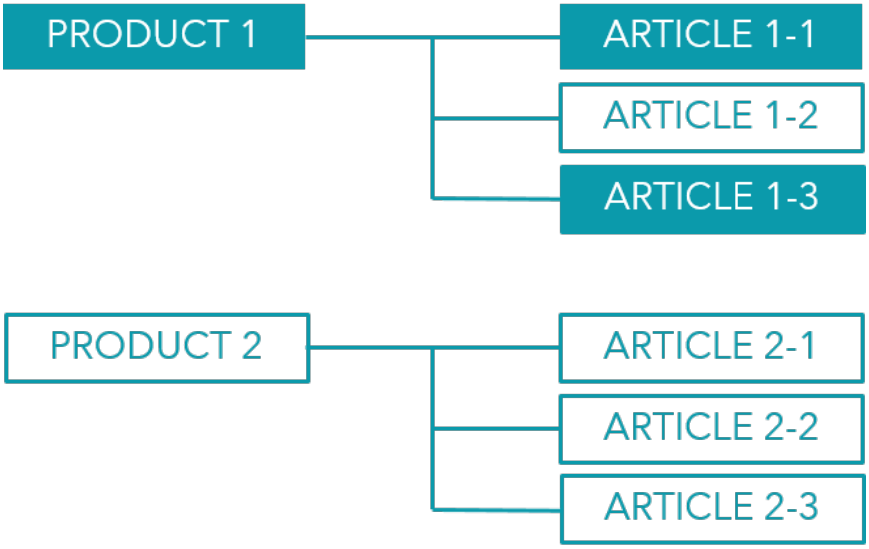


In the corresponding full load, you receive (schematic representation):

```
<TBCATALOG channel="chxx" creation="1359554835">          <!-- Creation timestamp  -->
        <PRODUCTDATA type="full">                         <!-- Call mode "Full-Load" -->
        <!-- delta request control: 0 -->
                <Product 1>
                        <Article 1-1>
                        <Article 1-2>
                        <Article 1-3>
                <Product 2>
                        <Article 2-1>
                        <Article 2-2>
                        <Article 2-3>
        </PRODUCTDATA>
</TBCATALOG>
```

PRODUCT DATA CATALOG WITH CHANGE OF PRODUCT DATA

Before the delivery of the next catalog, we deactivated article 2 in product 1 and the complete product 2.



Depending on whether the next product data retrieval is in full or delta load, you will get different results.

OPTION 1: DELTA LOAD

The subsequently provided delta load will contain (schematic representation):

```
<TBCATALOG channel="chxx" creation="1359555698">
        <PRODUCTDATA type="delta">
        <!-- delta request control: 1359554835 -->
                <Product 1>
                        <Article 1-1>
                        <Article 1-3>
                <Product 2 mode="delete">
        </PRODUCTDATA>
</TBCATALOG>
```

Please note that the articles 1-2 are implicitly disabled. Only at completely disabled products, the deletion is communicated explicitly in the feed.

The node for the deleted product only contains the product number to identify the product:

```
<PRODUCT mode="delete">
        <P_NR>2712</P_NR>
</PRODUCT>
```

OPTION 2: FULL LOAD

The provided full load shows the change implicitly and then contains (schematic representation):

```
<TBCATALOG channel="chxx" creation="1359555698">
    <PRODUCTDATA type="full">
    <!-- delta request control: 0 -->
        <Product 1>
            <Article 1-1>
            <Article 1-3>
    </PRODUCTDATA>
</TBCATALOG>
```

No longer provided products/articles of a full load must be disabled/deleted. They are not listed explicitly!

PRODUCT DATA CATALOG WITHOUT CHANGE OF PRODUCT DATA

If there are no changes until the next polling time, the requested delta load provides an empty catalog:

```
<TBCATALOG channel="chxx" creation="1359556548">
    <PRODUCTDATA type="delta">
    </PRODUCTDATA>
</TBCATALOG>
```

A full load would again contain all products and articles valid at the time of retrieval and would accordingly be identical in content to the catalog in the section "Option 2: Full Load 20".

5. DEPLOYMENT LOGIC OF INVENTORY INFORMATION

The inventory information is always exported on the basis of the articles sent to you, i.e. you only get inventory information when an article has already been exported with the product data and was interpreted as valid in the last export. Also, you can retrieve stock data as full or delta load.

5.1 STOCK DATA IN FULL LOAD

The retrieval of stock data in "full load" contains all articles valid at the time of retrieval based upon the currently valid product data provision. At a call, you will receive a file according to the following scheme:

```
<?xml version="1.0" encoding="UTF-8"?>
<TBSTOCK changedate="1378374296">                    <!-- creation timestamp -->
    <ARTICLE>
        <A_NR>2700000022XL</A_NR>
        <A_STOCK>19</A_STOCK>
    </ARTICLE>
    <ARTICLE>
        <A_NR>2700000022L</A_NR>
        <A_STOCK>47</A_STOCK>
    </ARTICLE>
</TBSTOCK>
```

The creation time of the last inventory data retrieval is to be used as value for "delta" when retrieving a delta load. An exception is when product data is retrieved in "full" mode. Please note the chapter "Processing Sequence of Full/Delta Loads 21".

A full load of inventory data is by default permitted only once an hour.

Please agree with your Tradebyte contact upon the time of full load.

5.2 STOCK DATA IN DELTA LOAD

If the stock data is retrieved via delta load, only article stock is included, changed since the specified timestamp "delta".

At a call, you will receive a file according to the following scheme:

```
<?xml version="1.0" encoding="UTF-8"?>
<TBSTOCK changedate="1378374296">
        <ARTICLE>
                <A_NR>2700000022XL</A_NR>
                <A_STOCK>0</A_STOCK>                    <!-- Article sold out -->
        </ARTICLE>
        <ARTICLE>
                <A_NR>2700000022L</A_NR>
                <A_STOCK>47</A_STOCK>
        </ARTICLE>
</TBSTOCK>
```

Always transfer the creation timestamp of the previous stock retrieval at the next "delta" call, irrespective if it was a full or a delta load. An exception is the retrieval of product data in "full" mode in the meantime. For details, please refer to the chapter "Processing Sequence of Full/Delta Loads 23".

The retrieval of new stock data (delta load) is allowed up to a maximum frequency of 15 minutes (in the standard) .

6. PROCESSING SEQUENCE OF FULL/DELTA LOADS

In general, the retrieved product and stock data is complete and must be processed in the order in which they were created and passed on. This can be fixed by means of the given timestamp.

```
<?xml version="1.0" encoding="UTF-8"?>
```

or

```
<TBCATALOG channel="chxx" creation="1359555698">
```

As stock information will be explicitly included in the product data, the following principles apply to the sequence of processing:

- Delta loads must always be fully processed in the creation chronology.
- A later retrieved full load supersedes all previously retrieved delta exports of the same type of poll, i.e. a full load of stock information supersedes all previously retrieved delta stock loads.
- A full load of the complete product data always replaces all previously retrieved delta loads regardless of whether product or stock information was retrieved in the delta loads.

7. REFERENCING OF DATA

The transmitted data can be passed in TB.Cat on different types of referencing. The main distinction is the way in which this information can then be re-interpreted or are to be interpreted.

Currently, we distinguish three types of referencing:

7.1 KEY

The exchange of the transferred information is based on a defined key. It is indicated by the predetermined attribute "identifier=key". The key to interpret is then defined as an attribute in the form "key=defined key". The actual content is transferred as an XML value if it is relevant to the given information (e.g. for components).

This type of referencing is always used when the receiver of the data will assign and interpret the information accurately.

```
<P_BRAND identifier="key" key="supjam" name="Superjackenmeister" />
<P_COMPONENTDATA>
	<P_COMPONENT identifier="key" key="width" name="Breite">
		<VALUE xml:lang="de-DE">30cm</VALUE>
	</P_COMPONENT>
	<P_COMPONENT identifier="key" key="color" name="Farbe">
		<VALUE xml:lang="de-DE">Grün</VALUE>
	</P_COMPONENT>
</P_COMPONENTDATA>
<P_TAGS>
	<P_TAG identifier="key" key="gender" name="Zielgruppe">
		<VALUES>
			<VALUE identifier="key" key="male" name="Herren" />
		</VALUES>
	</P_TAG>
</P_TAGS>
```

7.2 NAME

The transferred information is provided based on the name (designation) defined by the sender. This is indicated by the predetermined attribute "identifier=name". The identifier/name will be indicated with the attribute "key=<Name/Designator>". The actual content is transferred as an XML value if it is relevant to the given information.

This type of referencing is used when received data can be assigned via the transferred area (component, attribute, brand) but any information can/should still not be interpreted accurately.

```
<P_BRAND identifier="name" key="Superjackenmeister" />
<P_COMPONENTDATA>
	<P_COMPONENT identifier="name" key="Breite">
		<VALUE xml:lang="de-DE">30cm</VALUE>
	</P_COMPONENT>
	<P_COMPONENT identifier="name" key="Farbe">
		<VALUE xml:lang="de-DE">Grün</VALUE>
	</P_COMPONENT>
</P_COMPONENTDATA>
<P_TAGS>
	<P_TAG identifier="name" key="Geschlecht">
		<VALUES>
			<VALUE identifier="name" key="Herren" />
		</VALUES>
	</P_TAG>
</P_TAGS>
```

7.3 ID

The provided information is transmitted based on the database ID of the sender. This is indicated by the predetermined attribute "identifier=id". The database ID is then indicated with the attribute "key=<DB_ID>". The actual content is transferred as an XML value if it is relevant to the given information.

In addition, you get an appropriate CLASSIFICATION part for all the nodes that are indicated as ID-based. This is included as a table of contents at the beginning of the TB.Cat.xml. This CLASSIFICATION lists the used database IDs together with the name defined by the sender for each node (components, attributes, etc.).

This type of referencing is used when the transmitter is leading the information transfer. Usually, this can only be used with one data provider since the same information is communicated with different IDs for several senders.

```
<PRODUCTDATA type="full">
	<PRODUCT>
	...
		<P_BRAND identifier="id" key="577" />
		<P_COMPONENTDATA>
			<P_COMPONENT identifier="id" key="4">
				<VALUE xml:lang="de-DE">100% Jackenmaterial</VALUE>
			</P_COMPONENT>
		</P_COMPONENTDATA>
		<P_TAGS>
			<P_TAG identifier="id" key="55">
				<VALUES>
					<VALUE identifier="id" key="1953" />
				</VALUES>
		</P_TAGS>
	...
	</PRODUCT>
</PRODUCTDATA>
```

8. MULTILINGUALISM

The TB.Cat is able to provide multiple language information (translations, etc.) for a product. For each multilingual and maintainable information (component, product title, etc.), you receive a language tag in the attribute "xml:lang", from which results the respective language. At least one language is always transferred.

```
<P_COMPONENT identifier="key" key="wid" name="Länge">
        <VALUE xml:lang="de-DE">110 cm</VALUE>
        <VALUE xml:lang="en-US">43.31 in</VALUE>
</P_COMPONENT>
```

This structure is even valid if the data is delivered in one language only:

```
<P_COMPONENT identifier="key" key="wid" name="Länge">
        <VALUE xml:lang="de-DE">110 cm</VALUE>
</P_COMPONENT>
```

9. STRUCTURE OF PRODUCT DATA AT THE TB.CAT EXPORT

The following text describes the nodes for the transfer within an exported product data catalog. The examples are related to monolingual and multilingual export. The transfer is not mixed! If the export contains several languages, the translation of all translatable elements is provided.

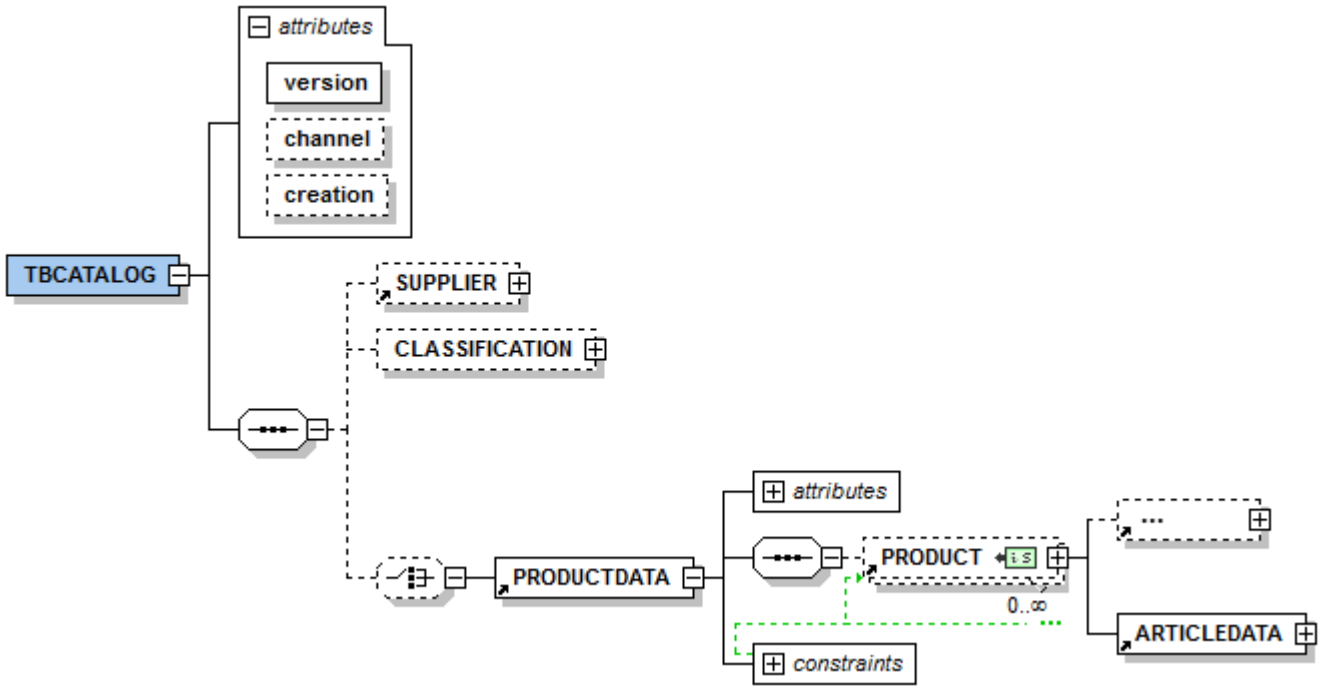
Please note the charts for the fields regarding mandatory and optional nodes. In the screenshots, solid lines indicate mandatory information, and dotted lines are for optional information.

The descriptions of mandatory and optional fields in the following chapters refer to the respective node and not the overall XSD.

Example:

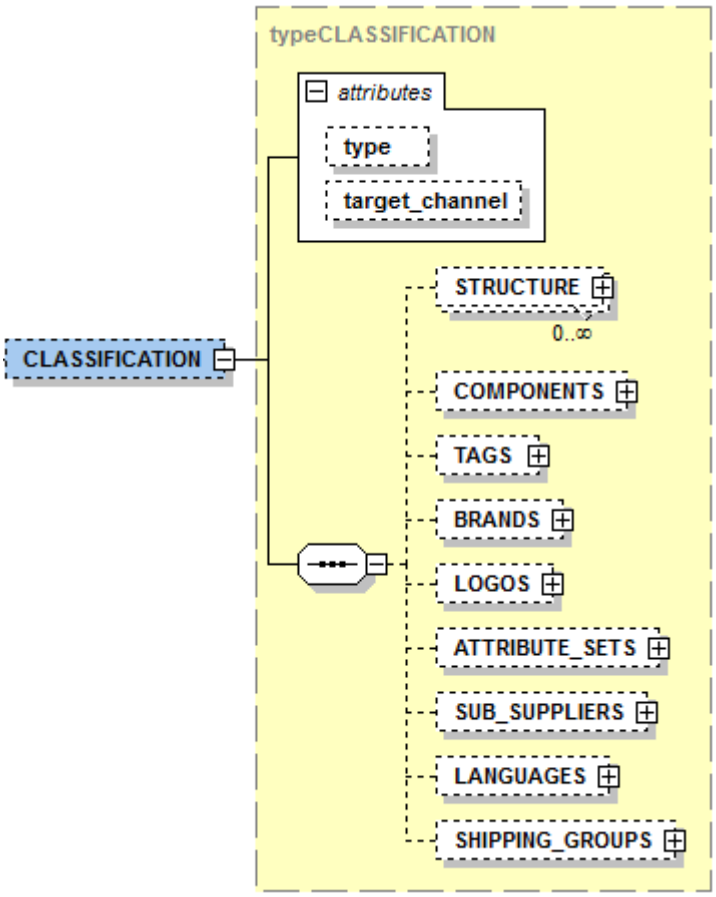
At "TBCATALOG," the "PRODUCTDATA" is framed by a solid line to show that this is mandatory. But one level above, the whole node, which includes the "PRODUCTDATA", is indicated as optional by a dotted line. This means: If the optional node is provided, the included "PRODUCTDATA" is mandatory. Another level above, the whole node, which lists <SUPPLIER>, <CLASSIFICATION>, and <PRODUCTDATA>, is also mandatory. For all sub-nodes are optional, this means: <TBCATALOG> needs at least one of the nodes <SUPPLIER>, <CLASSIFICATION>, or <PRODUCTDATA>.

