

TB.Market

TB.One



TB.Market

TB.One

TB.Cat 1.3: Classification Push

Structure Definition of the Channel

Version 1.0

Change History

Version	Valid as of	Author	Update/Notes/Documents History
1.0	29.08.2014	Christine Czerwinski	First issue (adjustment DocNo.1003 to TB.Cat 1.3)

Copyright © Tradebyte Software GmbH. All Rights Reserved.

This document contains proprietary information of Tradebyte and may not be disclosed or used except in accordance with applicable agreements. This material is protected by copyright laws. This material may not be reproduced, distributed, or altered in any manner by any entity without the written consent of Tradebyte and the owner of this material, unless in accordance with applicable agreements, contracts or licenses.

For permission to reproduce or distribute, please contact:
 Tradebyte Software GmbH, Bahnhofspatz 8, D-91522 Ansbach
 E-mail: Support@Tradebyte.com
<http://www.tradebyte.com>

Note

It was made every effort to ensure that the information contained in this document is complete and correct at the time of publication. The right to change the information remains reserved.

This customer document describes all the functions known today. It is possible that some functions are described that are not available to the customer. The exact functionality is dependent on the selected edition and the current release.

Trademarks

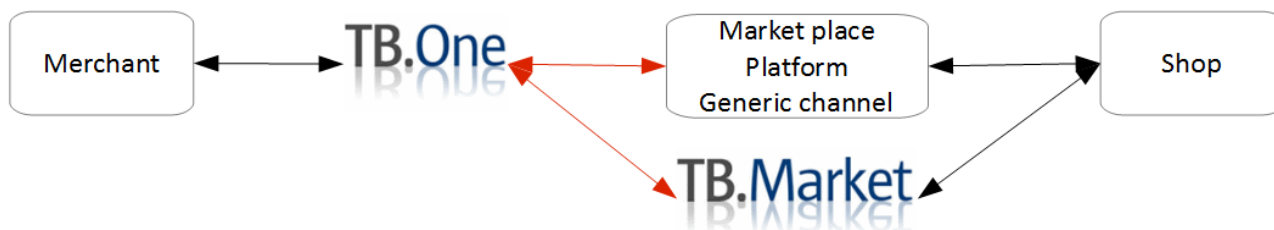
Microsoft, Windows and Excel are registered trademarks of Microsoft Inc., USA.
 Portable Document Format (PDF) and PostScript are trademarks of Adobe Systems, Inc.
 All other company names and logos and brand or product names are trademarks or registered trademarks of their respective owners.

Table of Contents

1	Aim and Purpose of the Document	4
2	Definitions and Conventions	4
2.1	Channel	4
2.2	Structure	4
2.3	Conventions	4
3	Data exchange via REST	4
3.1	Call the REST interface	5
3.2	Required Fields.....	5
3.3	Return Values	5
3.4	Classification Push.....	5
4	Referencing of data	5
4.1	Key.....	6
4.2	Name	6
5	Structure of the TB.CAT Classification	8
	TBCATALOG / CLASSIFICATION	8
	TBCATALOG / CLASSIFICATION / STRUCTURE	9
	TBCATALOG / CLASSIFICATION / STRUCTURE / STRUCTURE_ITEM	10
	TBCATALOG / ... / STRUCTURE_ITEM / ATTRIBUTE_SET	11
	TBCATALOG / CLASSIFICATION / COMPONENTS	12
	TBCATALOG / CLASSIFICATION / COMPONENTS / DATA_TYPE	13
	TBCATALOG / CLASSIFICATION / COMPONENTS / COMPONENT_LISTS	14
	TBCATALOG / CLASSIFICATION / TAGS	15
	TBCATALOG / CLASSIFICATION / BRANDS	16
	TBCATALOG / CLASSIFICATION / LOGOS.....	17
	TBCATALOG / CLASSIFICATION / ATTRIBUTE_SETS	18
	TBCATALOG / ... / ATTRIBUTE_SETS / ATTRIBUTE_SET_ITEM.....	19
	TBCATALOG / CLASSIFICATION / LANGUAGES.....	21
	TBCATALOG / CLASSIFICATION / SHIPPING_GROUPS.....	22
6	Example Classification	23
6.1	Flat Structure.....	23
6.2	Tree Structure	25

1 Aim and Purpose of the Document

This document is intended for operators of marketplaces, platforms and TB.Market and describes the transfer of categories and mandatory values to a connected TB.One.



2 Definitions and Conventions

2.1 Channel

As a channel we call each distribution channel; there is no distinction between marketplaces, platforms, and a generic channel (e.g. own shop). As a channel operator, you will receive a channel ID (Channel_ID) and a channel sign from Tradebyte through which you identify yourself when communicating. You will receive the channel ID and the channel sign from your Tradebyte contact.

2.2 Structure

The connection of TB.One/TB.Market to a channel requires that you as a channel operator define a structure (e.g. categories or commodity groups) to which the offered products/articles must be assigned (by the data provider). This structure is created as part of a so-called classification which may not only contain the structure elements (such as categories or commodity groups) of your channel but also the definition of mandatory fields for components, attributes and brands. Within the classification you determine which mandatory information you expect from your suppliers, i.e. the connected TB.One clients. Mandatory fields are defined by means of attribute sets. These attribute sets can either be set globally across all categories or may be different for each category.

The provision to TB.Market or TB.One clients generally takes place in an XML file using the REST interface.

2.3 Conventions

Gray lines in the field descriptions may not be delivered. These are either system nodes, which are generated by TB.One, or currently unused nodes.

3 Data exchange via REST

The data exchange between the channel and TB.One takes place by means of an XML file that contains your "classification". The transmission takes place via a REST interface. Whether you receive data from a TB.One or send it, you must validate it against the current TB.CAT data schema. Your Tradebyte representative will advise you about the model to be used. The validation against the current scheme must be checked at each sending/reception, so that any updates can be intercepted. You can find the schema files in our server at

https://api.trade-server.net/schema/all_in_one/tb-cat_1_3_all_in_one.xsd

By means of the REST interface, you can transfer your XML file directly to your TB.Market/TB.One.

3.1 Call the REST interface

By means of the REST interface, you can transfer your classification file directly to your TB.Market/TB.One. After the successful PUT, the classification is valid immediately. Please note that due to new requirements/definitions in a classification (e.g. a new mandatory field) articles may no longer can be exported as long as this duty (value mapping) is not met by the data supplier. Basic structure of the REST URI:

```
https://rest.trade-server.net/ACCOUNT-NR/TARGET/?channel=CHANNEL_ID
```

The REST interface call requires the authentication by user name and password.

3.2 Required Fields

Field	Description
USR/PWD	Username and password for the REST access
ACCOUNT-NR	Merchant number for which the classification is to apply.
TARGET	Specifies the resource group where the call aims to, in this case products (products)
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.

