Date: 03.05.2023

Changelog MODIFY_CONTENT_IDML Version 4.0.0

This changelog describes all new features, enhancements and corrections that are provided with the current version **MODIFY_CONTENT_IDML 4.0.0** (compared to version MODIFY_CONTENT_IDMP 3.1.0) both in the XML Schema and in the documentation.

The version MODIFY_CONTENT_IDML 4.0.0 is published on 2022-xx-xx and becomes valid from 2022-xx-xx.

Content:

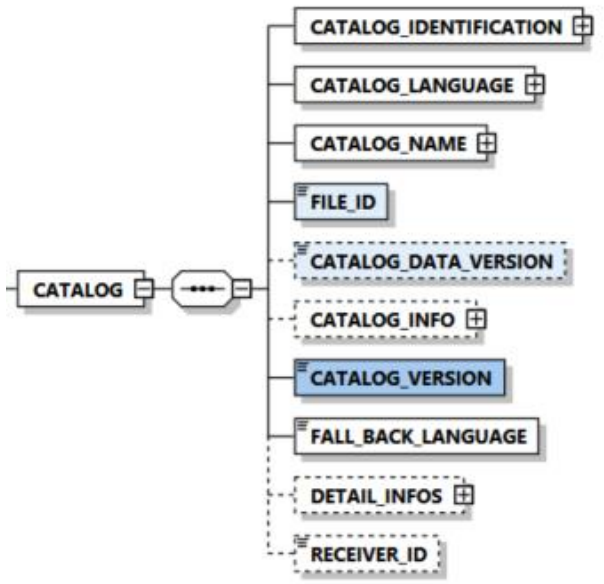
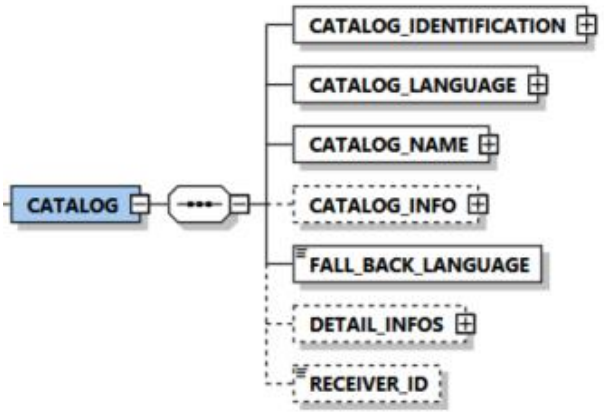
A = Added = Adding new elements or attributes
C = Changed = Changes to existing elements, attributes or descriptive texts.
R = Removed = Deletion of elements or attributes
F = Fixed = Error corrections to existing elements, attributes or descriptive texts.

Released

Display of all changes recorded for version 4.0.0

1. Delete unnecessary elements

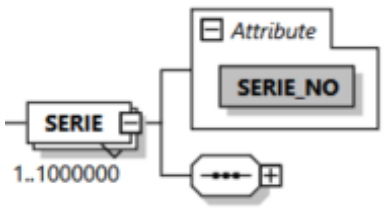
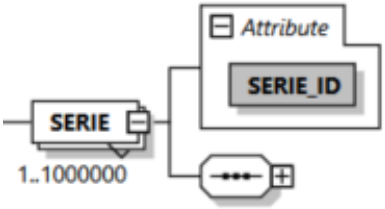
1.1.	R	The elements FILE_ID, CATALOG_DATA_VERSION and CATALOG_VERSION are deleted	Decision : 2022-03-09

3.1.0	4.0.0
	

The elements FILE_ID, CATALOG_DATA_VERSION and CATALOG_VERSION below CATALOG are omitted, since the FILE_ID is not used in the backpacks, the CATALOG_DATA_VERSION is dependent on CATALOG_MARK and DATA_VERSION, both of which no longer exist, and versioning via the CATALOG_VERSION is not necessary, since it has been decided that the most recent version, which is identified by the FILE_RELEASE_DATE, is always valid.

2. Series identification

2.1. C Attribute SERIE_NO below SERIE becomes SERIE_ID Decision : 2022-09-06

3.1.0	4.0.0
	

Since the 6-digit SERIE_NO was no longer sufficient as a NonNegativeInteger, it becomes the SERIE_ID as a 36-set string.