9.1 TBCATALOG



NODE	MUST	TYPE	DESCRIPTION
<?xml version="1.0" encoding="UTF-8"?>	Y	XML	Coding of the export file, currently always UTF-8
TBCATALOG	Y	XML	Root node
version	Y	XML	TB.Cat version, here 1.3
SUPPLIER	N	XML	Information to the data supplier, e.g. TB.Market name
CLASSIFICATION 30	N	XML	Classification part for the ID definition of the category tree, components, attributes, brands, and logos, if it is referenced by IDs (see chap.7 24).
PRODUCTDATA 35	N	XML	Product data
ARTICLEDATA 61	N	XML	Article data: At this export type, articles will always be associated with product data.

9.2 TBCATALOG / CLASSIFICATION



NODE	MUST	TYPE	DESCRIPTION
attributes	*N*	*XML*	*Only relevant for the import*
STRUCTURE	N	XML	Category tree; In a tree structure of categories the PARENT_ID identifies the structure, i.e. the next higher-level category. In a flat category structure, the PARENT_ID is always 0.
COMPONENTS	N	XML	Properties with user-defined values
TAGS	N	XML	Properties with values from a value list
BRANDS	N	XML	Brand list
LOGOS	N	XML	Logo/Icon list
ATTRIBUTE_SETS	N	XML	Mandatory and optional values per category

NODE	MUST	TYPE	DESCRIPTION
SUB_SUPPLIERS	N	XML	Previous suppliers (referring to product data)
LANGUAGES	N	XML	Definition of export keys for languages
SHIPPING_GROUPS	N	XML	Definition of shipping types

Example-XML

```
<CLASSIFICATION>
	<STRUCTURE id="1">
		<STRUCTURE_ITEM>
			<KEY>11</KEY>
			<NAME>Accessoires</NAME>
			<PARENT_ID>0</PARENT_ID>
			<SORT>0</SORT>
		</STRUCTURE_ITEM>
		<STRUCTURE_ITEM>
			<KEY>12</KEY>
			<NAME>Badmöbel</NAME>
			<PARENT_ID>0</PARENT_ID>
			<SORT>0</SORT>
		</STRUCTURE_ITEM>
	</STRUCTURE>
	<COMPONENTS>
		<COMPONENT>
			<KEY>color</KEY>
			<NAME>Farbe</NAME>
		</COMPONENT>
		<COMPONENT>
			<KEY>material</KEY>
			<NAME>Material</NAME>
		</COMPONENT>
```

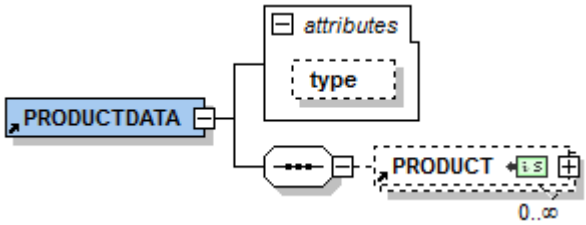
```
</COMPONENTS>
<TAGS>
        <TAG>
                <KEY>design</KEY>
                <NAME>Design</NAME>
                <VALUES>
                        <VALUE>
                                <ID>1</ID>
                                <KEY>1</KEY>
                                <NAME>retro</NAME>
                        </VALUE>
                        <VALUE>
                                <ID>2</ID>
                                <KEY>451</KEY>
                                <NAME>klassisch</NAME>
                        </VALUE>
                </VALUES>
        </TAG>
</TAGS>
<BRANDS>
        <BRAND>
                <KEY>hofre</KEY>
                <NAME>Hosenfreund</NAME>
        </BRAND>
        <BRAND>
                <KEY>supjam</KEY>
                <NAME>Superjackenmeister</NAME>
        </BRAND>
</BRANDS>
<LOGOS>
        <LOGO>
                <ID>1</ID>
```

```
                <KEY>60wasch</KEY>
                <NAME>60° Wäsche</NAME>
                <MEDIA type="logo" sort="0">http://www.imageserver.tld/123.jpg</MEDIA>
            </LOGO>
        </LOGOS>
        <ATTRIBUTE_SETS>
            <ATTRIBUTE_SET global="true">
                <NAME>Wertemapping</NAME>
                <ALLOW_OTHER_VARIANTS>true</ALLOW_OTHER_VARIANTS>
                <ALLOW_OTHER_COMPONENTS>true</ALLOW_OTHER_COMPONENTS>
                <ALLOW_OTHER_TAGS>true</ALLOW_OTHER_TAGS>
                <ATTRIBUTE_SET_ITEM minOccurs="0" maxOccurs="1">
                    <TAG identifier="key" key="color"/>
                </ATTRIBUTE_SET_ITEM>
                <ATTRIBUTE_SET_ITEM minOccurs="0" maxOccurs="1">
                    <TAG identifier="key" key="careinstructions"/>
                </ATTRIBUTE_SET_ITEM>
                <ATTRIBUTE_SET_ITEM minOccurs="0" maxOccurs="1">
                    <TAG identifier="key" key="Energieeffizienzklasse"/>
                </ATTRIBUTE_SET_ITEM>
            </ATTRIBUTE_SET>
            <ATTRIBUTE_SET>
            <ATTRIBUTE_SET>
                <NAME>Multimediaartikel mit Speichermöglichkeiten</NAME>
                <ATTRIBUTE_SET_ITEM minOccurs="0" maxOccurs="1">
                    <TAG identifier="key" key="WritableMediaClassification"/>
                </ATTRIBUTE_SET_ITEM>
            </ATTRIBUTE_SET>
        </ATTRIBUTE_SETS>
        <SUB_SUPPLIERS>
            <SUB_SUPPLIER>
                <ID>3</ID>
```

```
            <KEY>4711</KEY>
            <NAME>Hosenfreund GmbH</NAME>
            <PREFIX>HOF-</PREFIX>
        </SUB_SUPPLIER>
        <SUB_SUPPLIER>
            <ID>5</ID>
            <KEY>4712</KEY>
            <NAME>Superjackenmeister AG</NAME>
            <PREFIX>SUJ-</PREFIX>
        </SUB_SUPPLIER>
    </SUB_SUPPLIERS>
    <LANGUAGES>
        <LANGUAGE>
            <ID>58</ID>
            <KEY>de_DE</KEY>
            <NAME>Deutsch</NAME>
            <MANDATORY>true</MANDATORY>
        </LANGUAGE>
        <LANGUAGE>
            <ID>59</ID>
            <KEY>en_US</KEY>
            <NAME>Englisch</NAME>
        </LANGUAGE>
    </LANGUAGES>
    <SHIPPING_GROUPS>
        <SHIPPING_GROUP>
            <ID>58</ID>
            <NAME>paketversandfähig</NAME>
        </SHIPPING_GROUP>
        <SHIPPING_GROUP>
            <ID>59</ID>
            <NAME>Spedition 1-Mann-Handling</NAME>
```

```
            </SHIPPING_GROUP>
        </SHIPPING_GROUPS>
</CLASSIFICATION>
```

9.3 TBCATALOG / PRODUCTDATA

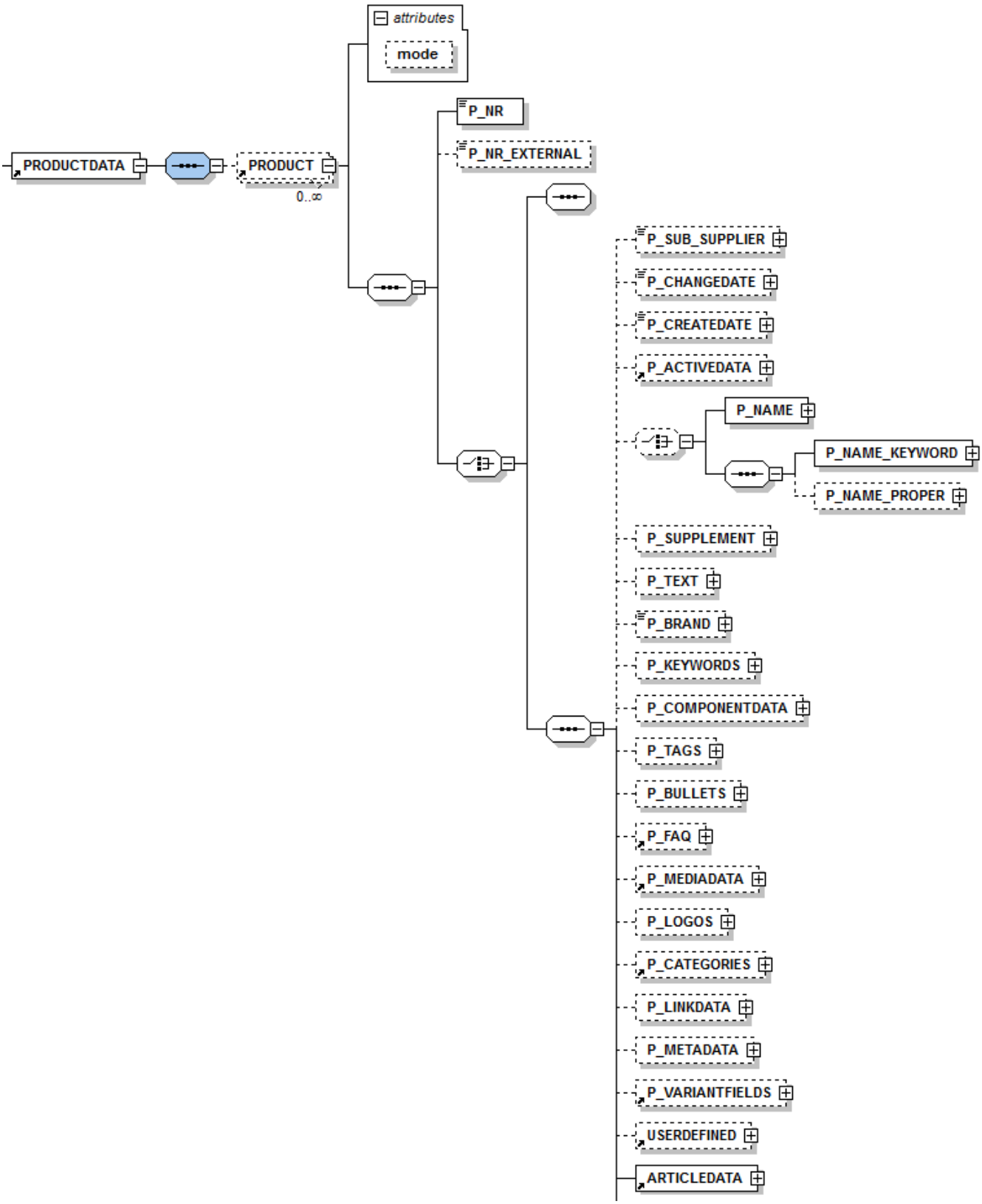


NODE	MUST	TYPE	DESCRIPTION
PRODUCTDATA	Y		Product data
PRODUCTDATA[type]	N	ENUM	Type of transmitted product data Possible values are: full = full product data catalog delta = delta load with changes
PRODUCT [36]	Y	XML	Data on a product. The node may be repeated as desired.

Example-XML

```
<PRODUCTDATA type="full">
        <PRODUCT>...</PRODUCT>
        <PRODUCT>...</PRODUCT>
</PRODUCTDATA>
```

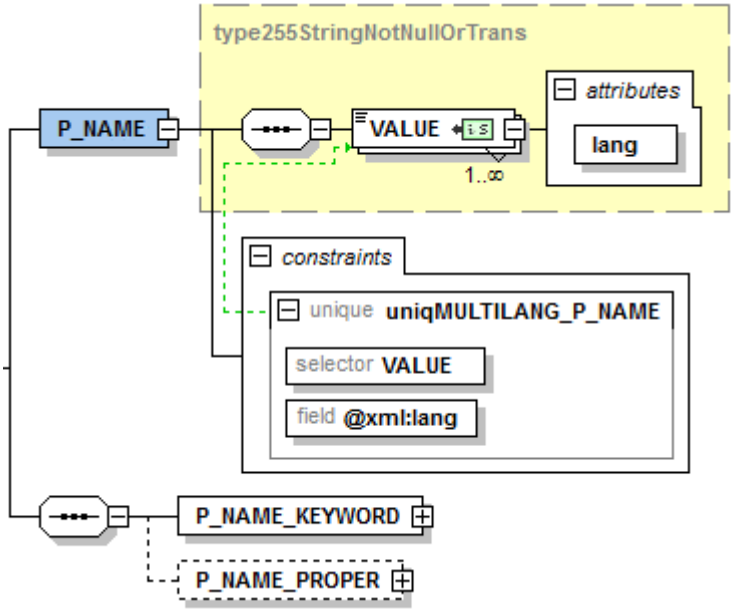
TBCATALOG / PRODUCTDATA / PRODUCT



NODE	MUST	TYPE	DESCRIPTION
mode	N	string	If a product is to be deleted you'll receive here <mode="delete"> (only delta load).
P_NR	Y	CHAR(50)	Unique product ID assigned by TB.One
P_NR_EXTERNAL	N	CHAR(50)	Product number of the supplier
P_SUB_SUPPLIER	N	XML	Information on suppliers (if multiple suppliers)
P_CHANGEDATE	N	Unix	Timestamp of the last product change
P_CREATEDATE	N	Unix	Timestamp of the product creation
P_ACTIVEDATA	N	XML	Activation flag for the product. You'll receive only active products, therefore, this is not necessarily to process.
P_NAME 39	Y	CHAR(255)	Product designation
P_NAME_KEYWORD 39	Y	CHAR(255)	Instead of the product designation P_NAME, the name may also be transferred split into common noun and proper name (e.g. "Jacket Reinhold" vs. "Jacket"/"Reinhold"). This must be arranged separately. Per default, P_NAME is provided here.
P_NAME_PROPER 39	N	CHAR(255)	
P_SUPPLEMENT 41	N	CHAR(255)	Additional promotional headline for the product
P_TEXT 42	N	TEXT	Product description (body text)
P_BRAND 43	N	XML	Brand
P_KEYWORDS 44	N	XML	Keywords/search terms for this product
P_COMPONENTDATA 45	N	XML	Product components
P_TAGS 47	N	XML	Product attributes

NODE	MUST	TYPE	DESCRIPTION
P_BULLETS 49	N	XML	Bullet points
P_FAQ 50	N	XML	Not used by default
P_MEDIADATA 50	N	XML	Product media (images, documents,...)
P_LOGOS 52	N	XML	Product logos (care instructions,...)
P_CATEGORIES 54	N	XML	Category assignments
P_LINKDATA 55	N	XML	Links (cross-selling, accessories,...)
P_METADATA	-	XML	Not used by default
P_VARIANTFIELDS 56	N	XML	For products with variants: definition of the variant forming properties
USERDEFINED 57	N	XML	Custom fields
ARTICLEDATA 58	Y	XML	Article Data

TBCATALOG / PRODUCTDATA / PRODUCT / P_NAME



NODE	MUST	TYPE	DESCRIPTION
P_NAME oder P_NAME_KEYWORD P_NAME_PROPER	Y Y N	TEXT	The product designation can be provided in one field (P_NAME) or in the two fields common noun and proper name (P_NAME_KEYWORD and P_NAME_PROPER).
VALUE xml:lang="lang"	Y	TEXT	Definition of the language with the corresponding translation

Example-XML

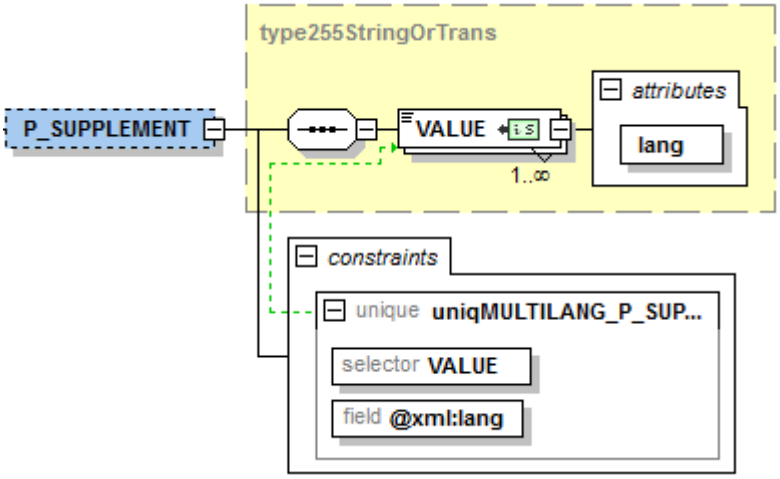
```
<P_NAME>
        <VALUE xml:lang="de-DE">Outdoor-Jacke Reinhold</VALUE>
        <VALUE xml:lang="en-US">Outdoor-Jacket Reinhold</VALUE>
</P_NAME>
```

or

```
<P_NAME_KEYWORD>
        <VALUE xml:lang="de-DE">Outdoor-Jacke</VALUE>
</P_NAME_KEYWORD>
<P_NAME_PROPER>
```

```
        <VALUE xml:lang="de-DE">Reinhold</VALUE>
</P_NAME_PROPER>
```