3.3 Return Values

Basically, a REST call always 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 respective content, you must take into account the http status code (header) response.

- 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

3.4 Classification Push

Method	PUT	Send
Target	products	valid for the products
Action	classification	a classification file in TB.Cat XML format to TB.One that is valid for the specified channel and which serves the primary categorization of the products at product data creation. By means of the classification, categories and values are specified where the suppliers need to map their own values to.
Caller	Channel	
Example	<pre>...ACCOUNT-NR/products/classification?channel=CHID</pre> Required XML file: see chapter 6.	
Response	200 OK	

4 Referencing of data

The data transmitted can be exported with different types of referencing. The main distinction is the way in which this information can then be re-interpreted or are to be interpreted. Actually we distinguish two types of referencing within the classification:

4.1 Key

The transferred information is exchanged based on a defined key. This is indicated by the given 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.

```

<TAGS>
  <TAG>
    <KEY>shippingcondition</KEY>
    <NAME>Versandzustand</NAME>
    <VALUES>
      <VALUE>
        <KEY>45</KEY>
        <NAME>montiert</NAME>
      </VALUE>
      <VALUE>
        <KEY>46</KEY>
        <NAME>teilmontiert</NAME>
      </VALUE>
      <VALUE>
        <KEY>47</KEY>
        <NAME>zerlegt</NAME>
      </VALUE>
    </VALUES>
  </TAG>
</TAGS>
...
<ATTRIBUTE_SETS>
  <ATTRIBUTE_SET global="true">
    <NAME>globales Mapping</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="1" maxOccurs="1">
      <TAG identifier="key" key="shippingcondition"/>
    </ATTRIBUTE_SET_ITEM>
  </ATTRIBUTE_SET>
</ATTRIBUTE_SETS>

```

4.2 Name

The transferred information is provided based on the name (designation) which the sender defined. This is indicated by the given 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 the receiver receives data which can be assigned by means of the transferred area (component, attribute, and brand) but any information can/should not be interpreted accurately.

```

<TAGS>
  <TAG>
    <KEY>shippingcondition</KEY>
    <NAME>Versandzustand</NAME>
    <VALUES>
      <VALUE>
        <KEY>45</KEY>
        <NAME>montiert</NAME>
      </VALUE>
      <VALUE>
        <KEY>46</KEY>
        <NAME>teilmontiert</NAME>
      </VALUE>
      <VALUE>
        <KEY>47</KEY>
        <NAME>zerlegt</NAME>
      </VALUE>
    </VALUES>
  </TAG>
</TAGS>

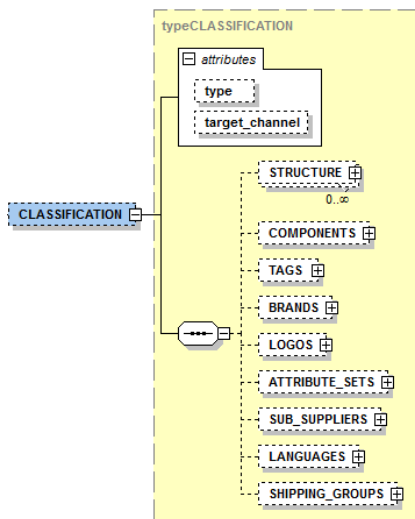
```

```
        </VALUE>
      </VALUES>
    </TAG>
  </TAGS>
  ...
  <ATTRIBUTE_SETS>
    <ATTRIBUTE_SET global="true">
      <NAME>globales Mapping</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="1" maxOccurs="1">
        <TAG identifier="name" key="Versandzustand"/>
      </ATTRIBUTE_SET_ITEM>
    </ATTRIBUTE_SET>
  </ATTRIBUTE_SETS>
```

5 Structure of the TB.CAT Classification

TBCATALOG / CLASSIFICATION

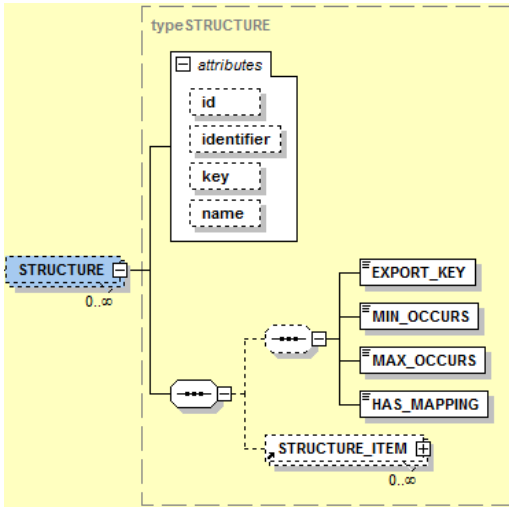
The classification part optionally contains lists with categories, components, attributes, brands and logos. This part of the TB.CAT serves the definition of the desired (mandatory) values by you.



Node	Must	Data type	Description
CLASSIFICATION[type]	N	ENUM	Type of the Classification feed Possible values are: merge = existing and provided values are summarized; not provided values are ignored. This is the default setting
CLASSIFICATION[target_channel]	N	CHAR(12)	Channel sign of the target channel
STRUCTURE	N	XML	Categories
COMPONENTS	N	XML	Properties with continuous text values
TAGS	N	XML	"Machine" evaluable properties with values from a value list such as filter attributes in online shops
BRANDS	N	XML	Brand list
LOGOS	N	XML	Logo/Icon list
ATTRIBUTE_SETS	N	XML	Definition of mandatory values
SUB_SUPPLIERS	N	XML	Only relevant in TB.CAT Export - indicates within a TB.Market export a reference to the connected TB.One clients/suppliers
LANGUAGES	N	XML	Definition of the languages you offer in your store, and which must be supplied by merchants.
SHIPPING_GROUPS	N	XML	If you define shipping groups here you set shipping types binding at your channel for all connected merchants (TB.Ones).

TBCATALOG / CLASSIFICATION / STRUCTURE

This part is used to define a category tree (STRUCTURE), where the products must be assigned to.