Description in the documentation:

This attribute is used to specify the serial ID of the base catalogue.

```
<xs:attribute name="SERIE_ID" use="required">
    <xs:annotation> ... </xs:annotation>
    <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:maxLength value="36"/>
            <xs:minLength value="1"/>
        </xs:restriction>
    </xs:simpleType>
</xs:attribute>
```

3. corrected patterns in texts

3.1. F Changed pattern in the TEXT element under ITEM_TEXT Decision : 2023-02-09

3.1.0

```
<xs:element name="TEXT">
    <xs:annotation>...</xs:annotation>
    <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:pattern value="[\S ]{1,30}"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
```

4.0.0.

```
<xs:element name="TEXT">
    <xs:annotation>...</xs:annotation>
    <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:pattern value="[\p{L}\p{N}\p{P}\p{S}][\p{L}\p{N}\p{P}\p{S} ]{0,29}"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
```

Since \S was interpreted differently, it was decided to use a different regular expression. The new pattern now allows uniquely protected spaces in the TEXT under SHORT_TEXT below ITEM_TEXT. The element allows 1-30-character texts.

3.2. F Changed pattern in languagetext30 Decision : 2023-02-09

3.1.0

```
<xs:element name="TEXT">
    <xs:annotation>...</xs:annotation>
    <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:pattern value="[\p{L}\p{N}\p{P}\p{S}][\p{L}\p{N}\p{P}\p{S} ]{1,30}"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
```

4.0.0.

```
<xs:element name="TEXT">
    <xs:annotation>...</xs:annotation>
    <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:pattern value="[\p{L}\p{N}\p{P}\p{S}][\p{L}\p{N}\p{P}\p{S} ]{0,29} "/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
```

The pattern in the TEXT element below the complex type languagetext30 was created incorrectly in the last version, as it allowed 2-31-character texts in its original version. This has now been changed to 1-30-character texts.

3.3. F Changed pattern in languagetext40 Decision : 2023-02-09

3.1.0

```
<xs:element name="TEXT">
    <xs:annotation>...</xs:annotation>
    <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:pattern value="[\p{L}\p{N}\p{P}\p{S}][\p{L}\p{N}\p{P}\p{S} ]{1,40}"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
```

4.0.0.

```
<xs:element name="TEXT">
    <xs:annotation>...</xs:annotation>
    <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:pattern value="[\p{L}\p{N}\p{P}\p{S}][\p{L}\p{N}\p{P}\p{S} ]{0,39}"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
```

The pattern in the TEXT element below the complex type languagetext40 was created incorrectly in the last version, as it allowed 2-41-character texts in its original version. This has now been changed to 1-40-character texts.

3.4. F Changed pattern in languagetext60 Decision : 2023-02-09

3.1.0

```
<xs:element name="TEXT">
    <xs:annotation>...</xs:annotation>
    <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:pattern value="[\p{L}\p{N}\p{P}\p{S}][\p{L}\p{N}\p{P}\p{S} ]{1,60}"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
```