TBCATALOG / PRODUCTDATA / PRODUCT / P_SUPPLEMENT

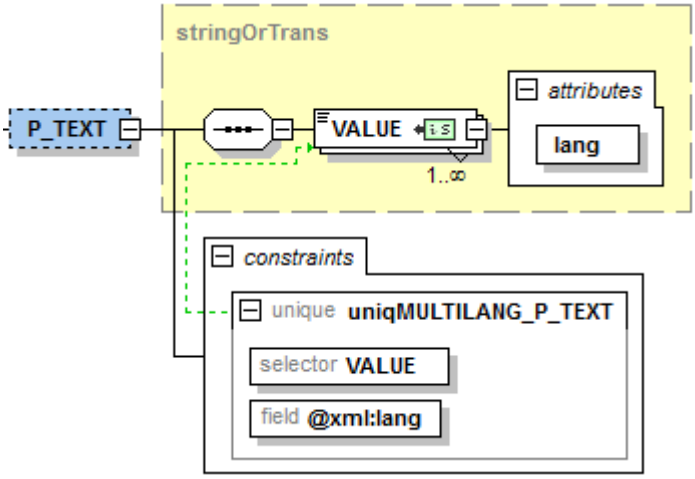


NODE	MUST	TYPE	DESCRIPTION
P_SUPPLEMENT	N	TEXT	The special text is e.g. a second advertising headline.
VALUE xml:lang="lang"	Y	TEXT	Definition of the language with appropriate text

Example-XML

```
<P_SUPPLEMENT>
    <VALUE xml:lang="de-DE">Außergewöhnlich robust und funktional!</VALUE>
</P_SUPPLEMENT>
```

TBCATALOG / PRODUCTDATA / PRODUCT / P_TEXT

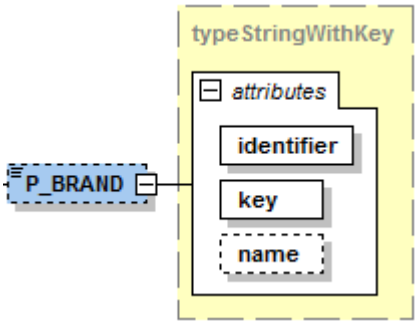


NODE	MUST	TYPE	DESCRIPTION
P_TEXT	N	TEXT	Product description as a continuous text without self-promotion
VALUE xml:lang="lang"	N	TEXT	Definition of the language with value

Example-XML

```
<P_TEXT>
    <VALUE xml:lang="de-DE">
        Die Outdoorjacke "Reinhold" ist eine außergewöhnlich robuste und funktionale
        Jacke! Durch das innovative G-1000 Eco und G-1000 Heavy Duty-Material bietet sie
        eine enorme Widerstandsfähigkeit und einen effektiven Schutz vor Wind und Witte-
        rung, Kälte, Nässe und Sturm. Die Outdoorjacke "Reinhold" ist ein sehr anpas-
        sungsfähiger Allrounder für das ganze Jahr, die Sie begeistern wird.
    </VALUE>
</P_TEXT>
```

TBCATALOG / PRODUCTDATA / PRODUCT / P_BRAND



NODE	MUST	TYPE	DESCRIPTION
P_BRAND	N	- empty -	Brand of the product
P_BRAND[identifier]	Y	ENUM	Identifier for referencing [24]
P_BRAND[key]	Y	CHAR(50)	Value stored in the identifier assignment statement
P_BRAND[name]	N	CHAR(100)	Name stored in the identifier assignment statement

Example-XML (key-based)

```
<P_BRAND identifier="key" key="supjam" name="Superjackenmeister" />
```

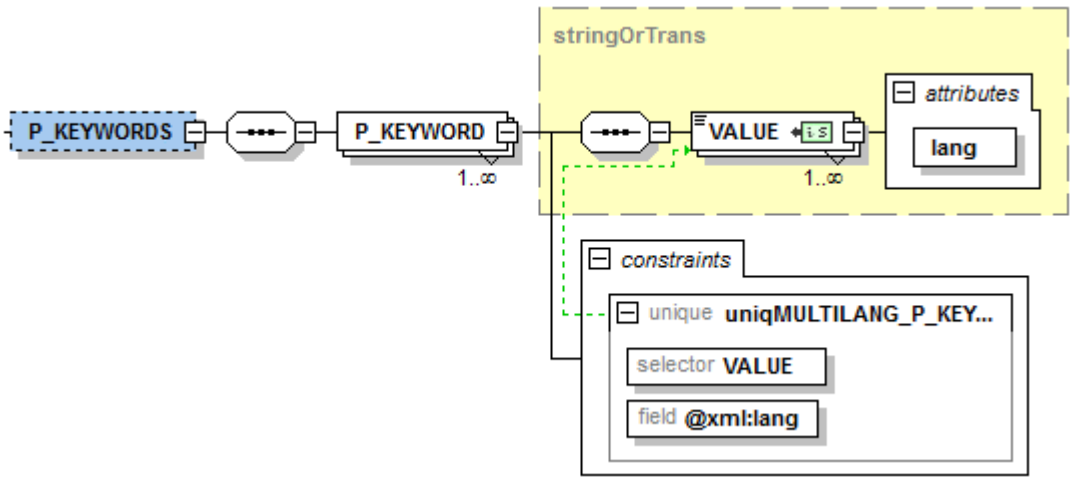
Example-XML (name-based)

```
<P_BRAND identifier="name" key="Superjackenmeister" />
```

Example-XML (ID-based)

```
<P_BRAND identifier="ID" key="4712" />
```

TBCATALOG / PRODUCTDATA / PRODUCT / P_KEYWORDS

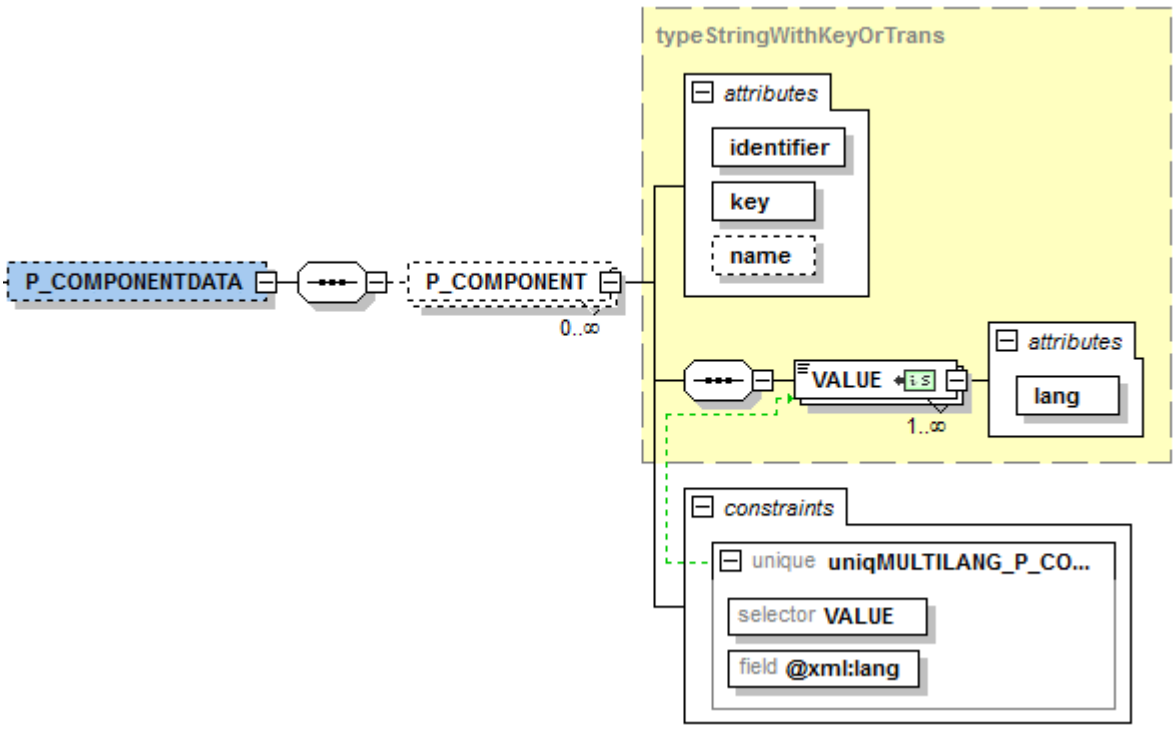


NODE	MUSTE	TYPE	DESCRIPTION
P_KEYWORDS	N	- empty -	Search terms for the product
P_KEYWORD	Y	CHAR(255)	Single search term
VALUE xml:lang="lang"	N	TEXT	Definition of the language with value

Example-XML

```
<P_KEYWORDS>
	<P_KEYWORD>
		<VALUE xml:lang="de-DE">Kapuzenjacke</VALUE>
	</P_KEYWORD>
	<P_KEYWORD>
		<VALUE xml:lang="de-DE">Anorak</VALUE>
		<VALUE xml:lang="en-US">Anorak</VALUE>
	</P_KEYWORD>
</P_KEYWORDS>
```

TBCATALOG / PRODUCTDATA / PRODUCT / P_COMPONENTDATA



NODE	MUST	TYPE	DESCRIPTION
P_COMPONENTDATA	N	XML	Product components
P_COMPONENT	Y	TEXT	Text of the product component
P_COMPONENT[identifier]	Y	ENUM	Identifier for referencing [24]
P_COMPONENT[key]	Y	CHAR(50)	Value stored in the *identifier* assignment statement
P_COMPONENT[name]	N	CHAR(100)	Name stored in the *identifier* assignment statement
VALUE xml:lang="lang"	N	TEXT	Definition of the language

Example-XML (key-based)

```
<P_COMPONENTDATA>
    <P_COMPONENT identifier="key" key="width" name="Breite">
        <VALUE xml:lang="de-DE">30cm</VALUE>
```

```
        </P_COMPONENT>
        <P_COMPONENT identifier="key" key="color" name="Farbe">
                <VALUE xml:lang="de-DE">Grün</VALUE>
                <VALUE xml:lang="en-US">Green</VALUE>
        </P_COMPONENT>
</P_COMPONENTDATA>
```

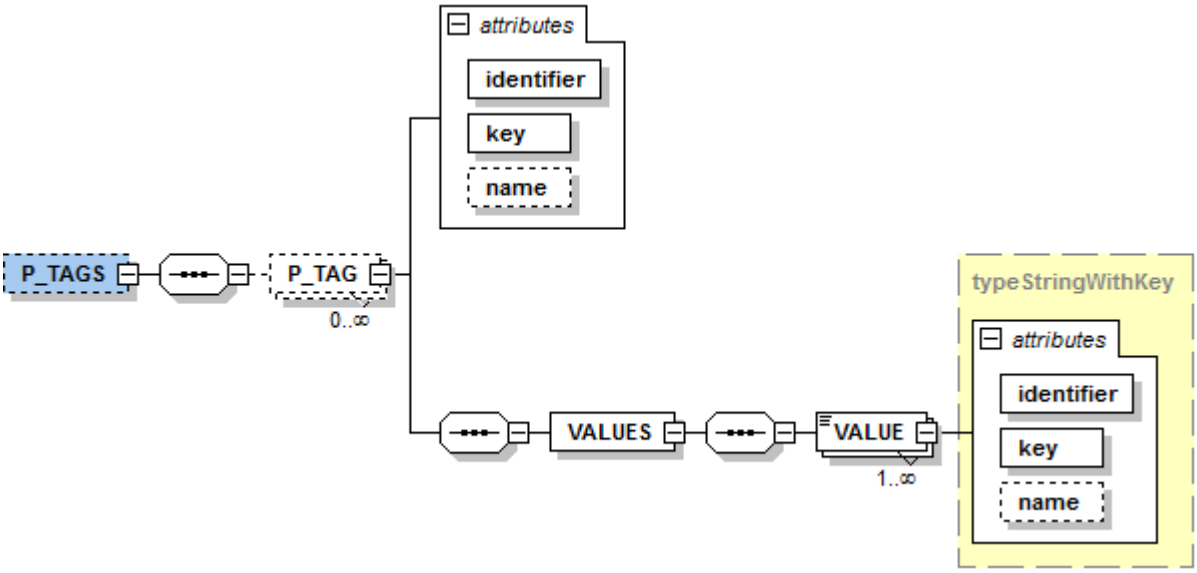
Example-XML (name-based)

```
<P_COMPONENTDATA>
        <P_COMPONENT identifier="name" key="Breite">
                <VALUE xml:lang="de-DE">30cm</VALUE>
        </P_COMPONENT>
        <P_COMPONENT identifier="name" key="Farbe">
                <VALUE xml:lang="de-DE">Grün</VALUE>
                <VALUE xml:lang="en-US">Green</VALUE>
        </P_COMPONENT>
</P_COMPONENTDATA>
```

Example-XML (ID-based)

```
<P_COMPONENTDATA>
        <P_COMPONENT identifier="ID" key="12">
                <VALUE xml:lang="de-DE">30cm</VALUE>
        </P_COMPONENT>
        <P_COMPONENT identifier="ID" key="13">
                <VALUE xml:lang="de-DE">Grün</VALUE>
                <VALUE xml:lang="en-US">Green</VALUE>
        </P_COMPONENT>
</P_COMPONENTDATA>
```

TBCATALOG / PRODUCTDATA / PRODUCT / P_TAGS



NODE	MUST	TYPE	DESCRIPTION
P_TAGS	N	XML	Product attributes
P_TAG	N	XML	Product attribute
P_TAG[identifier]	Y	ENUM	Identifier for referencing [24]
P_TAG[key]	Y	CHAR(50)	Value stored in the *identifier* assignment statement
P_TAG[name]	N	CHAR(100)	Name stored in the *identifier* assignment statement
VALUES	Y	XML	Attribute values
VALUE	Y	XML	Single attribute value
VALUE[identifier]	Y	ENUM	Identifier for referencing [24]
VALUE[key]	Y	CHAR(50)	Value stored in the *identifier* assignment statement

NODE	MUST	TYPE	DESCRIPTION
VALUE[name]	N	CHAR(100)	Name stored in the *identifier* assignment statement

Example-XML (key-based)

```
<P_TAGS>
	<P_TAG identifier="key" key="gender" name="Zielgruppe">
		<VALUES>
			<VALUE identifier="key" key="male" />
		</VALUES>
	</P_TAG>
</P_TAGS>
```

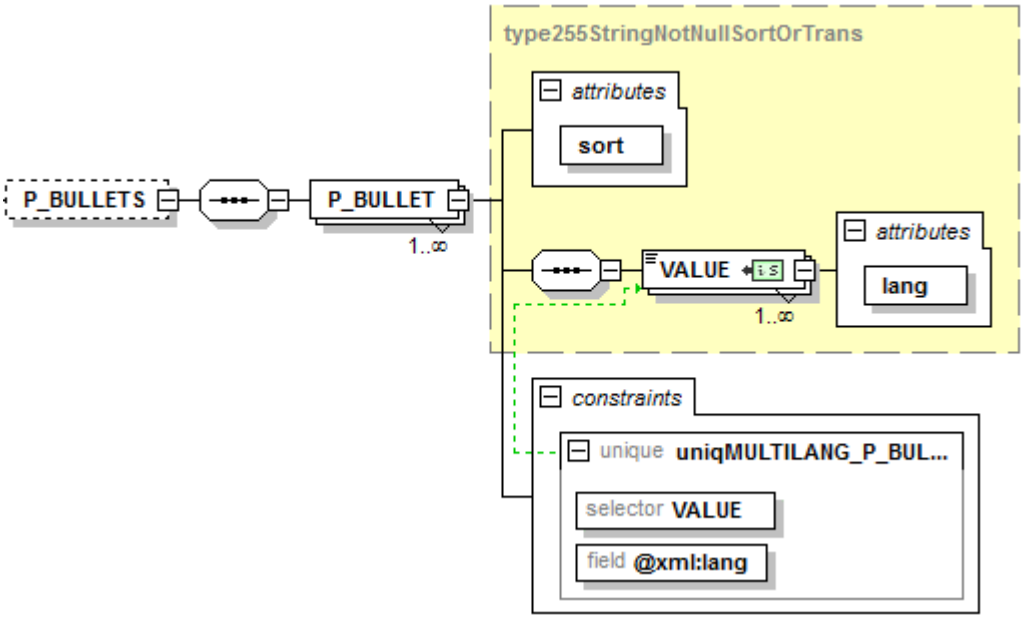
Example-XML (name-based)

```
<P_TAGS>
	<P_TAG identifier="name" key="Zielgruppe">
		<VALUES>
			<VALUE identifier="name" key="männlich" />
		</VALUES>
	</P_TAG>
</P_TAGS>
```

Example-XML (ID-based)

```
<P_TAGS>
	<P_TAG identifier="ID" key="15">
		<VALUES>
			<VALUE identifier="ID" key="17" />
		</VALUES>
	</P_TAG>
</P_TAGS>
```

TBCATALOG / PRODUCTDATA / PRODUCT / P_BULLETS



NODE	MUST	TYPE	DESCRIPTION
P_BULLETS	N	XML	Bullet points
P_BULLET	Y	CHAR(255)	Single bullet point
P_BULLET[sort]	Y	INT	Sorting within the bullet points
VALUE xml:lang="lang"	Y	TEXT	Definition of the language with value

Example-XML

```
<P_BULLETS>
	<P_BULLET sort="10">
		<VALUE xml:lang="de-DE">atmungsaktiv</VALUE>
	</P_BULLET>
	<P_BULLET sort="20">
		<VALUE xml:lang="de-DE">wasserdicht</VALUE>
	</P_BULLET>
	<P_BULLET sort="30">
		<VALUE xml:lang="de-DE">winddicht</VALUE>
		<VALUE xml:lang="en-US">wind proof</VALUE>
```

```
        </P_BULLET>
</P_BULLETS>
```

TBCATALOG / PRODUCTDATA / PRODUCT / P_FAQ

This node is not part of the TB.CAT standard features and is only available when using TB.Market.