Node	Must	Data type	Description
STRUCTURE[id]	N	INT	Variant of the category tree (currently not used)
STRUCTURE[identifier]	N	ENUM	Referencing base for the structure (key or name)
STRUCTURE[key]	N		Value for referencing base
STRUCTURE[name]	N		Additional name of the structure, if identifier = key
EXPORT_KEY	Y	STRING	Export key of the structure
MIN_OCCURS	Y	INT(0/1)	Value 1 sets the specification of a channel category at mandatory.
MAX_OCCURS	Y	STRING	Controls the number of categories that can be assigned to a product. Possible values are: Value 1 to n: one to n categories can be assigned (product appears in the shop in 1 to n menu items) Unbounded
HAS_MAPPING	Y	STRING	Structural information is generally recorded in TB.One/TB.Market in an ordered sequence. This field determines whether the value mapping must be maintained only for the first item or for all items in that list. Possible values are: all: A product must meet all the mandatory fields which arise from the first and the second (or any further) structural assignment. first: Only the mandatory fields of the first category assignment is checked. Mandatory fields of additional assignments are not checked. no: Required fields are not checked.
STRUCTURE_ITEM	N	XML	Category node

Sample XML (key-based)

```
<STRUCTURE identifier="key" key="nav_pri" />
```

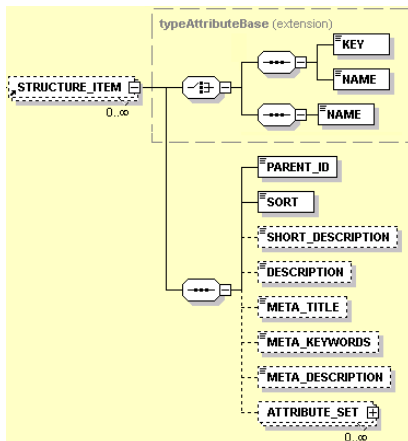
or

```
<STRUCTURE identifier="key" key="nav_pri" name="Kategorienbaum" />
```

Sample XML (name-based)

```
<STRUCTURE identifier="name" key="Kategorienbaum" />
```

TBCATALOG / CLASSIFICATION / STRUCTURE / STRUCTURE_ITEM



Node	Must	Data type	Description
STRUCTURE_ITEM	N	XML	Category node
PARENT_ID	Y	INT	Reference to the ID of the parent node (Main Category: value = 0). Always pass "0" for a flat hierarchy.
SORT	Y	INT	Sort value of the category within the associated parent node. Without sorting values, TB.One sorts the categories alphabetically.
SHORT_DESCRIPTION	N	TEXT	Brief description of the category node
DESCRIPTION	N	TEXT	Description of the category node
META_TITLE	N	CHAR(255)	Meta title of the category node
META_KEYWORDS	N	CHAR(255)	Meta Keywords of the category node
META_DESCRIPTION	N	CHAR(255)	Meta-Description of the category node
ATTRIBUTE_SET	N	XML	Assignment of attribute sets to this category node. The definition of the attribute sets take place in the node "Attribute Sets".
Import via Key			
KEY	Y	CHAR(50)	Import key of the category node
NAME	Y	CHAR(200)	Name of the category node
Import via Name			
NAME	Y	CHAR(200)	Name of the category node

Sample XML (key-based, flat)

```

<STRUCTURE>
  <STRUCTURE_ITEM>
    <KEY>bush</KEY>
    <NAME>Büsche</NAME>
    <PARENT_ID>0</PARENT_ID>
    <SORT>10</SORT>
  </STRUCTURE_ITEM>
  <STRUCTURE_ITEM>
    <KEY>4712</KEY>
    <NAME>Büsche | Beeren</NAME>
    <PARENT_ID>0</PARENT_ID>
    <SORT>20</SORT>
  </STRUCTURE_ITEM>
</STRUCTURE>

```

The result of the category list under "Channels > Channel name > Categories":

Sample XML (key-based, tree)

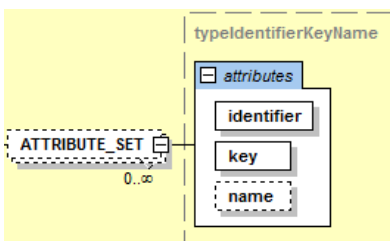
```
<STRUCTURE>
  <STRUCTURE_ITEM>
    <ID>4712</ID>
    <KEY>te12-05</KEY>
    <NAME>Pflanzen</NAME>
    <PARENT_ID>0</PARENT_ID>
    <SORT>10</SORT>
  </STRUCTURE_ITEM>
  <STRUCTURE_ITEM>
    <ID>4713</ID>
    <KEY>te12-06</KEY>
    <NAME>Obstbäume</NAME>
    <PARENT_ID>4712</PARENT_ID>
    <SORT>20</SORT>
  </STRUCTURE_ITEM>
</STRUCTURE>
```

The result of the category list under "Channels > Channel name > Categories":

Here, a product can only be in one category and products can only be assigned to the lowest hierarchy node.

TBCATALOG / ... / STRUCTURE_ITEM / ATTRIBUTE_SET

By means of the node TBCATALOG / CLASSIFICATION / ATTRIBUTE_SETS you can define mandatory and desired fields for your categories. Here they are assigned to the current structure node.



Node	Must	Data type	Description
ATTRIBUTE_SET	N		Attribute definition for a category node
ATTRIBUTE_SET[identifier]	Y	ENUM	Identifier for referencing
ATTRIBUTE_SET[key]	Y	CHAR(50)	Value stored in the <i>identifier</i> assignment statement
ATTRIBUTE_SET[name]	N	CHAR(100)	Name stored in the <i>identifier</i> assignment statement

Sample XML (key-based)

```
<ATTRIBUTE_SET identifier="key" key="zeit"/>
```

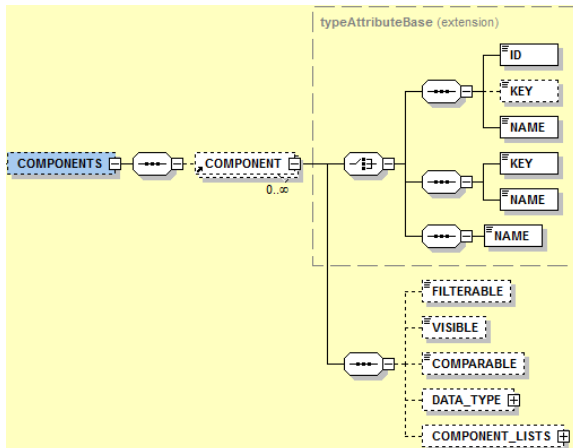
or

```
<ATTRIBUTE_SET identifier="key" key="zeit" name="Zeiten"/>
```

Sample XML (name-based)

```
<ATTRIBUTE_SET identifier="name" key="Zeiten"/>
```

TBCATALOG / CLASSIFICATION / COMPONENTS



Node	Must	Data type	Description
COMPONENTS	N	XML	Component node
COMPONENT	N	XML	Individual component
COMPONENT[filterable]	N	BOOLEAN	For the use of the component in a generic channel you can specify whether the component is to serve as a filter or comparison criterion or whether it is visible. This is optional. Is displayed in TB.One/TB.Market, but only passed on, so that for example the web shop can further process the information
COMPONENT[visible]	N	BOOLEAN	
COMPONENT[comparable]	N	BOOLEAN	
DATA_TYPE	N	XML	Specifies the data type of the component
COMPONENT_LISTS	N	XML	List of values for the component
Import via Key			
KEY	Y	CHAR(50)	Import key of the node
NAME	Y	CHAR(200)	Name of the node
Import via Name			
NAME	Y	CHAR(200)	Name of the node

Sample XML (name-based)

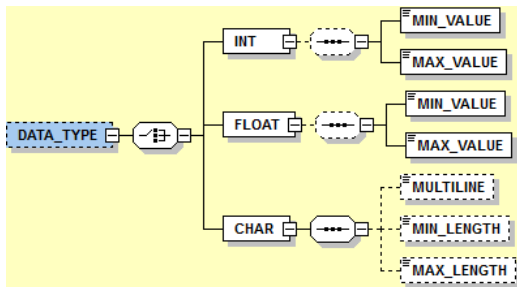
```
<COMPONENTS>
  <COMPONENT >
    <NAME>Farbe</NAME>
  </COMPONENT>"/>
</COMPONENTS>
```

Sample XML (key-based)

```
<COMPONENTS>
  <COMPONENT >
    <key>color</key>
    <NAME>Farbe</NAME>
  </COMPONENT>"/>
</COMPONENTS>
```

TBCATALOG / CLASSIFICATION / COMPONENTS / DATA_TYPE

For the component you can specify the data type. If you provide the node, choose one of the data types (see examples below).