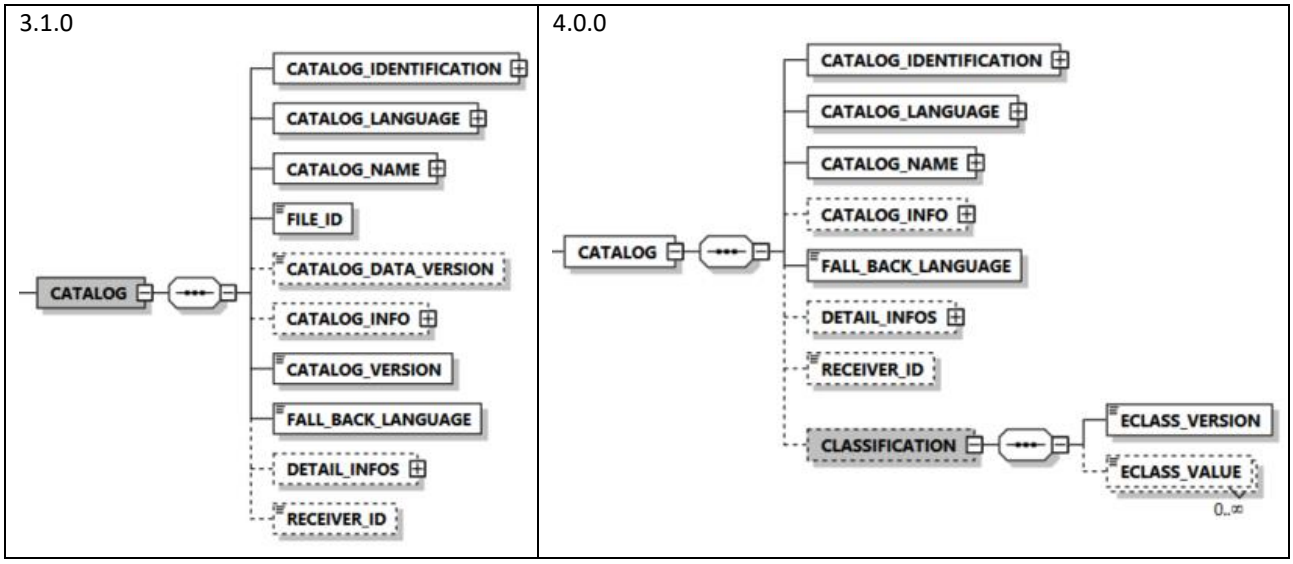
4.0.0.

```
<xs:element name="TEXT">
    <xs:annotation>...</xs:annotation>
    <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:pattern value="[\p{L}\p{N}\p{P}\p{S}][\p{L}\p{N}\p{P}\p{S} ]{0,59}"/>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
```

The pattern in the TEXT element below the complex type languagetext60 was created incorrectly in the last version, as it allowed 2-61-character texts in its original version. This has now been changed to 1-60-character texts.

4. Classification according to ECLASS

4.1. A New optional complex Type CLASSIFICATION under CATALOG Decision : 2023-02-09



As the only classification scheme in the future, ECLASS will be maintained at various nodes in the IDM format. At the catalogue level, the ECLASS version valid for the entire catalogue is stored, as well as ECLASS features that apply to all articles in the catalogue.

CLASSIFICATION:

The optional element CLASSIFICATION below CATALOG is a complexType.

Description in the documentation:

This element is used to store the classification information at catalog level.

If the catalog contains only one classification, the ECLASS_VERSION must be specified.

Information at lower levels for the same properties overwrites the value specified at catalog level.

ECLASS_VERSION:

The element ECLASS_VERSION below CLASSIFICATION is of type integer and mandatory.

Description in the documentation:

The classification version is stored in this element.

Only the major no. of the version is specified.

ECLASS_VALUE:

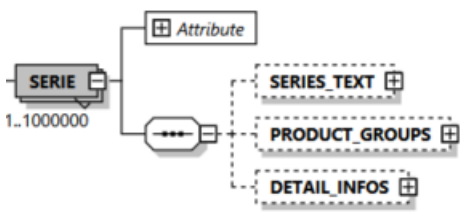
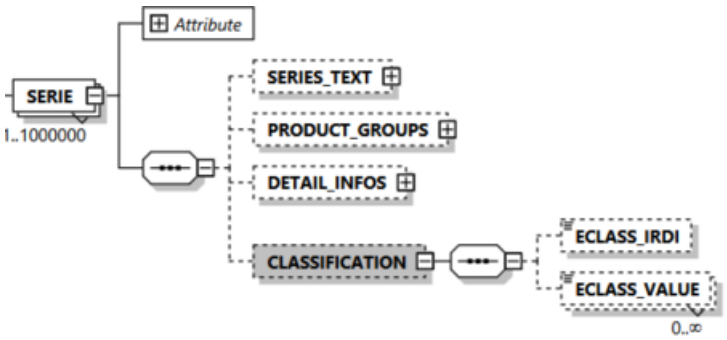
The optional element ECLASS_VALUE below CLASSIFICATION is of type string and can be created as often as desired.