Please reach out to your Tradebyte contact if required.

TBCATALOG / PRODUCTDATA / PRODUCT / P_MEDIADATA

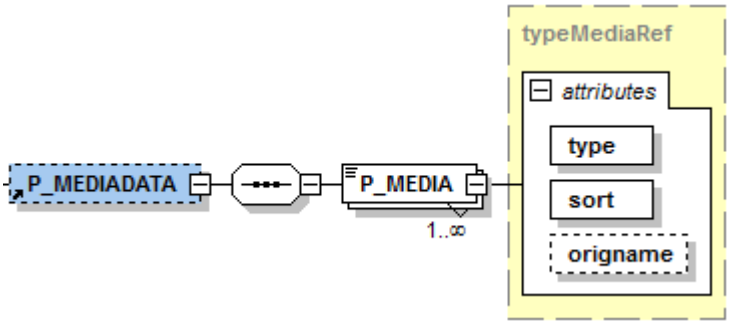
Since TB.One does not only handle images but also other file types, we refer to it as "media". Via the TB.Cat, media is transferred with a reference (HTTP) to the Tradebyte server. For each designated medium, a media type is communicated.

Generally, we differentiate between product and article media. Product media apply to the entire product, article media for the designated variant.

The Tradebyte server may only be used as a transfer option. The media must be retrieved once and then be saved for further use. The media URLs may therefore explicitly not be referenced directly in the shop.

The media made available are provided in the original size.

The media related to a media URL does not change. When new media is assigned to a product, a new media URL is communicated.



NODE	MUST	TYPE	DESCRIPTION
P_MEDIADATA	N	XML	Product media
P_MEDIA	Y	CHAR(255)	URL of product media
P_MEDIA[type]	Y	CHAR(50)	Media type
P_MEDIA[sort]	Y	INT	Sorting within the product media
P_MEDIA[origname]	N	CHAR(255)	Original name of the file

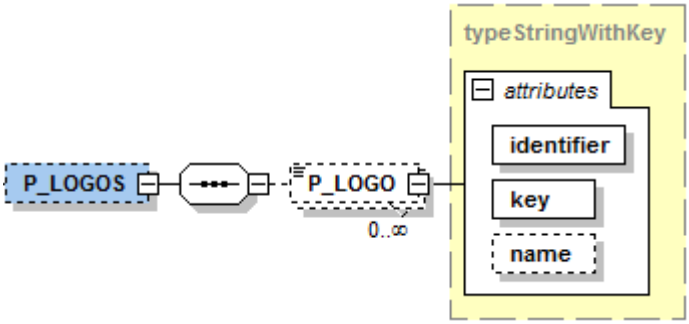
Example-XML

```
<P_MEDIADATA>
        <P_MEDIA type="image" sort="1">http://www.imageserver.tld/123.jpg</P_MEDIA>
        <P_MEDIA type="image" sort="2">http://www.imageserver.tld/124.jpg</P_MEDIA>
</P_MEDIADATA>
```

TBCATALOG / PRODUCTDATA / PRODUCT / P_LOGOS

Logos are usually used to represent standardized product features, e.g. care symbols (30° wash), certificates (FSC), or features (Bluetooth).

Please reach out to your Tradebyte contact if required.



NODE	MUST	TYPE	DESCRIPTION
P_LOGOS	N	XML	Product logos
P_LOGO	Y	- empty -	Product logo
P_LOGO[identifier]	Y	ENUM	Identifier for referencing [24]
P_LOGO[key]	Y	CHAR(50)	Value stored in the *identifier* assignment statement
P_LOGO[name]	N	CHAR(100)	Name stored in the *identifier* assignment statement

Example-XML (key-based)

```
<P_LOGOS>
    <P_LOGO identifier="key" key="machine_wash" />
    <P_LOGO identifier="key" key="iron_left" />
</P_LOGOS>
```

Example-XML (name-based)

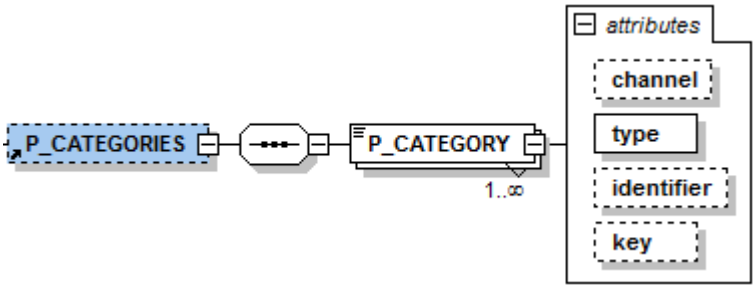
```
<P_LOGOS>
    <P_LOGO identifier="name" key="Maschinenwäsche" />
```

```
        <P_LOGO identifier="name" key="links bügeln" />
</P_LOGOS>
```

Example-XML (ID-based)

```
<P_LOGOS>
        <P_LOGO identifier="ID" key="18" />
        <P_LOGO identifier="ID" key="19" />
</P_LOGOS>
```

TBCATALOG / PRODUCTDATA / PRODUCT / P_CATEGORIES



NODE	MUST	TYPE	DESCRIPTION
P_CATEGORIES	N	XML	Product categories
P_CATEGORY	Y	CHAR(200)	Product category
P_CATEGORY[channel]	N	CHAR(12)	Channel indicator
P_CATEGORY[type]	Y	ENUM	Type of the product category Possible values are: primary = primary channel category secondary = secondary channel category
P_CATEGORY[identifier]	N	ENUM	Identifier for referencing [24]
P_CATEGORY[key]	N	CHAR(50)	Value stored in the *identifier* assignment statement

Example-XML (key-based)

```
<P_CATEGORIES>
    <P_CATEGORY channel="chxx" type="primary" identifier="key" key="T0815" name="Markenshops|Leuchtenfirma" />
</P_CATEGORIES>
```

Example-XML (name-based)

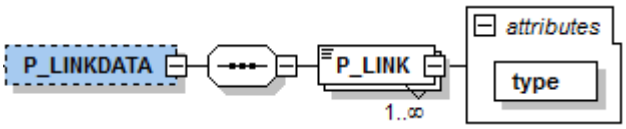
```
<P_CATEGORIES>
    <P_CATEGORY channel="chxx" type="primary" identifier="name" key="Markenshops|Leuchtenfirma" />
</P_CATEGORIES>
```

Example-XML (ID-based)

```
<P_CATEGORIES>
    <P_CATEGORY channel="chxx" type="primary" identifier="id" key="5" />
</P_CATEGORIES>
```

TBCATALOG / PRODUCTDATA / PRODUCT / P_LINKDATA

Product links are referenced by the product ID (P_NR) of the target product.



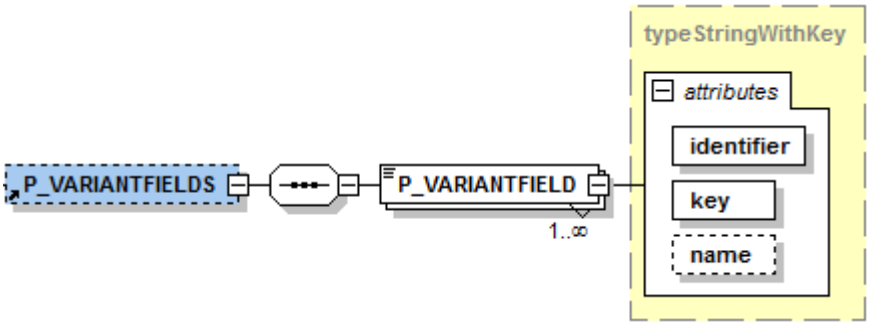
NODE	MUST	TYPE	DESCRIPTION
P_LINKDATA	N		Product links (cross-selling,..)
P_LINK	Y	CHAR(50)	Linked product ID (P_NR)
P_LINK[type]	Y	ENUM	Link types with description: cross – Combination option/further articles accessory – Accessory to the product device – Product to accessory upselling – Recommendation alternative – Alternative offer

Example-XML

```
<P_LINKDATA>
    <P_LINK type="cross">75611</P_LINK>
    <P_LINK type="accessory">75595</P_LINK>
</P_LINKDATA>
```

TBCATALOG / PRODUCTDATA / PRODUCT / P_VARIANTFIELDS

If a product includes more than one article, at least one variant dimension is specified. This node provides an overview of the variant dimensions used in the articles.



NODE	MUST	TYPE	DESCRIPTION
P_VARIANTFIELDS	N		Variant dimensions
P_VARIANTFIELD	Y	CHAR(50)	Name of the component
P_VARIANTFIELD [identifier]	Y	CHAR(20)	Identifier for referencing [24]
P_VARIANTFIELD [key]	Y	CHAR(50)	Value stored in the *identifier* assignment statement
P_VARIANTFIELD [name]	N	CHAR(100)	Name stored in the *identifier* assignment statement

Example-XML (key-based)

```
<P_VARIANTFIELDS>
    <P_VARIANTFIELD identifier="key" key="size" />
    <P_VARIANTFIELD identifier="key" key="color" />
</P_VARIANTFIELDS>
```

Example-XML (name-based)

```
<P_VARIANTFIELDS>
    <P_VARIANTFIELD identifier="name" key="Größe" />
    <P_VARIANTFIELD identifier="name" key="Farbe" />
</P_VARIANTFIELDS>
```

Example-XML (ID-based)

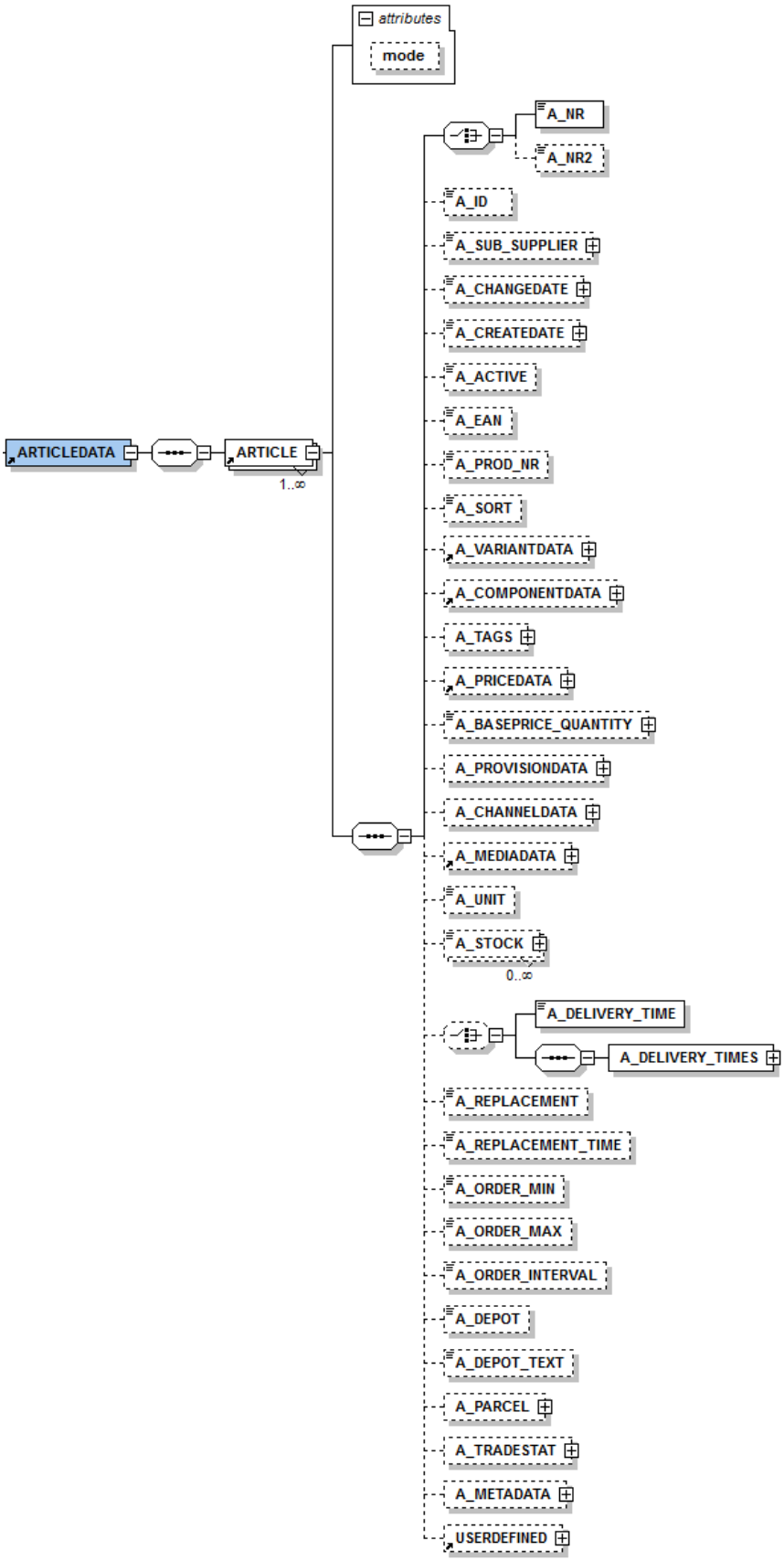
```
<P_VARIANTFIELDS>
        <P_VARIANTFIELD identifier="ID" key="20" />
        <P_VARIANTFIELD identifier="ID" key="21" />
</P_VARIANTFIELDS>
```

TBCATALOG / PRODUCTDATA / PRODUCT / USERDEFINED

User-defined fields if applicable.

TBCATALOG / PRODUCTDATA / PRODUCT / ARTICLEDATA

For each product, at least one article is transferred. If there is more than one article per product available (variants), then at least one variant-forming component (A_VARIANTDATA [61]) is defined.