Node	Must	Data type	Description
DATA_TYPE	N	XML	Definition of the data type for component
INT	Y	XML	Definition of component values as integer
MIN_VALUE	N	INT	Permitted value range (minimum and maximum values, and the values given are each still allowed). If MIN_VALUE is specified, also MAX_VALUE must be provided.
MAX_VALUE			
FLOAT	Y	XML	Definition of component values as floating-point number
MIN_VALUE	N	FLOAT	Permitted value range (minimum and maximum values, and the values given are each still allowed). If MIN_VALUE is specified, also MAX_VALUE must be provided.
MAX_VALUE			
CHAR	Y	XML	Definition of component values as text
MULTILINE	N	BOOLEAN	Specifies whether the text is single line or multiple lines. Without delivery of the node, both are possible. If, then: TRUE : single and multi-line text FALSE : only single-line text
MIN_LENGTH	N	INT	Minimum length of the given text
MAX_LENGTH	N	INT	Maximum length of the given text

Sample XML (INT)

```

<COMPONENT>
  <KEY>garant</KEY>
  <NAME>Garantiezeit</NAME>
  <DATA_TYPE>
    <INT>
      <MIN_VALUE>2</MIN_VALUE>
      <MAX_VALUE>10</MAX_VALUE>
    </INT>
  </DATA_TYPE>
</COMPONENT>
  
```

Sample XML (FLOAT)

```

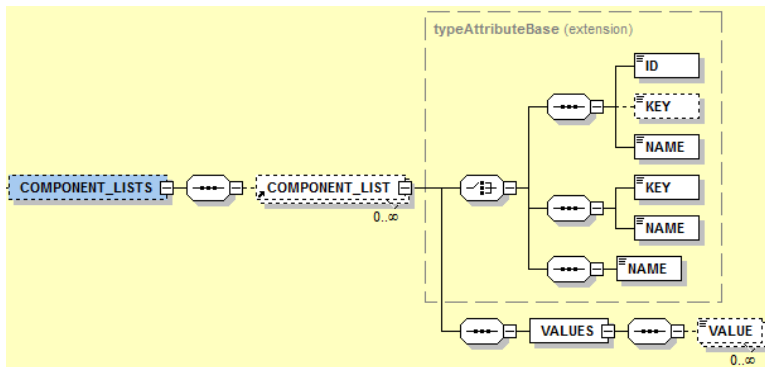
<COMPONENT>
  <KEY>pond</KEY>
  <NAME>Spaltkraft</NAME>
  <DATA_TYPE>
    <FLOAT>
      <MIN_VALUE>2.08</MIN_VALUE>
      <MAX_VALUE>25.99</MAX_VALUE>
    </FLOAT>
  </DATA_TYPE>
</COMPONENT>
  
```

Sample XML (CHAR)

```

<COMPONENT>
  <KEY>color</KEY>
  <NAME>Farbe</NAME>
  <DATA_TYPE>
    <CHAR>
      <MULTILINE>>false</MULTILINE>
      <MIN_LENGTH>1</MIN_LENGTH>
      <MAX_LENGTH>50</MAX_LENGTH>
    </CHAR>
  </DATA_TYPE>
</COMPONENT>
  
```

TBCATALOG / CLASSIFICATION / COMPONENTS / COMPONENT_LISTS



Node	Must	Data type	Description
COMPONENT_LISTS	N	XML	Nodes to define lists of values for component
COMPONENT_LIST	N	XML	Definition of one value list for the component
VALUES	Y	XML	List of possible values of the component
VALUE	N	XML	Single value
Import via Key			
KEY	Y	CHAR(50)	Import key of the node
NAME	Y	CHAR(200)	Name of the node
Import via Name			
NAME	Y	CHAR(200)	Name of the node

Sample XML (key-based)

```

<COMPONENT_LISTS>
  <COMPONENT_LIST>
    <KEY>supjamcol</key>
    <NAME>Farben von Superjackenmeister</NAME>
    <VALUES>
      <VALUE>rot</VALUE>
      <VALUE>grün</VALUE>
      <VALUE>blau</VALUE>
    </VALUES>
  </COMPONENT_LIST>
</COMPONENT_LISTS>
  
```

Sample XML (name-based)

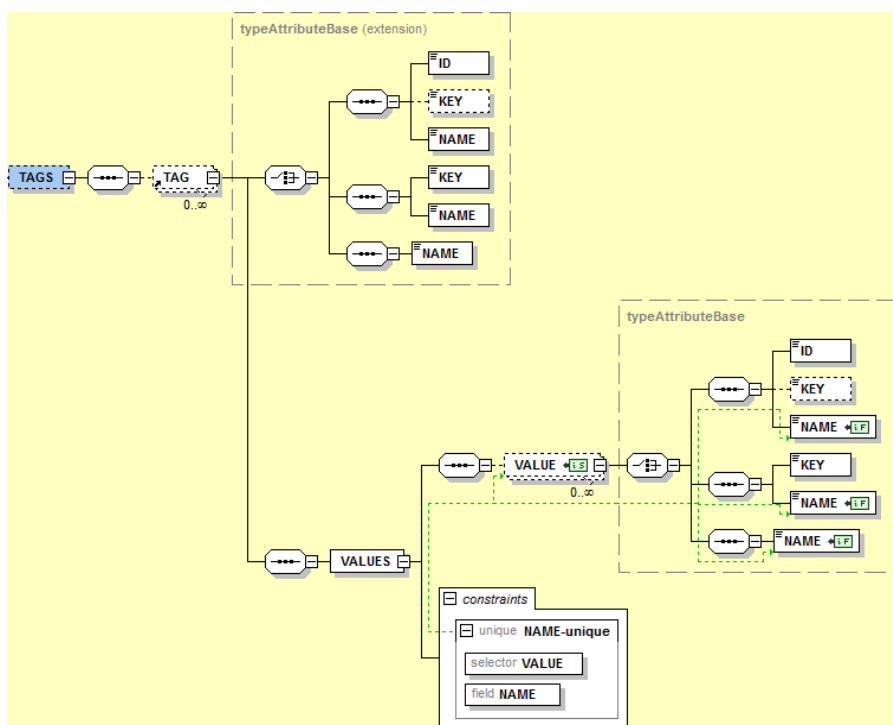
```

<COMPONENT_LISTS>
  <COMPONENT_LIST>
    <NAME>Farben von Superjackenmeister</NAME>
    <VALUES>
      <VALUE>rot</VALUE>
      <VALUE>grün</VALUE>
    </VALUES>
  </COMPONENT_LIST>
</COMPONENT_LISTS>
  
```

```

</VALUES>
</COMPONENT_LIST>
<COMPONENT_LIST>
  <NAME>Farben von Hosenfreund</NAME>
  <VALUES>
    <VALUE>lila</VALUE>
    <VALUE>blau</VALUE>
  </VALUES>
</COMPONENT_LIST>
</COMPONENT_LISTS>
    
```

TBCATALOG / CLASSIFICATION / TAGS



Node	Must	Data type	Description
TAGS	N	XML	Definition of attributes
TAG	N	XML	Definition of one attribute
VALUES	Y	XML	Definition of a values list for the attribute
VALUE	N	XML	Single value
Import via Key			
KEY	Y	CHAR(50)	Import key of the node
NAME	Y	CHAR(200)	Name of the node (unique)
Import via Name			
NAME	Y	CHAR(200)	Name of the node (unique)

Sample XML (key-based)

```

<TAGS>
  <TAG>
    <KEY>gender</KEY>
    <NAME>Zielgruppe</NAME>
    <VALUES>
      <VALUE>
        <KEY>male</KEY>
        <NAME>Mann</NAME>
      </VALUE>
    </VALUES>
  </TAG>
</TAGS>
    
```

```

</VALUE>
<VALUE>
  <KEY>female</KEY>
  <NAME>Frau</NAME>
</VALUE>
</VALUES>
</TAG>
</TAGS>

```

Sample XML (name-based)

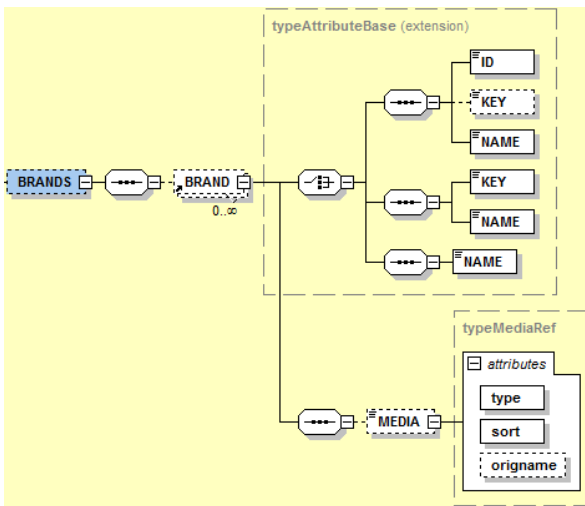
```

<TAGS>
  <TAG>
    <NAME>Geschlecht</NAME>
    <VALUES>
      <VALUE>
        <NAME>Mann</NAME>
      </VALUE>
      <VALUE>
        <NAME>Frau</NAME>
      </VALUE>
      <VALUE>
        <NAME>Unisex</NAME>
      </VALUE>
    </VALUES>
  </TAG>
</TAGS>

```

TBCATALOG / CLASSIFICATION / BRANDS

If you define a brand list within the Classification, a TB.One user can only use brands of your brand list for value mapping. The statement of brands is optional.



Node	Must	Data type	Description
BRANDS	N	XML	Brand node
BRAND	N	XML	Definition of a brand
MEDIA	N	STRING	(Not relevant for classification Import)
Import via Key			
KEY	Y	CHAR(50)	Import key of the node
NAME	Y	CHAR(200)	Name of the node
Import via Name			
NAME	Y	CHAR(200)	Name of the node

Sample XML (key-based)

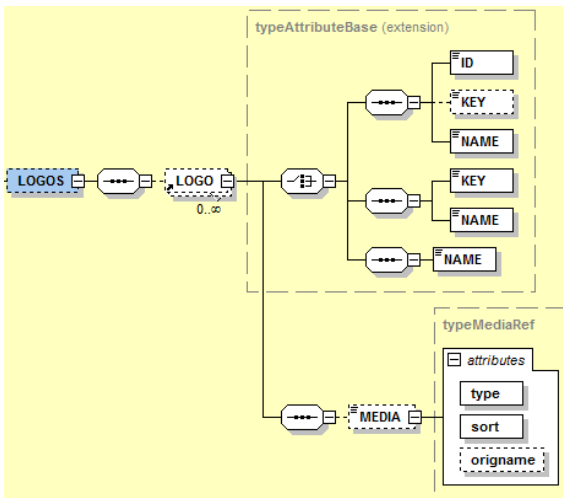
```

<BRANDS>
  <BRAND>
    <KEY>supjam</KEY>
    <NAME>Superjackenmeister</NAME>
    <MEDIA type="image" sort="10" origname=" http://foo.server/blau.jpg" />
  </BRAND>
</BRANDS>

```

TBCATALOG / CLASSIFICATION / LOGOS

Logos are usually used to represent standardized product features, such as care symbols (30° wash), certifications (FSC) or features (Bluetooth). The provision in the classification part is not necessary.



Node	Must	Data type	Description
LOGOS	N	XML	Logo node
LOGO	N	XML	Definition of a logo
MEDIA	N	STRING	(Not relevant for classification Import)
Import via Key			
KEY	Y	CHAR(50)	Import key of the node
NAME	Y	CHAR(200)	Name of the node
Import via Name			
NAME	Y	CHAR(200)	Name of the node

Sample XML (key-based)

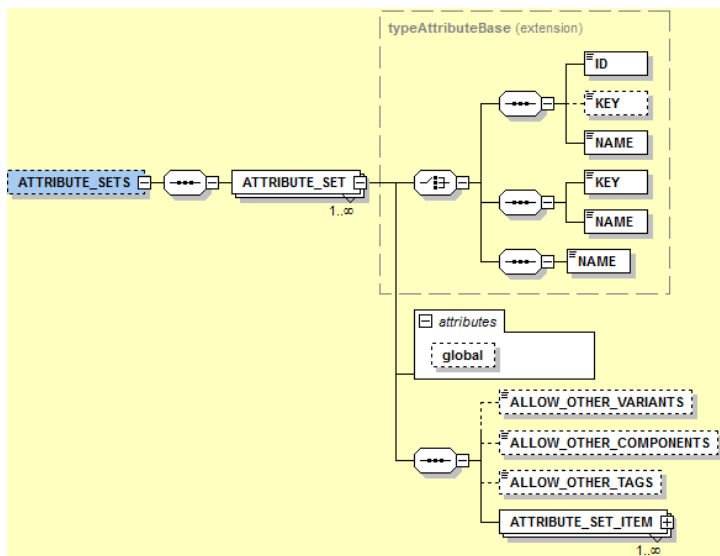
```

<LOGOS>
  <LOGO>
    <KEY>30</KEY>
    <NAME>30°C Wäsche</NAME>
    <MEDIA type="logo" sort="10" origname=" http://foo.server/30grad.jpg" />
  </LOGO>
</LOGOS>

```

TBCATALOG / CLASSIFICATION / ATTRIBUTE_SETS

By means of attribute sets, you can define mandatory and desire fields within your categories.