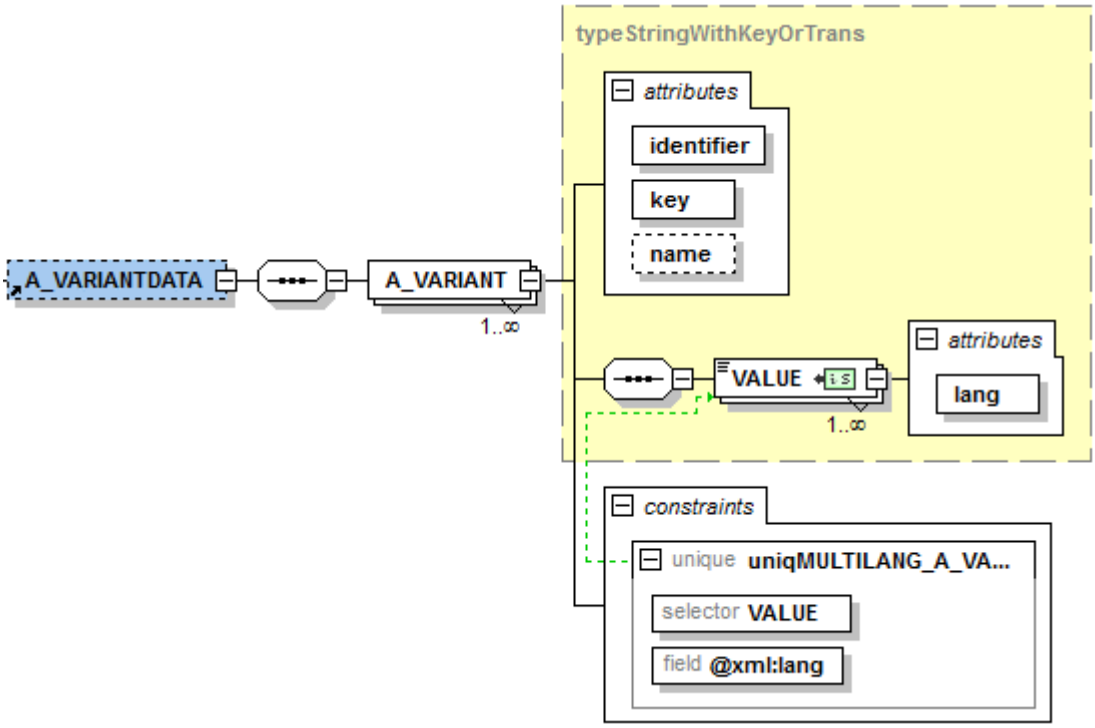


NODE	MUST	TYPE	DESCRIPTION
ARTICLEDATA	Y	array	Article data node
ARTICLE	Y	array	Data for an article
mode	N	string	If an article is to be deleted, this node contains <mode="delete">. The default value is updated.
A_NR	Y	CHAR(50)	Article number. This may change.
A_NR2	N	CHAR(50)	Additional/second article number
A_ID	N	INT	Unique article ID from TB.One
A_SUB_SUPPLIER	N	string	Information on the article supplier (If more than one supplier available)
A_CHANGEDATE	N	Unix	Unix timestamp of the last article change
A_CREATEDATE	N	Unix	Unix timestamp of the article creation
A_ACTIVE	N	INT(0/1)	Article active, always 1 Disabled articles are no longer provided.
A_EAN	N	INT(13) fixed	EAN code of the article; 13 digits
A_PROD_NR	N	CHAR(50)	Manufacturer article number
A_SORT	N	INT	Article sorting within the product
A_VARIANTDATA 61	N	XML	Variant values
A_COMPONENTDATA 63	N	XML	Article components
A_TAGS 65	N	XML	Article attributes

NODE	MUST	TYPE	DESCRIPTION
A_PRICEDATA 67	N	XML	Price and tax information
A_BASEPRICE_QUANTITY 69	N	INT	Reference data for base price calculation
A_PROVISIONDATA 70	N	XML	Commission information for channels
A_CHANNELDATA	N	XML	Channel article number
A_MEDIADATA 71	N	XML	Article media (e.g. images)
A_UNIT	N	ENUM	Unit of the article (the default is "ST" for "piece"). Possible values are currently ST, CM and QM. Floating point numbers are not supported at orders/stocks.
A_STOCK	N	INT	Stock
A_DELIVERY_TIME/S 72	N	INT	Delivery time in business days
A_REPLACEMENT	N	INT(1/0)	Indicator for replaceability (1 = replaceable)
A_REPLACEMENT_TIME	N	INT	Replacement time in business days
A_ORDER_MIN	N	INT	Definition of a minimum order quantity
A_ORDER_MAX	N	INT	Definition of a maximum order quantity
A_ORDER_INTERVAL	N	INT	Increments when entering an order quantity
A_PARCEL 73	N	XML	Information on delivery type and shipping dimensions
A_TRADESTAT 76	N	XML	Intrastat number and country of origin
A_METADATA	N	XML	Not used by default
USERDEFINED 57	N	XML	Custom fields on article level

TBCATALOG / ... / ARTICLEDATA / ARTICLE / A_VARIANTDATA

If more than one article per product is transferred (variants), there are at least one variant property and its values. All variants of a product necessarily have the same variant dimension.



NODE	MUST	TYPE	DESCRIPTION
A_VARIANTDATA	N	XML	Variant dimensions
A_VARIANT	Y	CHAR(255)	Value of the variant dimension
A_VARIANT[identifier]	Y	ENUM	Identifier for referencing [24]
A_VARIANT[key]	Y	CHAR(50)	Value stored in the *identifier* assignment statement
A_VARIANT[name]	N	CHAR(100)	Name stored in the *identifier* assignment statement
VALUE xml:lang="lang"	Y	TEXT	Definition of the language

Example-XML (key-based)

```
<A_VARIANTDATA>
	<A_VARIANT identifier="key" key="color" name="Farbe">
		<VALUE xml:lang="de-DE">Hellgrün</VALUE>
		<VALUE xml:lang="en-US">light green</VALUE>
	</A_VARIANT>
	<A_VARIANT identifier="key" key="size" name="Größe">
		<VALUE xml:lang="de-DE">XL</VALUE>
	</A_VARIANT>
</A_VARIANTDATA>
```

Example-XML (name-based)

```
<A_VARIANTDATA>
	<A_VARIANT identifier="name" key="Farbe">
		<VALUE xml:lang="de-DE">Hellgrün</VALUE>
		<VALUE xml:lang="en-US">light green</VALUE>
	</A_VARIANT>
</A_VARIANTDATA>
```

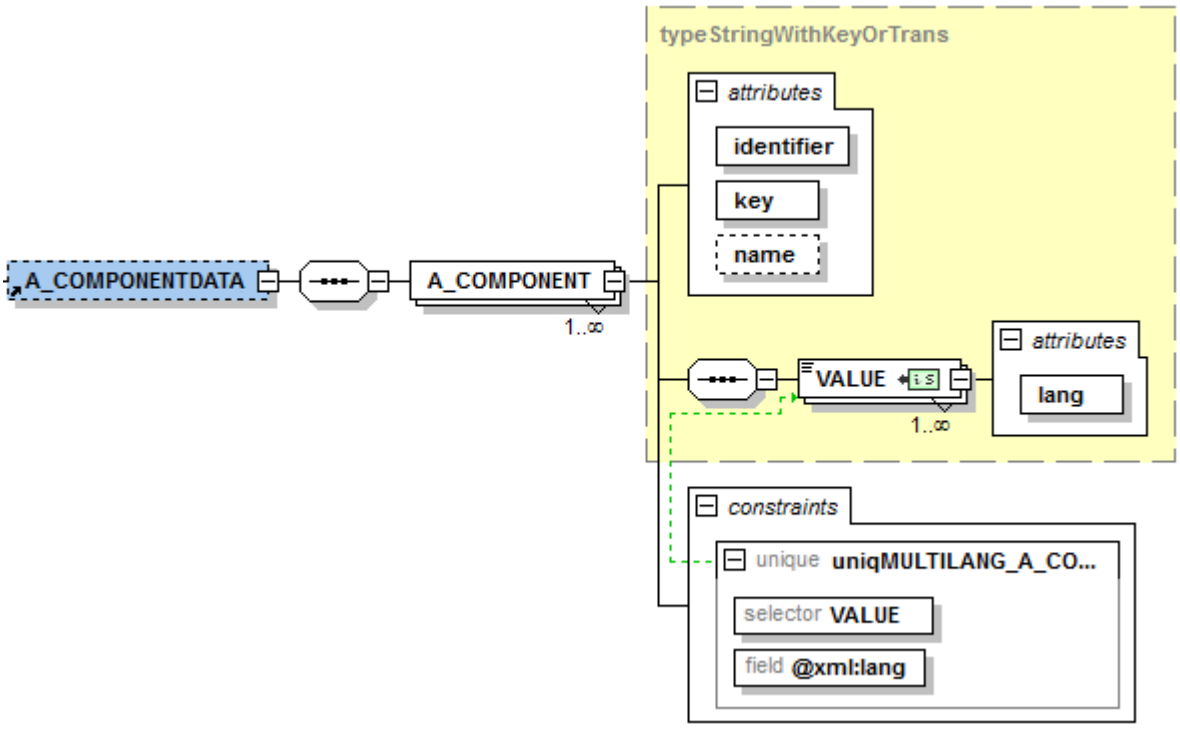
Example-XML (ID-based)

```
<A_VARIANTDATA>
	<A_VARIANT identifier="ID" key="26">
		<VALUE xml:lang="de-DE">Hellgrün</VALUE>
		<VALUE xml:lang="en-US">light green</VALUE>
	</A_VARIANT>
	<A_VARIANT identifier="ID" key="27">
		<VALUE xml:lang="de-DE">XL</VALUE>
	</A_VARIANT>
</A_VARIANTDATA>
```

TBCATALOG / ... / ARTICLEDATA / ARTICLE / A_COMPONENTDATA

Components on the article level (see also P_COMPONENTDATA [45]) differ for each article but do not form variants.

Example: *A USB flash drive has variants of memory size 32GB and 64GB. Depending on the memory size, the speed is different - 32 and 45MB/s. But there is no stick with 32GB and 45MB/s and no one with 64GB and 32MB/s. Here, the access speed could be stored as an article component.*



NODE	MUST	TYPE	DESCRIPTION
A_COMPONENTDATA	N	XML	Article components
A_COMPONENT	Y	TEXT	Definition of each article component
A_COMPONENT[identifier]	Y	ENUM	Identifier for referencing [24]
A_COMPONENT[key]	Y	CHAR(50)	Value stored in the *identifier* assignment statement
A_COMPONENT[name]	N	CHAR(100)	Name stored in the *identifier* assignment statement
VALUE xml:lang="lang"	Y	TEXT	Definition of the language

Example-XML (key-based)

```
<A_COMPONENTDATA>
	<A_COMPONENT identifier="key" key="readspeed" name=Geschwindigkeit">
		<VALUE xml:lang="de-DE">45 MB/s</VALUE>
		<VALUE xml:lang="en-US">45 MB/s</VALUE>
	</A_COMPONENT>
</A_COMPONENTDATA>
```

Example-XML (name-based)

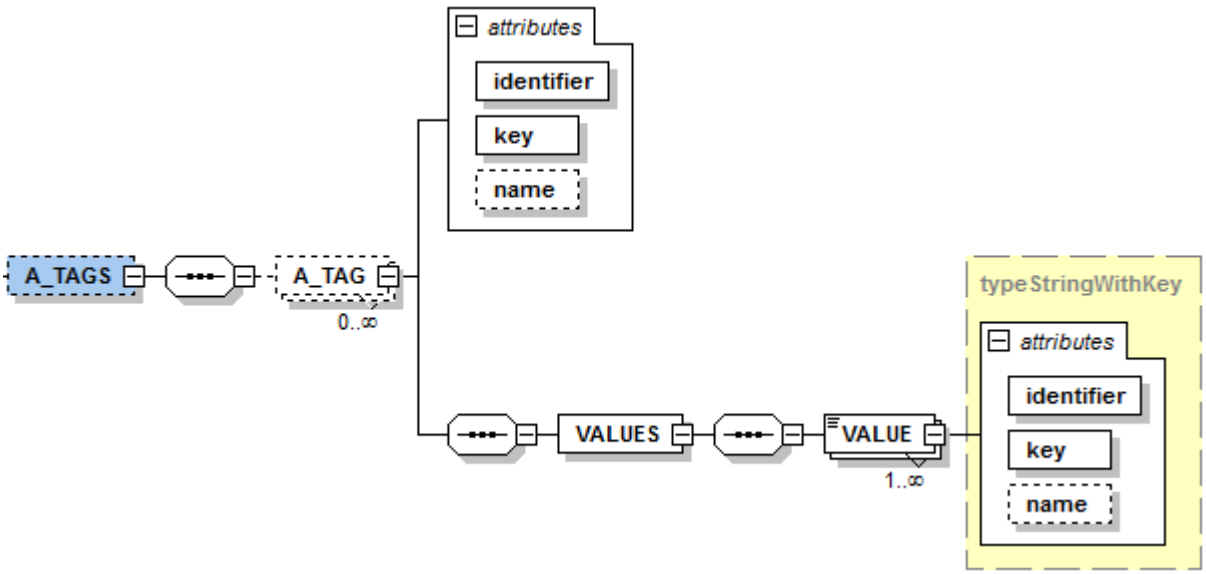
```
<A_COMPONENTDATA>
	<A_COMPONENT identifier="name" key="Geschwindigkeit">
		<VALUE xml:lang="de-DE">45 MB/s</VALUE>
		<VALUE xml:lang="en-US">45 MB/s</VALUE>
	</A_COMPONENT>
</A_COMPONENTDATA>
```

Example-XML (ID-based)

```
<A_COMPONENTDATA>
	<A_COMPONENT identifier="ID" key="30">
		<VALUE xml:lang="de-DE">45 MB/s</VALUE>
		<VALUE xml:lang="en-US">45 MB/s</VALUE>
	</A_COMPONENT>
</A_COMPONENTDATA>
```

TBCATALOG / ... / ARTICLEDATA / ARTICLE / A_TAGS

Attributes at article level (see also P_TAGS [47]).



NODE	MUST	TYPE	DESCRIPTION
A_TAGS	N	XML	Article attributes
A_TAG	Y	XML	Artikelattribut
A_TAG[identifier]	Y	ENUM	Identifier for referencing [24]
A_TAG[key]	Y	CHAR(50)	Value stored in the *identifier* assignment statementabe
A_TAG[name]	N	CHAR(100)	Name stored in the *identifier* assignment statement
VALUES	Y	XML	Attribute values
VALUE	Y	XML	Single attribute value
VALUE[identifier]	Y	ENUM	Identifier for referencing [24]
VALUE[key]	Y	CHAR(50)	Value stored in the *identifier* assignment statement

NODE	MUST	TYPE	DESCRIPTION
VALUE[name]	N	CHAR(100)	Name stored in the *identifier* assignment statement

Example-XML (key-based)

```
<A_TAGS>
	<A_TAG identifier="key" key="basecolor" name="Basisfarbe">
		<VALUES>
			<VALUE identifier="key" key="red" />
			<VALUE identifier="key" key="green" />
		</VALUES>
	</A_TAG>
</A_TAGS>
```

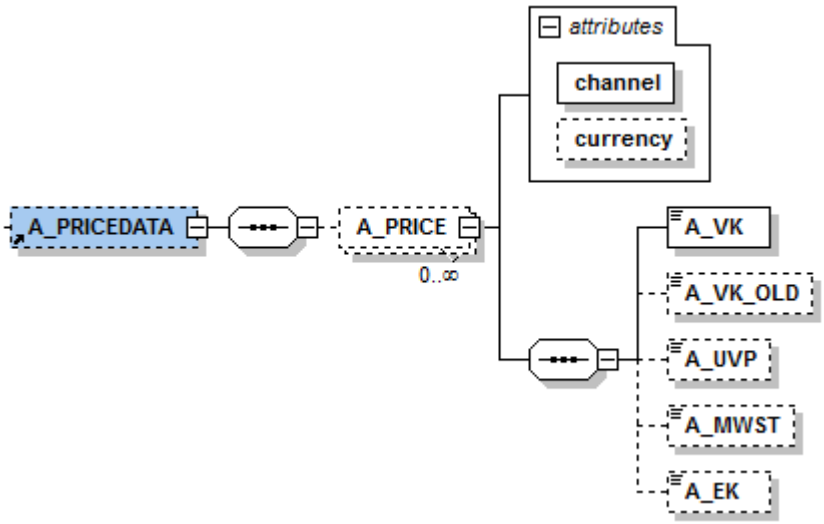
Example-XML (name-based)

```
<A_TAGS>
	<A_TAG identifier="name" key="Basisfarbe">
		<VALUES>
			<VALUE identifier="name" key="Rot" />
		</VALUES>
	</A_TAG>
</A_TAGS>
```

Example-XML (ID-based)

```
<A_TAGS>
	<A_TAG identifier="ID" key="40">
		<VALUES>
			<VALUE identifier="ID" key="60" />
		</VALUES>
	</A_TAG>
</A_TAGS>
```

TBCATALOG / ... / ARTICLEDATA / ARTICLE / A_PRICEDATA



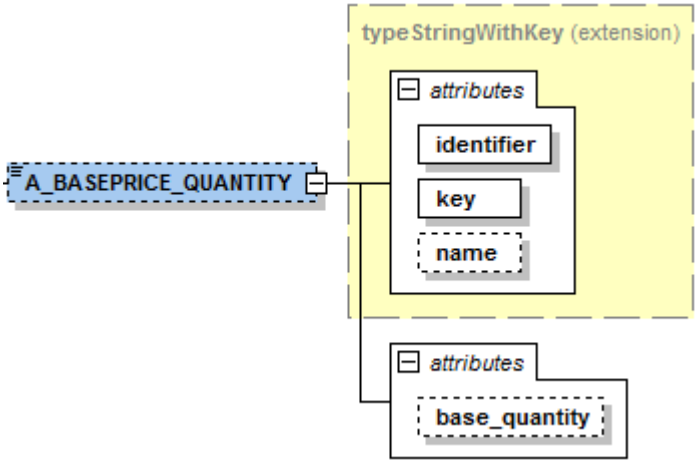
NODE	MUST	TYPE	DESCRIPTION
A_PRICEDATA	N		Article prices
A_PRICE	Y		Pricing
A_PRICE[channel]	Y	CHAR(12)	Channel sign
A_PRICE[currency]	N	ENUM	Currency sign
A_VK	Y	FLOAT	Gross retail price
A_VK_OLD	N	FLOAT	Former retail price
A_UVP	N	FLOAT	Manufacturer's suggested retail price (MSRP)
A_MWST	N	INT	Indicator for the VAT rate 1 = reduced tax rate 2 = normal tax rate 3 = VAT exempt
A_EK	N	FLOAT	Optional purchase price (net)

Example-XML

```
<A_PRICEDATA>
	<A_PRICE channel="chxx" currency="EUR">		<!-- Main channel  -->
		<A_VK>69.97</A_VK>
		<A_VK_OLD>89.95</A_VK_OLD>
		<A_UVP>99.95</A_UVP>
		<A_MWST>2</A_MWST>
		<A_EK>38.32</A_EK>
	</A_PRICE>
	<A_PRICE channel="chxx_de" currency="EUR">		<!-- Sub channel 1  -->
		<A_VK>69.97</A_VK>
		<A_VK_OLD>89.95</A_VK_OLD>
		<A_UVP>99.95</A_UVP>
		<A_MWST>2</A_MWST>
		<A_EK>38.32</A_EK>
	</A_PRICE>
	<A_PRICE channel="chxx_fr" currency="EUR">		<!-- Sub channel 1-->
		<A_VK>69.97</A_VK>
		<A_VK_OLD>89.95</A_VK_OLD>
		<A_UVP>99.95</A_UVP>
		<A_MWST>2</A_MWST>
		<A_EK>38.32</A_EK>
	</A_PRICE>
<A_PRICEDATA>
```

TBCATALOG / ... / ARTICLEDATA / ARTICLE / A_BASEPRICE_QUANTITY

Information about quantity per pack and base price reference for articles with mandatory base price.