Node	Must	Data type	Description
ATTRIBUTE_SETS	N	XML	Root node for attribute sets
ATTRIBUTE_SET	N	XML	Attribute set definition
ATTRIBUTE_SET[global]	N	BOOLEAN	If you set the attribute set to "global", it is applied to all defined category nodes.
ALLOW_OTHER_VARIANTS	N	BOOLEAN	If you wish to suppress any other values (variants, components, attributes) in addition to the values defined by you, you set this value to "false".
ALLOW_OTHER_COMPONENTS	N	BOOLEAN	
ALLOW_OTHER_TAGS	N	BOOLEAN	
ATTRIBUTE_SET_ITEM	Y	XML	Element of the attribute set
Import via Key			
KEY	Y	CHAR(50)	Import key of the node
NAME	Y	CHAR(200)	Name of the node
Import via Name			
NAME	Y	CHAR(200)	Name of the node

Sample XML

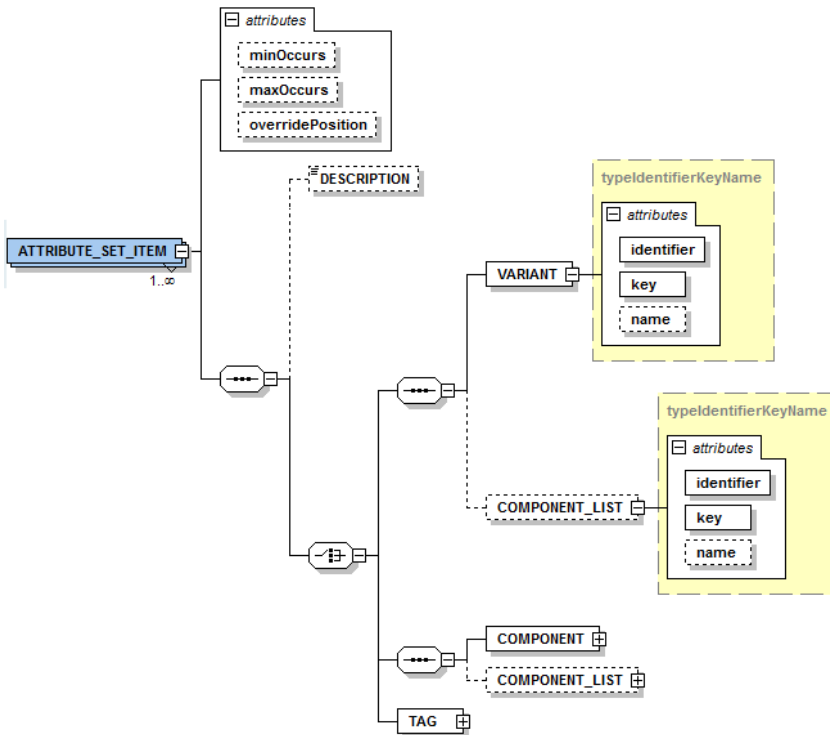
```

<ATTRIBUTE_SETS>
  <ATTRIBUTE_SET global="true" />
  <ATTRIBUTE_SET global="false" />
  <ATTRIBUTE_SET />
</ATTRIBUTE_SETS>
    
```

<!-- corresponds to "global=false" -->

It is useful to set the value to "true", if [STRUCTURE](#) - HAS_MAPPING has the value "first".

TBCATALOG / ... / ATTRIBUTE_SETS / ATTRIBUTE_SET_ITEM



Node	Must	Data type	Description															
ATTRIBUTE_SET_ITEM	Y	XML	Element of an attribute set															
ATTRIBUTE_SET_ITEM [min/maxOccurs]	N	INT	Defines the frequency of occurrence of the element defined below. Standard for both elements "1", i.e. the element is a mandatory input. <table border="1"> <thead> <tr> <th>Min</th> <th>Max</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>1</td> <td>Need not be set but must not be used more than once</td> </tr> <tr> <td>1</td> <td>1</td> <td>Mandatory value</td> </tr> <tr> <td>1</td> <td>unbounded</td> <td>Mandatory value that can be used multiple times</td> </tr> <tr> <td>0</td> <td>unbounded</td> <td>Optional value that may be used multiple times</td> </tr> </tbody> </table>	Min	Max	Description	0	1	Need not be set but must not be used more than once	1	1	Mandatory value	1	unbounded	Mandatory value that can be used multiple times	0	unbounded	Optional value that may be used multiple times
Min	Max	Description																
0	1	Need not be set but must not be used more than once																
1	1	Mandatory value																
1	unbounded	Mandatory value that can be used multiple times																
0	unbounded	Optional value that may be used multiple times																
ATTRIBUTE_SET_ITEM [overridePosition]	N	STRING	Determines at which element of the product a property (attributes/components) this attribute set is exported (product/article). Possible values are: <table border="1"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>no</td> <td>Exported as stored from the merchant</td> </tr> <tr> <td>article</td> <td>Export at article</td> </tr> <tr> <td>product</td> <td>Export at product</td> </tr> </tbody> </table>	Value	Description	no	Exported as stored from the merchant	article	Export at article	product	Export at product							
Value	Description																	
no	Exported as stored from the merchant																	
article	Export at article																	
product	Export at product																	
DESCRIPTION	N	TEXT	Optional description of how this attribute shall be used. Shown in TB.One in the value mapping as hint.															
VARIANT/COMPONENT/TAG	Y	XML	Assignment of a variant dimension, component or attribute to the attribute set															
COMPONENT_LIST	N	XML	For variants and components you may reference to a value list defined under COMPONENT_LISTS which is valid within this															

Node	Must	Data type	Description
			attribute set.
VARIANT[identifier] COMPONENT[identifier] TAG[identifier]	Y	ENUM	Identifier for referencing (key, name)
VARIANT[key] COMPONENT[key] TAG[key]	Y	CHAR(50)	Value stored in the <i>identifier</i> assignment statement
VARIANT[name] COMPONENT[name] TAG[name]	N	CHAR(100)	Name stored in the <i>identifier</i> assignment statement
Import via Key			
KEY	Y	CHAR(50)	Import key of the node
NAME	Y	CHAR(200)	Name of the node
Import via Name			
NAME	Y	CHAR(200)	Name of the node