NODE	MUST	TYPE	DESCRIPTION
A_BASEPRICE_QUANTITY	N	FLOAT	Fill quantity
A_BASEPRICE_QUANTITY [identifier]	Y	ENUM	Identifier for referencing [24]
A_BASEPRICE_QUANTITY [key]	Y	CHAR(50)	Value stored in the *identifier* assignment statement
A_BASEPRICE_QUANTITY [name]	N	CHAR(100)	Name stored in the *identifier* assignment statement
A_BASEPRICE_QUANTITY [base_quantity]	Y/N	INT	The amount to which the base price relates (without indicating the unit). **Only mandatory for key-based import**

Example-XML (key-based)

```
<A_BASEPRICE_QUANTITY identifier="key" key="ml" name="Preis/100ml"
base_quantity="100">500</A_BASEPRICE_QUANTITY>
```

Example-XML (name-based)

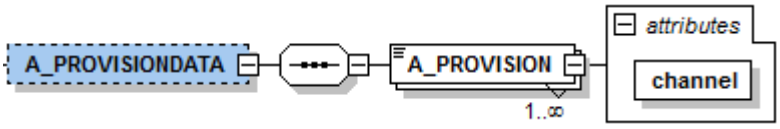
```
<A_BASEPRICE_QUANTITY identifier="name" key="100ml">500</A_BASEPRICE_QUANTITY>
```

Example-XML (ID-based)

```
<A_BASEPRICE_QUANTITY identifier="ID" key="02">500</A_BASEPRICE_QUANTITY>
```

TBCATALOG / ... / ARTICLEDATA / ARTICLE / A_PROVISIONDATA

If you agreed upon the use of commissions, a percentage or a corresponding indicator per article can be transmitted here.



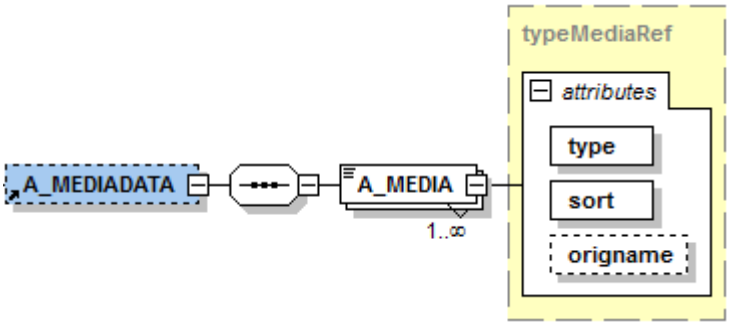
NODE	MUST	TYPE	DESCRIPTION
A_PROVISIONDATA	N		Commission information
A_PROVISION	Y	CHAR(50)	Import key of the commission rate from TB.One
A_PROVISION[channel]	Y	CHAR(12)	Channel indicator

Example-XML

```
<A_PROVISIONDATA>
        <A_PROVISION channel="chxx">Gruppe1</A_PROVISION>
        <A_PROVISION channel="chxx">PR00</A_PROVISION>
</A_PROVISIONDATA>
```

TBCATALOG / ... / ARTICLEDATA / ARTICLE / A_MEDIADATA

Media on article/variant level, usually related to media with a direct article reference (e.g. picture of a color variant). See also P_MEDIADATA 50.



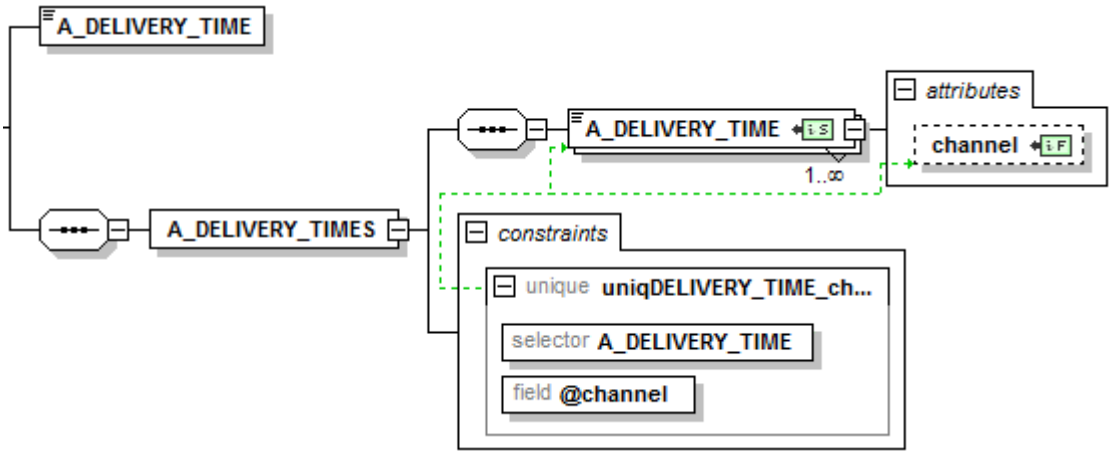
NODE	MUST	TYPE	DESCRIPTION
A_MEDIADATA	N	XML	Article media
A_MEDIA	Y	CHAR(255)	URL to article medium. If an image applies for several variants, the URL must be repeated for each variant.
A_MEDIA[type]	Y	ENUM	Media type
A_MEDIA[sort]	Y	INT	Sorting within the article media
A_MEDIA[origname]	N	CHAR(255)	Original name of the file

Example-XML

```
<A_MEDIADATA>
    <A_MEDIA type="image" sort="1">http://www.imageserver.tld/123.jpg</A_MEDIA>
    <A_MEDIA type="image" sort="2">http://www.imageserver.tld/124.jpg</A_MEDIA>
</A_MEDIADATA>
```

The details for media URLs in node P_MEDIADATA 50 apply accordingly.

TBCATALOG / ... / ARTICLEDATA / ARTICLE / A_DELIVERY_TIME(S)



NODE	MUST	TYPE	DESCRIPTION
A_DELIVERY_TIME	Y	INT	Delivery time for one target channel
Or A_DELIVERY_TIMES	Y	XML	Or delivery time node at multiple target channels
A_DELIVERY_TIME	Y	INT	Delivery time
A_DELIVERY_TIME[channel]	N	CHAR(12)	Channel indicator

Example-XML (1 delivery time):

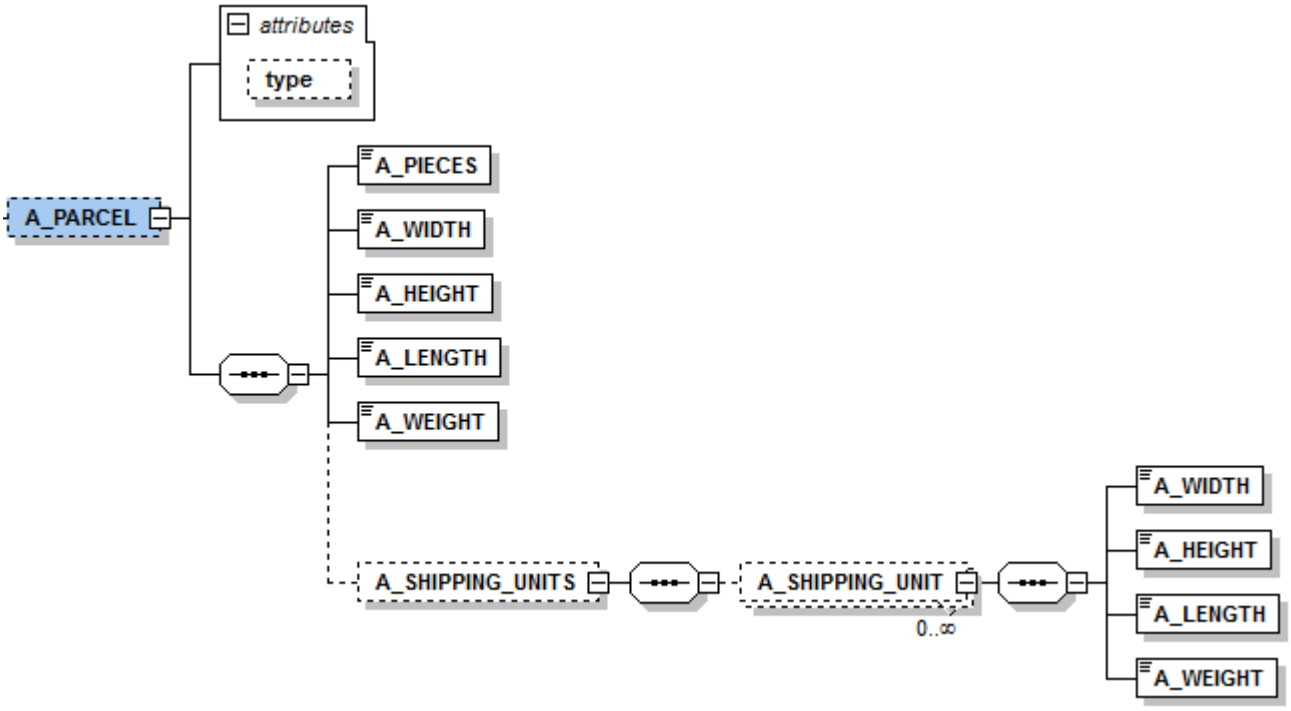
```
<A_DELIVERY_TIME>2</A_DELIVERY_TIME>
```

Example-XML (differing delivery time per sub-channel):

```
<A_DELIVERY_TIMES>
        <A_DELIVERY_TIME channel="chxx">2</A_DELIVERY_TIME>
        <A_DELIVERY_TIME channel="chxx_a">2</A_DELIVERY_TIME>
        <A_DELIVERY_TIME channel="chxx_b">4</A_DELIVERY_TIME>
        <A_DELIVERY_TIME channel="chxx_c">5</A_DELIVERY_TIME>
</A_DELIVERY_TIMES>
```

TBCATALOG / ... / ARTICLEDATA / ARTICLE / A_PARCEL

Information on modes of transport and package dimensions.



NODE	MUST	TYPE	DESCRIPTION
A_PARCEL	N		Information on shipping types and measurements
A_PARCEL[type]	N		Shipping type of the article
A_PIECES	Y	INT	Number of packages (more than 1 packing unit)
A_WIDTH	Y	INT	Width of the article in cm
A_HEIGHT	Y	INT	Height of the article in cm
A_LENGTH	Y	INT	Length of the article in cm
A_WEIGHT	Y	FLOAT	Weight of the article in kg
A_SHIPPING_UNIT	N	XML	For several packages: Packing dimensions per package
A_WIDTH	Y	INT	Width of the article in cm

NODE	MUST	TYPE	DESCRIPTION
A_HEIGHT	Y	INT	Height of the article in cm
A_LENGTH	Y	INT	Length of the article in cm
A_WEIGHT	Y	FLOAT	Weight of the article in kg

Example-XML (1 pack size):

```
<A_PARCEL type="other_parcel">
        <A_PIECES>1</A_PIECES>
        <A_WIDTH>90</A_WIDTH>
        <A_HEIGHT>30</A_HEIGHT>
        <A_LENGTH>50</A_LENGTH>
        <A_WEIGHT>2.4</A_WEIGHT>
</A_PARCEL>
```

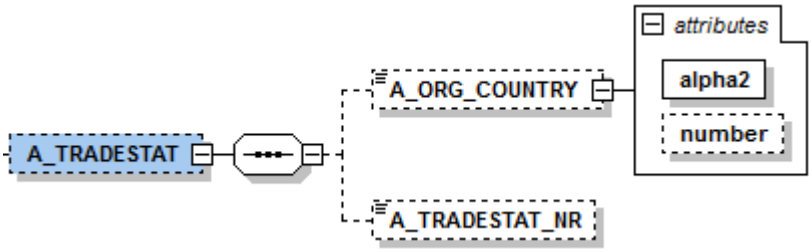
Example-XML (multiple pack sizes):

```
<A_PARCEL type="other_parcel">
        <A_PIECES>2</A_PIECES>
<A_WIDTH>38</A_WIDTH>
        <A_HEIGHT>5</A_HEIGHT>
        <A_LENGTH>28</A_LENGTH>
        <A_WEIGHT>1.7</A_WEIGHT>
        <A_SHIPPING_UNITS>
                <A_SHIPPING_UNIT>
                        <A_WIDTH>38</A_WIDTH>
                        <A_HEIGHT>2</A_HEIGHT>
                        <A_LENGTH>28</A_LENGTH>
                        <A_WEIGHT>0.85</A_WEIGHT>
                </A_SHIPPING_UNIT>
                <A_SHIPPING_UNIT>
                        <A_WIDTH>38</A_WIDTH>
                        <A_HEIGHT>2</A_HEIGHT>
```

```
                        <A_LENGTH>28</A_LENGTH>
                        <A_WEIGHT>0.85</A_WEIGHT>
                    </A_SHIPPING_UNIT>
                </A_SHIPPING_UNITS>
</A_PARCEL>
```

TBCATALOG / ... / ARTICLEDATA / ARTICLE / A_TRADESTAT

Information on country of origin and Intrastat number.



NODE	MUST	TYPE	DESCRIPTION
A_TRADESTAT	N	XML	Tradestat information
A_ORG_COUNTRY[alpha2]	Y	CHAR(2)	ISO 3166-1 Alpha 2 Code
A_TRADESTAT_NR	N	INT(8)	Intrastat number

Example-XML:

```
<A_TRADESTAT>
        <A_ORG_COUNTRY alpha2="DE" />
        <A_TRADESTAT_NR>94036090</A_TRADESTAT_NR>
</A_TRADESTAT>
```

10. EXAMPLES

Geben Sie hier den Text ein.