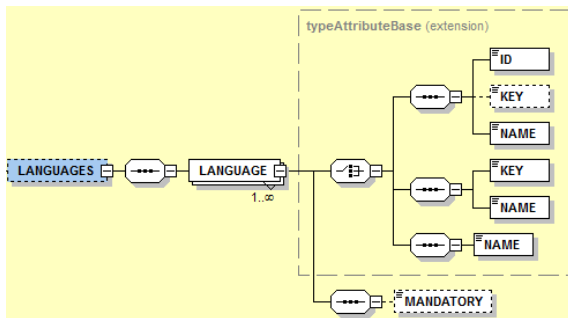
Sample XML (key-based)

```
<ATTRIBUTE_SETS>
  <ATTRIBUTE_SET>
    <NAME>Blumenmischung</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="1" maxOccurs="unbounded">
      <COMPONENT identifier="key" key="farbe"/>
    </ATTRIBUTE_SET_ITEM>
  </ATTRIBUTE_SET>
</ATTRIBUTE_SETS>
```

Sample XML (name-based)

```
<ATTRIBUTE_SETS>
  <ATTRIBUTE_SET global="true">
    <NAME>Blumenmischung</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="1" maxOccurs="unbounded">
      <DESCRIPTION>Die ERP Farbe kann von der tatsächlichen Farbe
abweichen.</DESCRIPTION>
      <COMPONENT identifier="name" key="ERP Farbe"/>
      <COMPONENT_LIST identifier="name" key="ERP Farben">
    </ATTRIBUTE_SET_ITEM>
  </ATTRIBUTE_SET>
</ATTRIBUTE_SETS>
```

TBCATALOG / CLASSIFICATION / LANGUAGES



Node	Must	Data type	Description
LANGUAGES	N	XML	Language definition
LANGUAGE	Y	XML	Definition of one language
MANDATORY	N	BOOLEAN	Definition as mandatory information (true/false, default is false)
Import via Key			
KEY	Y	CHAR(50)	Import key of the node
NAME	Y	CHAR(200)	Name of the node
Import via Name			
NAME	Y	CHAR(200)	Name of the node

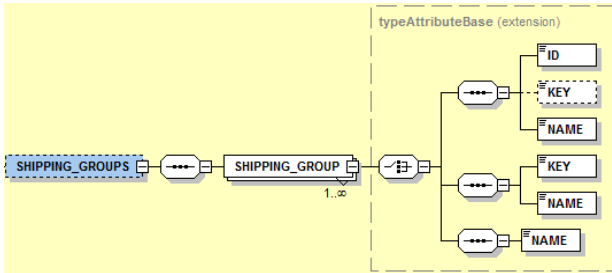
Sample XML (key-based)

```
<LANGUAGES>
  <LANGUAGE>
    <KEY>de-DE</KEY>
    <NAME>Deutsch</NAME>
    <MANDATORY>>true</MANDATORY>
  </LANGUAGE>
  <LANGUAGE>
    <KEY>en-US</KEY>
    <NAME>Englisch</NAME>
  </LANGUAGE>
</LANGUAGES>
```

Sample XML (name-based)

```
<LANGUAGES>
  <LANGUAGE>
    <NAME>Deutsch</NAME>
    <MANDATORY>>true</MANDATORY>
  </LANGUAGE>
  <LANGUAGE>
    <NAME>Englisch</NAME>
  </LANGUAGE>
</LANGUAGES>
```

TBCATALOG / CLASSIFICATION / SHIPPING_GROUPS



Node	Must	Data type	Description
SHIPPING_GROUPS	N	XML	Definition of shipping types
SHIPPING_GROUP	Y	XML	Definition of one shipping type
Import via Key			
KEY	Y	CHAR(50)	Import key of the node
NAME	Y	CHAR(200)	Name of the node
Import via Name			
NAME	Y	CHAR(200)	Name of the node

Sample XML (key-based)

```
<SHIPPING_GROUPS>
  <SHIPPING_GROUP>
    <KEY>other_parcel</KEY>
    <NAME>paketversandfähig</NAME>
  </SHIPPING_GROUP>
  <SHIPPING_GROUP>
    <KEY>other_spedition</KEY>
    <NAME>Spedition 1-Mann-Handling</NAME>
  </SHIPPING_GROUP>
</SHIPPING_GROUPS>
```

Sample XML (name-based)

```
<SHIPPING_GROUPS>
  <SHIPPING_GROUP>
    <NAME>paketversandfähig</NAME>
  </SHIPPING_GROUP>
  <SHIPPING_GROUP>
    <NAME>Spedition 1-Mann-Handling</NAME>
  </SHIPPING_GROUP>
</SHIPPING_GROUPS>
```

6 Example Classification

6.1 Flat Structure

```

<?xml version="1.0" encoding="UTF-8"?>
<TBCATALOG version="1.3" creation="1378277984"
xsi:noNamespaceSchemaLocation="http://api.trade-server.net/schema/all_in_one/tb-
cat_1_3_all_in_one.xsd">
  <CLASSIFICATION type="merge" target_channel="cudoku">
    <STRUCTURE identifier="key" key="nav_primary" name="Kategoriebaum">
      <EXPORT_KEY>nav_primary</EXPORT_KEY>
      <!-- optional -->
      <MIN_OCCURS>1</MIN_OCCURS>
      <MAX_OCCURS>unbounded</MAX_OCCURS>
      <HAS_MAPPING>all</HAS_MAPPING>
      <STRUCTURE_ITEM>
        <KEY>bush</KEY>
        <NAME>Büsche</NAME>
        <PARENT_ID>0</PARENT_ID>
        <SORT>0</SORT>
      </STRUCTURE_ITEM>
      <STRUCTURE_ITEM>
        <KEY>berrybush</KEY>
        <NAME>Büsche | Beeren</NAME>
        <PARENT_ID>0</PARENT_ID>
        <SORT>0</SORT>
        <ATTRIBUTE_SET identifier="key" key="color"/>
        <ATTRIBUTE_SET identifier="key" key="height"/>
      </STRUCTURE_ITEM>
    </STRUCTURE>
    <COMPONENTS>
      <COMPONENT>
        <KEY>color</KEY>
        <NAME>Farbe</NAME>
        <DATA_TYPE>
          <CHAR>
            <MULTILINE>>false</MULTILINE>
            <MIN_LENGTH>1</MIN_LENGTH>
            <MAX_LENGTH>50</MAX_LENGTH>
          </CHAR>
        </DATA_TYPE>
      </COMPONENT>
      <COMPONENT>
        <KEY>height</KEY>
        <NAME>Höhe</NAME>
        <DATA_TYPE>
          <FLOAT>
            <MIN_VALUE>0.5</MIN_VALUE>
            <MAX_VALUE>300</MAX_VALUE>
          </FLOAT>
        </DATA_TYPE>
      </COMPONENT>
    </COMPONENTS>
    <TAGS>
      <TAG>
        <KEY>location</KEY>
        <NAME>Standort</NAME>
        <VALUES>
          <VALUE>
            <KEY>sun</KEY>
            <NAME>sonnig</NAME>
          </VALUE>
        </VALUES>
      </TAG>
    </TAGS>
  </CLASSIFICATION>
</TBCATALOG>

```

```

    <VALUE>
      <KEY>shade</KEY>
      <NAME>halbschattig</NAME>
    </VALUE>
    <VALUE>
      <KEY>dark</KEY>
      <NAME>schattig</NAME>
    </VALUE>
  </VALUES>
</TAG>
</TAGS>
<BRANDS>
  <BRAND>
    <KEY>plajam</KEY>
    <NAME>Pflanzenmeister</NAME>
  </BRAND>
  <BRAND>
    <KEY>butri</KEY>
    <NAME>Buschtrimmer</NAME>
  </BRAND>
</BRANDS>
<ATTRIBUTE_SETS>
  <ATTRIBUTE_SET global="true">
    <NAME>globales Mapping</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="1" maxOccurs="1">
      <TAG identifier="key" key="color"/>
    </ATTRIBUTE_SET_ITEM>
    <ATTRIBUTE_SET_ITEM minOccurs="1" maxOccurs="1">
      <COMPONENT identifier="key" key="hight"/>
    </ATTRIBUTE_SET_ITEM>
  </ATTRIBUTE_SET>
</ATTRIBUTE_SETS>
<LANGUAGES>
  <LANGUAGE>
    <KEY>de-DE</KEY>
    <NAME>Deutsch</NAME>
  </LANGUAGE>
  <LANGUAGE>
    <KEY>en-US</KEY>
    <NAME>Englisch</NAME>
  </LANGUAGE>
</LANGUAGES>
<SHIPPING_GROUPS>
  <SHIPPING_GROUP>
    <KEY>te_d1</KEY>
    <NAME>DHL</NAME>
  </SHIPPING_GROUP>
  <SHIPPING_GROUP>
    <KEY>te_d2</KEY>
    <NAME>Spedition</NAME>
  </SHIPPING_GROUP>
</SHIPPING_GROUPS>
</CLASSIFICATION>
</TBCATALOG>

```


6.2 Tree Structure

```

<?xml version="1.0" encoding="UTF-8"?>
<TBCATALOG version="1.3" creation="1378277984"
xsi:noNamespaceSchemaLocation="http://api.trade-server.net/schema/all_in_one/tb-
cat_1_3_all_in_one.xsd">
  <CLASSIFICATION type="merge" target_channel="cudoku">
    <STRUCTURE identifier="key" key="nav_primary" name="Kategoriebaum">
      <EXPORT_KEY>nav_primary</EXPORT_KEY>
      <!-- optional -->
      <MIN_OCCURS>1</MIN_OCCURS>
      <MAX_OCCURS>unbounded</MAX_OCCURS>
      <HAS_MAPPING>all</HAS_MAPPING>
      <STRUCTURE_ITEM>
        <ID>200</ID>
        <KEY>bush</KEY>
        <NAME>Büsche</NAME>
        <PARENT_ID>0</PARENT_ID>
        <SORT>0</SORT>
      </STRUCTURE_ITEM>
      <STRUCTURE_ITEM>
        <ID>210</ID>
        <KEY>berrybush</KEY>
        <NAME>Beeren</NAME>
        <PARENT_ID>200</PARENT_ID>
        <SORT>0</SORT>
        <ATTRIBUTE_SET identifier="key" key="color"/>
        <ATTRIBUTE_SET identifier="key" key="height"/>
      </STRUCTURE_ITEM>
    </STRUCTURE>
    <COMPONENTS>
      <COMPONENT>
        <KEY>color</KEY>
        <NAME>Farbe</NAME>
        <DATA_TYPE>
          <CHAR>
            <MULTILINE>>false</MULTILINE>
            <MIN_LENGTH>1</MIN_LENGTH>
            <MAX_LENGTH>50</MAX_LENGTH>
          </CHAR>
        </DATA_TYPE>
      </COMPONENT>
      <COMPONENT>
        <KEY>height</KEY>
        <NAME>Höhe</NAME>
        <DATA_TYPE>
          <FLOAT>
            <MIN_VALUE>0.5</MIN_VALUE>
            <MAX_VALUE>300</MAX_VALUE>
          </FLOAT>
        </DATA_TYPE>
      </COMPONENT>
    </COMPONENTS>
    <TAGS>
      <TAG>
        <KEY>location</KEY>
        <NAME>Standort</NAME>
        <VALUES>
          <VALUE>
            <KEY>sun</KEY>
            <NAME>sonnig</NAME>
          </VALUE>
          <VALUE>
            <KEY>shade</KEY>
            <NAME>halbschattig</NAME>
          </VALUE>
        </VALUES>
      </TAG>
    </TAGS>
  </CLASSIFICATION>
</TBCATALOG>

```

```

    <VALUE>
      <KEY>dark</KEY>
      <NAME>schattig</NAME>
    </VALUE>
  </VALUES>
</TAG>
</TAGS>
<BRANDS>
  <BRAND>
    <KEY>plajam</KEY>
    <NAME>Pflanzenmeister</NAME>
  </BRAND>
  <BRAND>
    <KEY>butri</KEY>
    <NAME>Buschtrimmer</NAME>
  </BRAND>
</BRANDS>
<ATTRIBUTE_SETS>
  <ATTRIBUTE_SET global="true">
    <NAME>globales Mapping</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="1" maxOccurs="1">
      <TAG identifier="key" key="color"/>
    </ATTRIBUTE_SET_ITEM>
    <ATTRIBUTE_SET_ITEM minOccurs="1" maxOccurs="1">
      <COMPONENT identifier="key" key="height"/>
    </ATTRIBUTE_SET_ITEM>
  </ATTRIBUTE_SET>
</ATTRIBUTE_SETS>
<LANGUAGES>
  <LANGUAGE>
    <KEY>de-DE</KEY>
    <NAME>Deutsch</NAME>
  </LANGUAGE>
  <LANGUAGE>
    <KEY>en-US</KEY>
    <NAME>Englisch</NAME>
  </LANGUAGE>
</LANGUAGES>
<SHIPPING_GROUPS>
  <SHIPPING_GROUP>
    <KEY>te_1</KEY>
    <NAME>DHL</NAME>
  </SHIPPING_GROUP>
  <SHIPPING_GROUP>
    <KEY>te_2</KEY>
    <NAME>Spedition</NAME>
  </SHIPPING_GROUP>
</SHIPPING_GROUPS>
</CLASSIFICATION>
</TBCATALOG>

```