10.1 PRODUCT WITHOUT VARIANTS, MULTILINGUAL (KEY-BASED)

```
<?xml version="1.0" encoding="UTF-8"?>

<TBCATALOG version="1.3" creation="1378277984" channel="cudoku
xml:xsd="http://www.w3.org/2001/XMLSchema">

    <PRODUCTDATA type="full">

        <PRODUCT>

            <P_NR>87</P_NR>

            <P_NR_EXTERNAL>5986</P_NR_EXTERNAL>

            <P_CHANGEDATE type="unix_timestamp">1378275377</P_CHANGEDATE>

            <P_CREATEDATE type="unix_timestamp">1365784128</P_CREATEDATE>

            <P_ACTIVEDATA>

                <P_ACTIVE channel="cudokuacc1">1</P_ACTIVE>

            </P_ACTIVEDATA>

            <P_NAME_KEYWORD>

                <VALUE xml:lang="de-DE">Outdoor-Jacke</VALUE>

                <VALUE xml:lang="en-US">Outdoor-Jacket</VALUE>

            </P_NAME_KEYWORD>

            <P_NAME_PROPER>

                <VALUE xml:lang="de-DE">Reinhold</VALUE>

                <VALUE xml:lang="en-US">Reinhold</VALUE>

            </P_NAME_PROPER>

            <P_TEXT>

                <VALUE xml:lang="de-DE">

                    Die Outdoorjacke "Reinhold" ist eine außergewöhnlich robuste und funk-

                    tionale Jacke! Durch das innovative Material bietet sie eine enorme

                    Widerstandsfähigkeit und einen effektiven Schutz vor Wind und Witte-

                    rung, Kälte, Nässe und Sturm. Die Outdoorjacke "Reinhold" ist ein an-

                    passungsfähiger Allrounder, die Sie das ganze Jahr begeistern wird.

                </VALUE>
```

```
                <VALUE xml:lang="en-US">
                    The outdoor jacket "Reinhold" is an exceptionally robust and functio-
                    nal jacket! Through the innovative Material offers you a enormous
                    durability and an effective protection from wind and weather, cold,
                    wet and storm. The outdoor jacket "Reinhold" is a very flexible all-
                    rounder for the whole year, which will inspire you.
                </VALUE>
            </P_TEXT>
            <P_BRAND identifier="key" key="supjam" name="Superjackenmeister" />
            <P_KEYWORDS>
                <P_KEYWORD>
                    <VALUE xml:lang="de-DE">Kapuzenjacke</VALUE>
                    <VALUE xml:lang="en-US">hoodie jacket</VALUE>
                </P_KEYWORD>
                <P_KEYWORD>
                    <VALUE xml:lang="de-DE">Anorak</VALUE>
                    <VALUE xml:lang="en-US">Anorak</VALUE>
                </P_KEYWORD>
                <P_KEYWORD>
                    <VALUE xml:lang="de-DE">Funktionskleidung</VALUE>
                    <VALUE xml:lang="en-US">functional clothing</VALUE>
                </P_KEYWORD>
            </P_KEYWORDS>
            <P_COMPONENTDATA>
                <P_COMPONENT identifier="key" key="width" name="Breite">
                    <VALUE xml:lang="de-DE">110cm</VALUE>
                    <VALUE xml:lang="en-US">110cm</VALUE>
                </P_COMPONENT>
                <P_COMPONENT identifier="key" key="color" name="Farbe">
                    <VALUE xml:lang="de-DE">blau</VALUE>
                    <VALUE xml:lang="en-US">blue</VALUE>
                </P_COMPONENT>
```

```
</P_COMPONENTDATA>
<P_TAGS>
        <P_TAG identifier="key" key="gender" name="Zielgruppe">
                <VALUES>
                        <VALUE identifier="key" key="male" />
                </VALUES>
        </P_TAG>
</P_TAGS>
<P_BULLETS>
        <P_BULLET sort="10">
                <VALUE xml:lang="de-DE">atmungsaktiv</VALUE>
                <VALUE xml:lang="en-US">breathable</VALUE>
        </P_BULLET>
        <P_BULLET sort="20">
                <VALUE xml:lang="de-DE">wasserdicht</VALUE>
                <VALUE xml:lang="en-US">waterproof</VALUE>
        </P_BULLET>
        <P_BULLET sort="30">
                <VALUE xml:lang="de-DE">winddicht</VALUE>
                <VALUE xml:lang="en-US">windproof</VALUE>
        </P_BULLET>
</P_BULLETS>
<P_MEDIADATA>
        <P_MEDIA type="image" sort="0">http://foo.server/231.jpg</P_MEDIA>
        <P_MEDIA type="image" sort="1">http://foo.server/232.jpg</P_MEDIA>
</P_MEDIADATA>
<P_LOGOS>
        <P_LOGO identifier="key" key="machinewash" name="Maschinenwäsche" />
        <P_LOGO identifier="key" key="ironleft" name="links bügeln" />
</P_LOGOS>
<P_CATEGORIES>
        <P_CATEGORY channel="cudoku" type="primary" identifier="key" key="hefr"
```

```
                name="Herren|Jacken|Freizeit" />
        </P_CATEGORIES>
        <P_LINKDATA>
                <P_LINK type="cross">568</P_LINK>
                <P_LINK type="accessory">894</P_LINK>
        </P_LINKDATA>
        <ARTICLEDATA>
                <ARTICLE>
                        <A_NR>5986</A_NR>
                        <A_ID>105</A_ID>
                        <A_CHANGEDATE
                        type="unix_timestamp">1376396744</A_CHANGEDATE>
                        <A_CREATEDATE
                        type="unix_timestamp">1357648848</A_CREATEDATE>
                        <A_ACTIVE>1</A_ACTIVE>
                        <A_EAN>4250372205442</A_EAN>
                        <A_PROD_NR>56894</A_PROD_NR>
                        <A_NR2>381059</A_NR2>
                        <A_COMPONENTDATA>
                                <A_COMPONENT identifier="key" key="collar"
                                name="Ausschnitt">
                                        <VALUE xml:lang="de-DE">V-Ausschnitt</VALUE>
                                        <VALUE xml:lang="en-US">V-neck</VALUE>
                                </A_COMPONENT>
                        </A_COMPONENTDATA>
                        <A_TAGS>
                                <A_TAG identifier="key" key="basecolor" name="Basisfarbe">
                                        <VALUES>
                                                <VALUE identifier="key" key="blue"
                                                name="blau" />
                                        </VALUES>
                                </A_TAG>
```

```
</A_TAGS>
<A_PRICEDATA>
        <A_PRICE channel="cudoku" currency="EUR">
                <A_VK>87.99</A_VK>
                <A_VK_OLD>109.95</A_VK_OLD>
                <A_UVP>120.00</A_UVP>
                <A_MWST>2</A_MWST>
                <A_EK>59.90</A_EK>
        </A_PRICE>
        <A_PRICE channel="cudoku_en" currency="EUR">
                <A_VK>99.99</A_VK>
                <A_VK_OLD>109.95</A_VK_OLD>
                <A_UVP>120.00</A_UVP>
                <A_MWST>2</A_MWST>
                <A_EK>59.90</A_EK>
        </A_PRICE>
</A_PRICEDATA>
<A_MEDIADATA>
        <A_MEDIA type="image"
        sort="1">http://foo.server/blau.jpg</A_MEDIA>
        <A_MEDIA type="image"
        sort="2">http://foo.server/bldet.jpg</A_MEDIA>
</A_MEDIADATA>
<A_UNIT>ST</A_UNIT>
<A_STOCK>20</A_STOCK>
<A_DELIVERY_TIMES>
        <A_DELIVERY_TIME channel="cudoku">3</A_DELIVERY_TIME>
        <A_DELIVERY_TIME
        channel="cudoku_en">5</A_DELIVERY_TIME>
</A_DELIVERY_TIMES>
<A_REPLACEMENT>1</A_REPLACEMENT>
<A_REPLACEMENT_TIME>5</A_REPLACEMENT_TIME>
```

```
                    <A_ORDER_MIN>1</A_ORDER_MIN>
                    <A_ORDER_MAX>2</A_ORDER_MAX>
                    <A_ORDER_INTERVAL>1</A_ORDER_INTERVAL>
                    <A_PARCEL type="MeinVersand">
                        <A_PIECES>2</A_PIECES>
                        <A_WIDTH>90</A_WIDTH>
                        <A_HEIGHT>30</A_HEIGHT>
                        <A_LENGTH>50</A_LENGTH>
                        <A_WEIGHT>2.4</A_WEIGHT>
                        <A_SHIPPING_UNITS>
                            <A_SHIPPING_UNIT>
                                <A_WIDTH>90</A_WIDTH>
                                <A_HEIGHT>30</A_HEIGHT>
                                <A_LENGTH>50</A_LENGTH>
                                <A_WEIGHT>2.4</A_WEIGHT>
                            </A_SHIPPING_UNIT>
                            <A_SHIPPING_UNIT>
                                <A_WIDTH>90</A_WIDTH>
                                <A_HEIGHT>30</A_HEIGHT>
                                <A_LENGTH>50</A_LENGTH>
                                <A_WEIGHT>2.4</A_WEIGHT>
                            </A_SHIPPING_UNIT>
                        </A_SHIPPING_UNITS>
                    </A_PARCEL>
                    <A_TRADESTAT>
                        <A_ORG_COUNTRY alpha2="DE"/>
                        <A_TRADESTAT_NR>94036090</A_TRADESTAT_NR>
                    </A_TRADESTAT>
                </ARTICLE>
            </ARTICLEDATA>
        </PRODUCT>
    </PRODUCTDATA>
```

```
</TBCATALOG>
```

10.2 PRODUCT WITHOUT VARIANTS, MONOLINGUAL (KEY-BASED)

```
<?xml version="1.0" encoding="UTF-8"?>

<TBCATALOG version="1.3" creation="1378277984" channel="cudoku
xml:xsd="http://www.w3.org/2001/XMLSchema">

        <PRODUCTDATA type="full">

                <PRODUCT>

                        <P_NR>87</P_NR>

                        <P_NR_EXTERNAL>5986</P_NR_EXTERNAL>

                        <P_CHANGEDATE type="unix_timestamp">1378275377</P_CHANGEDATE>

                        <P_CREATEDATE type="unix_timestamp">1365784128</P_CREATEDATE>

                        <P_ACTIVEDATA>

                                <P_ACTIVE channel="cudoku">1</P_ACTIVE>

                        </P_ACTIVEDATA>

                        <P_NAME_KEYWORD>

                                <VALUE xml:lang="de-DE">Outdoor-Jacke</VALUE>

                        </P_NAME_KEYWORD>

                        <P_NAME_PROPER>

                                <VALUE xml:lang="de-DE">Reinhold</VALUE>

                        </P_NAME_PROPER>

                        <P_TEXT>

                                <VALUE xml:lang="de-DE">

                                        Die Outdoorjacke "Reinhold" ist eine außergewöhnlich robuste und funk-

                                        tionale Jacke! Durch das innovative Material bietet sie eine enorme

                                        Widerstandsfähigkeit und einen effektiven Schutz vor Wind und Witte-

                                        rung, Kälte, Nässe und Sturm. Die Outdoorjacke "Reinhold" ist ein an-

                                        passungsfähiger Allrounder, die Sie das ganze Jahr begeistern wird.

                                </VALUE>

                        </P_TEXT>

                        <P_BRAND identifier="key" key="supjam" name="Superjackenmeister" />

                        <P_KEYWORDS>

                                <P_KEYWORD>

                                        <VALUE xml:lang="de-DE">Kapuzenjacke</VALUE>
```

```
            </P_KEYWORD>
            <P_KEYWORD>
                <VALUE xml:lang="de-DE">Anorak</VALUE>
            </P_KEYWORD>
            <P_KEYWORD>
                <VALUE xml:lang="de-DE">Funktionskleidung</VALUE>
            </P_KEYWORD>
        </P_KEYWORDS>
        <P_COMPONENTDATA>
            <P_COMPONENT identifier="key" key="width" name="Breite">
                <VALUE xml:lang="de-DE">110cm</VALUE>
            </P_COMPONENT>
            <P_COMPONENT identifier="key" key="color" name="Farbe">
                <VALUE xml:lang="de-DE">blau</VALUE>
            </P_COMPONENT>
        </P_COMPONENTDATA>
        <P_TAGS>
            <P_TAG identifier="key" key="gender" name="Zielgruppe">
                <VALUES>
                    <VALUE identifier="key" key="male" />
                </VALUES>
            </P_TAG>
        </P_TAGS>
        <P_BULLETS>
            <P_BULLET sort="10">
                <VALUE xml:lang="de-DE">atmungsaktiv</VALUE>
            </P_BULLET>
            <P_BULLET sort="20">
                <VALUE xml:lang="de-DE">wasserdicht</VALUE>
            </P_BULLET>
            <P_BULLET sort="30">
                <VALUE xml:lang="de-DE">winddicht</VALUE>
```

```
            </P_BULLET>
        </P_BULLETS>
        <P_MEDIADATA>
            <P_MEDIA type="image" sort="0">http://foo.server/231.jpg</P_MEDIA>
            <P_MEDIA type="image" sort="1">http://foo.server/232.jpg</P_MEDIA>
        </P_MEDIADATA>
        <P_LOGOS>
            <P_LOGO identifier="key" key="machinewash" name="Maschinenwäsche" />
            <P_LOGO identifier="key" key="ironleft" name="links bügeln" />
        </P_LOGOS>
        <P_CATEGORIES>
            <P_CATEGORY channel="cudoku" type="primary" identifier="key" key="hefr"
            name="Herren|Jacken|Freizeit" />
            <P_CATEGORY channel="cudoku" type="secondary" identifier="key" key="heout"
            name="Herren|Jacken|Outdoor" />
        </P_CATEGORIES>
        <P_LINKDATA>
            <P_LINK type="cross">568</P_LINK>
            <P_LINK type="accessory">894</P_LINK>
        </P_LINKDATA>
        <ARTICLEDATA>
            <ARTICLE>
                <A_NR>5986</A_NR>
                <A_ID>105</A_ID>
                <A_CHANGEDATE
                type="unix_timestamp">1376396744</A_CHANGEDATE>
                <A_CREATEDATE
                type="unix_timestamp">1357648848</A_CREATEDATE>
                <A_ACTIVE>1</A_ACTIVE>
                <A_EAN>4250372205442</A_EAN>
                <A_PROD_NR>56894</A_PROD_NR>
                <A_NR2>381059</A_NR2>
```

```
<A_COMPONENTDATA>
    <A_COMPONENT identifier="key" key="collar"
    name="Ausschnitt">
        <VALUE xml:lang="de-DE">V-Ausschnitt</VALUE>
    </A_COMPONENT>
</A_COMPONENTDATA>
<A_TAGS>
    <A_TAG identifier="key" key="basecolor" name="Basisfarbe">
        <VALUES>
            <VALUE identifier="key" key="blue"
            name="blau" />
        </VALUES>
    </A_TAG>
</A_TAGS>
<A_PRICEDATA>
    <A_PRICE channel="cudoku" currency="EUR">
        <A_VK>87.99</A_VK>
        <A_VK_OLD>109.95</A_VK_OLD>
        <A_UVP>120.00</A_UVP>
        <A_MWST>2</A_MWST>
        <A_EK>59.90</A_EK>
    </A_PRICE>
    <A_PRICE channel="cudoku_en" currency="EUR">
        <A_VK>99.99</A_VK>
        <A_VK_OLD>109.95</A_VK_OLD>
        <A_UVP>120.00</A_UVP>
        <A_MWST>2</A_MWST>
        <A_EK>59.90</A_EK>
    </A_PRICE>
</A_PRICEDATA>
<A_MEDIADATA>
    <A_MEDIA type="image"
```

```
                sort="1">http://foo.server/blau.jpg</A_MEDIA>
                <A_MEDIA type="image"
                sort="2">http://foo.server/bldet.jpg</A_MEDIA>
        </A_MEDIADATA>
        <A_UNIT>ST</A_UNIT>
        <A_STOCK>20</A_STOCK>
        <A_DELIVERY_TIMES>
                <A_DELIVERY_TIME channel="cudoku">3</A_DELIVERY_TIME>
                <A_DELIVERY_TIME
                channel="cudoku_en">5</A_DELIVERY_TIME>
        </A_DELIVERY_TIMES>
        <A_REPLACEMENT>1</A_REPLACEMENT>
        <A_REPLACEMENT_TIME>5</A_REPLACEMENT_TIME>
        <A_ORDER_MIN>1</A_ORDER_MIN>
        <A_ORDER_MAX>2</A_ORDER_MAX>
        <A_ORDER_INTERVAL>1</A_ORDER_INTERVAL>
        <A_PARCEL type="MeinVersand">
        <A_PIECES>2</A_PIECES>
                <A_WIDTH>90</A_WIDTH>
                <A_HEIGHT>30</A_HEIGHT>
                <A_LENGTH>50</A_LENGTH>
                <A_WEIGHT>2.4</A_WEIGHT>
                <A_SHIPPING_UNITS>
                        <A_SHIPPING_UNIT>
                                <A_WIDTH>90</A_WIDTH>
                                <A_HEIGHT>30</A_HEIGHT>
                                <A_LENGTH>50</A_LENGTH>
                                <A_WEIGHT>2.4</A_WEIGHT>
                        </A_SHIPPING_UNIT>
                        <A_SHIPPING_UNIT>
                                <A_WIDTH>90</A_WIDTH>
                                <A_HEIGHT>30</A_HEIGHT>
```

```
                                        <A_LENGTH>50</A_LENGTH>
                                        <A_WEIGHT>2.4</A_WEIGHT>
                                    </A_SHIPPING_UNIT>
                                </A_SHIPPING_UNITS>
                            </A_PARCEL>
                            <A_TRADESTAT>
                                <A_ORG_COUNTRY alpha2="DE"/>
                                <A_TRADESTAT_NR>94036090</A_TRADESTAT_NR>
                            </A_TRADESTAT>
                        </ARTICLE>
                    </ARTICLEDATA>
                </PRODUCT>
            </PRODUCTDATA>
</TBCATALOG>
```

10.3 PRODUCT WITHOUT VARIANTS, MULTILINGUAL (ID-BASED)

```
<?xml version="1.0" encoding="UTF-8"?>

<TBCATALOG version="1.3" creation="1378277984" channel="cudoku
xml:xsd="http://www.w3.org/2001/XMLSchema">

    <PRODUCTDATA type="full">

        <PRODUCT>

            <P_NR>87</P_NR>

            <P_NR_EXTERNAL>5986</P_NR_EXTERNAL>

            <P_CHANGEDATE type="unix_timestamp">1378275377</P_CHANGEDATE>

            <P_CREATEDATE type="unix_timestamp">1365784128</P_CREATEDATE>

            <P_ACTIVEDATA>

                <P_ACTIVE channel="cudoku">1</P_ACTIVE>

            </P_ACTIVEDATA>

            <P_NAME_KEYWORD>

                <VALUE xml:lang="de-DE">Outdoor-Jacke</VALUE>

                <VALUE xml:lang="en-US">Outdoor-Jacket</VALUE>

            </P_NAME_KEYWORD>

            <P_NAME_PROPER>

                <VALUE xml:lang="de-DE">Reinhold</VALUE>

                <VALUE xml:lang="en-US">Reinhold</VALUE>

            </P_NAME_PROPER>

            <P_TEXT>

                <VALUE xml:lang="de-DE">

                    Die Outdoorjacke "Reinhold" ist eine außergewöhnlich robuste und funk-
                    tionale Jacke! Durch das innovative Material bietet sie eine enorme
                    Widerstandsfähigkeit und einen effektiven Schutz vor Wind und Witte-
                    rung, Kälte, Nässe und Sturm. Die Outdoorjacke "Reinhold" ist ein an-
                    passungsfähiger Allrounder, die Sie das ganze Jahr begeistern wird.

                </VALUE>

                <VALUE xml:lang="en-US">

                    The outdoor jacket "Reinhold" is an exceptionally robust and functio-
                    nal jacket! Through the innovative Material offers you a enormous
```

```
                durability and an effective protection from wind and weather, cold,
                wet and storm. The outdoor jacket "Reinhold" is a very flexible all-
                rounder for the whole year, which will inspire you.
            </VALUE>
        </P_TEXT>
        <P_BRAND identifier="ID" key="4711" />
        <P_KEYWORDS>
            <P_KEYWORD>
                <VALUE xml:lang="de-DE">Kapuzenjacke</VALUE>
                <VALUE xml:lang="en-US">hoodie jacket</VALUE>
            </P_KEYWORD>
            <P_KEYWORD>
                <VALUE xml:lang="de-DE">Anorak</VALUE>
                <VALUE xml:lang="en-US">Anorak</VALUE>
            </P_KEYWORD>
            <P_KEYWORD>
                <VALUE xml:lang="de-DE">Funktionskleidung</VALUE>
                <VALUE xml:lang="en-US">functional clothing</VALUE>
            </P_KEYWORD>
        </P_KEYWORDS>
        <P_COMPONENTDATA>
            <P_COMPONENT identifier="ID" key="4712">
                <VALUE xml:lang="de-DE">110cm</VALUE>
                <VALUE xml:lang="en-US">110cm</VALUE>
            </P_COMPONENT>
            <P_COMPONENT identifier="ID" key="4713">
                <VALUE xml:lang="de-DE">blau</VALUE>
                <VALUE xml:lang="en-US">blue</VALUE>
            </P_COMPONENT>
        </P_COMPONENTDATA>
        <P_TAGS>
            <P_TAG identifier="ID" key="4714">
```

```
                    <VALUES>
                        <VALUE identifier="ID" key="4715" />
                    </VALUES>
                </P_TAG>
            </P_TAGS>
            <P_BULLETS>
                <P_BULLET sort="10">
                    <VALUE xml:lang="de-DE">atmungsaktiv</VALUE>
                    <VALUE xml:lang="en-US">breathable</VALUE>
                </P_BULLET>
                <P_BULLET sort="20">
                    <VALUE xml:lang="de-DE">wasserdicht</VALUE>
                    <VALUE xml:lang="en-US">waterproof</VALUE>
                </P_BULLET>
                <P_BULLET sort="30">
                    <VALUE xml:lang="de-DE">winddicht</VALUE>
                    <VALUE xml:lang="en-US">windproof</VALUE>
                </P_BULLET>
            </P_BULLETS>
            <P_MEDIADATA>
                <P_MEDIA type="image" sort="0">http://foo.server/231.jpg</P_MEDIA>
                <P_MEDIA type="image" sort="1">http://foo.server/232.jpg</P_MEDIA>
            </P_MEDIADATA>
            <P_LOGOS>
                <P_LOGO identifier="ID" key="4716" />
                <P_LOGO identifier="ID" key="4717" />
            </P_LOGOS>
            <P_CATEGORIES>
                <P_CATEGORY channel="cudoku" type="primary" identifier="ID" key="4718" />
                <P_CATEGORY channel="cudoku" type="secondary" identifier="ID"
                key="1295" />
            </P_CATEGORIES>
```

```
<P_LINKDATA>
        <P_LINK type="cross">568</P_LINK>
        <P_LINK type="accessory">894</P_LINK>
</P_LINKDATA>
<ARTICLEDATA>
        <ARTICLE>
                <A_NR>5986</A_NR>
                <A_ID>105</A_ID>
                <A_CHANGEDATE
                type="unix_timestamp">1376396744</A_CHANGEDATE>
                <A_CREATEDATE
                type="unix_timestamp">1357648848</A_CREATEDATE>
                <A_ACTIVE>1</A_ACTIVE>
                <A_EAN>4250372205442</A_EAN>
                <A_PROD_NR>56894</A_PROD_NR>
                <A_NR2>381059</A_NR2>
                <A_COMPONENTDATA>
                        <A_COMPONENT identifier="ID" key="1299">
                                <VALUE xml:lang="de-DE">V-Ausschnitt</VALUE>
                                <VALUE xml:lang="en-US">V-neck</VALUE>
                        </A_COMPONENT>
                </A_COMPONENTDATA>
                <A_TAGS>
                        <A_TAG identifier="ID" key="4589">
                                <VALUES>
                                        <VALUE identifier="ID" key="4689" />
                                </VALUES>
                        </A_TAG>
                </A_TAGS>
                <A_PRICEDATA>
                        <A_PRICE channel="cudoku" currency="EUR">
                                <A_VK>87.99</A_VK>
```

```
                    <A_VK_OLD>109.95</A_VK_OLD>
                    <A_UVP>120.00</A_UVP>
                    <A_MWST>2</A_MWST>
                    <A_EK>59.90</A_EK>
                </A_PRICE>
            </A_PRICEDATA>
            <A_MEDIADATA>
                <A_MEDIA type="image"
                sort="1">http://foo.server/blau.jpg</A_MEDIA>
                <A_MEDIA type="image"
                sort="2">http://foo.server/bldet.jpg</A_MEDIA>
            </A_MEDIADATA>
            <A_UNIT>ST</A_UNIT>
            <A_STOCK>20</A_STOCK>
            <A_DELIVERY_TIME>3</A_DELIVERY_TIME>
            <A_REPLACEMENT>1</A_REPLACEMENT>
            <A_REPLACEMENT_TIME>5</A_REPLACEMENT_TIME>
            <A_ORDER_MIN>1</A_ORDER_MIN>
            <A_ORDER_MAX>2</A_ORDER_MAX>
            <A_ORDER_INTERVAL>1</A_ORDER_INTERVAL>
            <A_PARCEL type="MeinVersand">
                <A_PIECES>2</A_PIECES>
                <A_WIDTH>90</A_WIDTH>
                <A_HEIGHT>30</A_HEIGHT>
                <A_LENGTH>50</A_LENGTH>
                <A_WEIGHT>2.4</A_WEIGHT>
                <A_SHIPPING_UNITS>
                    <A_SHIPPING_UNIT>
                        <A_WIDTH>90</A_WIDTH>
                        <A_HEIGHT>30</A_HEIGHT>
                        <A_LENGTH>50</A_LENGTH>
                        <A_WEIGHT>2.4</A_WEIGHT>
```

```
                                        </A_SHIPPING_UNIT>
                                        <A_SHIPPING_UNIT>
                                            <A_WIDTH>90</A_WIDTH>
                                            <A_HEIGHT>30</A_HEIGHT>
                                            <A_LENGTH>50</A_LENGTH>
                                            <A_WEIGHT>2.4</A_WEIGHT>
                                        </A_SHIPPING_UNIT>
                                    </A_SHIPPING_UNITS>
                                </A_PARCEL>
                                <A_TRADESTAT>
                                    <A_ORG_COUNTRY alpha2="DE"/>
                                    <A_TRADESTAT_NR>94036090</A_TRADESTAT_NR>
                                </A_TRADESTAT>
                            </ARTICLE>
                        </ARTICLEDATA>
                    </PRODUCT>
                </PRODUCTDATA>
</TBCATALOG>
```

10.4 PRODUCT WITH VARIANTS, MONOLINGUAL (ID-BASED)

```
<?xml version="1.0" encoding="UTF-8"?>
<TBCATALOG version="1.3" creation="1378277984" channel="cudoku
xml:xsd="http://www.w3.org/2001/XMLSchema">
    <PRODUCTDATA type="full">
        <PRODUCT>
            <P_NR>87</P_NR>
            <P_NR_EXTERNAL>5986</P_NR_EXTERNAL>
            <P_CHANGEDATE type="unix_timestamp">1378275377</P_CHANGEDATE>
            <P_CREATEDATE type="unix_timestamp">1365784128</P_CREATEDATE>
            <P_ACTIVEDATA>
                <P_ACTIVE channel="cudoku">1</P_ACTIVE>
```

```
</P_ACTIVEDATA>
<P_NAME_KEYWORD>
        <VALUE xml:lang="de-DE">Outdoor-Jacke</VALUE>
</P_NAME_KEYWORD>
<P_NAME_PROPER>
        <VALUE xml:lang="de-DE">Reinhold</VALUE>
</P_NAME_PROPER>
<P_TEXT>
        <VALUE xml:lang="de-DE">
                Die Outdoorjacke "Reinhold" ist eine außergewöhnlich robuste und funk-
                tionale Jacke! Durch das innovative Material bietet sie eine enorme
                Widerstandsfähigkeit und einen effektiven Schutz vor Wind und Witte-
                rung, Kälte, Nässe und Sturm. Die Outdoorjacke "Reinhold" ist ein an-
                passungsfähiger Allrounder, die Sie das ganze Jahr begeistern wird.
        </VALUE>
</P_TEXT>
<P_BRAND identifier="ID" key="4711" />
<P_KEYWORDS>
        <P_KEYWORD>
                <VALUE xml:lang="de-DE">Kapuzenjacke</VALUE>
        </P_KEYWORD>
        <P_KEYWORD>
                <VALUE xml:lang="de-DE">Anorak</VALUE>
        </P_KEYWORD>
        <P_KEYWORD>
                <VALUE xml:lang="de-DE">Funktionskleidung</VALUE>
        </P_KEYWORD>
</P_KEYWORDS>
<P_COMPONENTDATA>
        <P_COMPONENT identifier="ID" key="4712">
                <VALUE xml:lang="de-DE">110cm</VALUE>
        </P_COMPONENT>
```

```
            <P_COMPONENT identifier="ID" key="4713">
                <VALUE xml:lang="de-DE">blau</VALUE>
            </P_COMPONENT>
        </P_COMPONENTDATA>
        <P_TAGS>
            <P_TAG identifier="ID" key="4714">
                <VALUES>
                    <VALUE identifier="ID" key="4715" />
                </VALUES>
            </P_TAG>
        </P_TAGS>
        <P_BULLETS>
            <P_BULLET sort="10">
                <VALUE xml:lang="de-DE">atmungsaktiv</VALUE>
            </P_BULLET>
            <P_BULLET sort="20">
                <VALUE xml:lang="de-DE">wasserdicht</VALUE>
            </P_BULLET>
            <P_BULLET sort="30">
                <VALUE xml:lang="de-DE">winddicht</VALUE>
            </P_BULLET>
        </P_BULLETS>
        <P_MEDIADATA>
            <P_MEDIA type="image" sort="0">http://foo.server/231.jpg</P_MEDIA>
        </P_MEDIADATA>
        <P_LOGOS>
            <P_LOGO identifier="ID" key="4716" />
            <P_LOGO identifier="ID" key="4717" />
        </P_LOGOS>
        <P_CATEGORIES>
            <P_CATEGORY channel="cudoku" type="primary" identifier="ID" key="4718" />
        </P_CATEGORIES>
```

```
<P_LINKDATA>
        <P_LINK type="cross">568</P_LINK>
        <P_LINK type="accessory">894</P_LINK>
</P_LINKDATA>
<ARTICLEDATA>
        <ARTICLE>
                <A_NR>5986</A_NR>
                <A_ID>105</A_ID>
                <A_CHANGEDATE
                type="unix_timestamp">1376396744</A_CHANGEDATE>
                <A_CREATEDATE
                type="unix_timestamp">1357648848</A_CREATEDATE>
                <A_ACTIVE>1</A_ACTIVE>
                <A_EAN>4250372205442</A_EAN>
                <A_PROD_NR>56894</A_PROD_NR>
                <A_NR2>381059</A_NR2>
                <A_COMPONENTDATA>
                        <A_COMPONENT identifier="ID" key="1299">
                                <VALUE xml:lang="de-DE">V-Ausschnitt</VALUE>
                        </A_COMPONENT>
                </A_COMPONENTDATA>
                <A_TAGS>
                        <A_TAG identifier="ID" key="4589">
                                <VALUES>
                                        <VALUE identifier="ID" key="4689" />
                                </VALUES>
                        </A_TAG>
                </A_TAGS>
                <A_PRICEDATA>
                        <A_PRICE channel="cudoku" currency="EUR">
                                <A_VK>87.99</A_VK>
                                <A_VK_OLD>109.95</A_VK_OLD>
```

```
                    <A_UVP>120.00</A_UVP>
                    <A_MWST>2</A_MWST>
                    <A_EK>59.90</A_EK>
                </A_PRICE>
            </A_PRICEDATA>
            <A_MEDIADATA>
                <A_MEDIA type="image"
                sort="1">http://foo.server/blau.jpg</A_MEDIA>
            </A_MEDIADATA>
            <A_UNIT>ST</A_UNIT>
            <A_STOCK>20</A_STOCK>
            <A_DELIVERY_TIME>3</A_DELIVERY_TIME>
            <A_REPLACEMENT>1</A_REPLACEMENT>
            <A_REPLACEMENT_TIME>5</A_REPLACEMENT_TIME>
            <A_ORDER_MIN>1</A_ORDER_MIN>
            <A_ORDER_MAX>2</A_ORDER_MAX>
            <A_ORDER_INTERVAL>1</A_ORDER_INTERVAL>
            <A_PARCEL type="MeinVersand">
                <A_PIECES>2</A_PIECES>
                <A_WIDTH>90</A_WIDTH>
                <A_HEIGHT>30</A_HEIGHT>
                <A_LENGTH>50</A_LENGTH>
                <A_WEIGHT>2.4</A_WEIGHT>
                <A_SHIPPING_UNITS>
                    <A_SHIPPING_UNIT>
                        <A_WIDTH>90</A_WIDTH>
                        <A_HEIGHT>30</A_HEIGHT>
                        <A_LENGTH>50</A_LENGTH>
                        <A_WEIGHT>2.4</A_WEIGHT>
                    </A_SHIPPING_UNIT>
                    <A_SHIPPING_UNIT>
                        <A_WIDTH>90</A_WIDTH>
```

```
                                                    <A_HEIGHT>30</A_HEIGHT>
                                                    <A_LENGTH>50</A_LENGTH>
                                                    <A_WEIGHT>2.4</A_WEIGHT>
                                                </A_SHIPPING_UNIT>
                                            </A_SHIPPING_UNITS>
                                        </A_PARCEL>
                                        <A_TRADESTAT>
                                            <A_ORG_COUNTRY alpha2="DE"/>
                                            <A_TRADESTAT_NR>94036090</A_TRADESTAT_NR>
                                        </A_TRADESTAT>
                                    </ARTICLE>
                                </ARTICLEDATA>
                            </PRODUCT>
                        </PRODUCTDATA>
</TBCATALOG>
```