Description in the documentation:

The values of the ECLASS property are stored in this element. For this purpose, the path along the multi-level property structure is specified. The underscore (_) serves as a separator.

```
<xs:element name="CLASSIFICATION" minOccurs="0">
    <xs:annotation>...</xs:annotation>
    <xs:complexType>
        <xs:sequence>
            <xs:element name="ECLASS_VERSION">
                <xs:annotation>...</xs:annotation>
                <xs:simpleType>
                    <xs:restriction base="xs:integer">
                        <xs:pattern value=""/>
                    </xs:restriction>
                </xs:simpleType>
            </xs:element>
            <xs:element name="ECLASS_VALUE" type="xs:string" minOccurs="0" maxOccurs="unbounded">
                <xs:annotation>...</xs:annotation>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
</xs:element>
```

4.2. A New optional complex Type CLASSIFICATION under SERIES Decision : 2023-02-09

3.1.0	4.0.0
	

As the only classification scheme in the future, ECLASS is maintained at various nodes in the IDM format. The ECLASS class and any number of ECLASS characteristics, which apply to all items in the series, are stored on the series.

CLASSIFICATION:

The optional element CLASSIFICATION below SERIES is a complexType.

Description in the documentation:

This element can be used to store classification information at series level.

Specifications at lower levels for the class or the same properties overwrites the value specified at series level.

ECLASS_IRDI:

The optional element ECLASS_IRDI below CLASSIFICATION is of type string and optional. The pattern restricts to the IRDI values possible for classes.

Description in the documentation:

This element stores the IRDI of the 4th level of the hierarchical ECLASS class structure.

ECLASS_VALUE:

The optional element ECLASS_VALUE below CLASSIFICATION is of type string and can be created as often as desired.

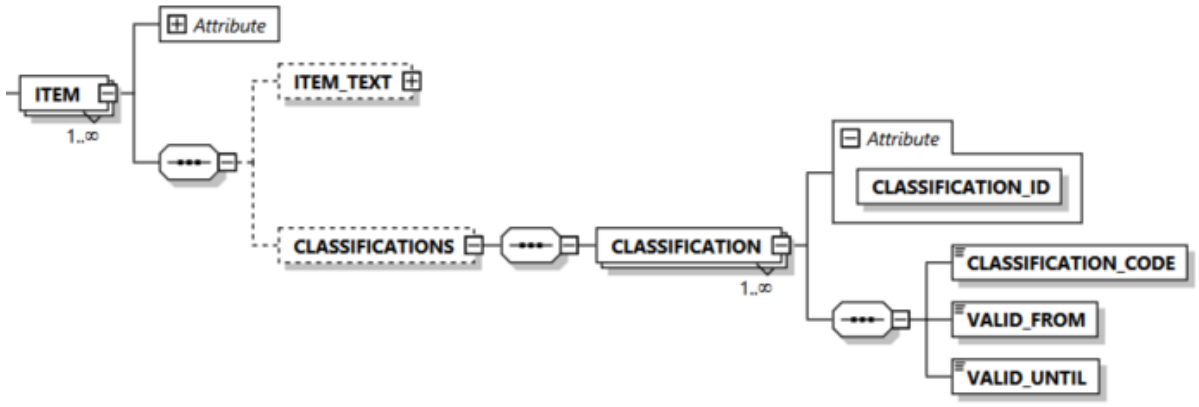
Description in the documentation:

The values of the ECLASS property are stored in this element. For this purpose, the path along the multi-level property structure is specified. The underscore (_) serves as a separator.

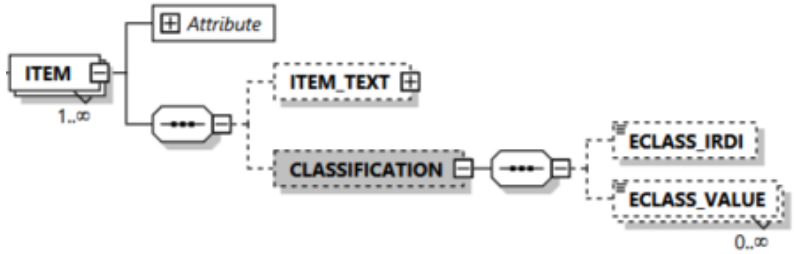
```
<xs:element name="CLASSIFICATION" minOccurs="0">
    <xs:annotation>...</xs:annotation>
    <xs:complexType>
        <xs:sequence>
            <xs:element name="ECLASS_IRDI" minOccurs="0">
                <xs:annotation>...</xs:annotation>
                <xs:simpleType>
                    <xs:restriction base="xs:string">
                        <xs:pattern value="0173-1#01-[A-Z]{3}[0-9]{3}#[0-9]{3}"/>
                    </xs:restriction>
                </xs:simpleType>
            </xs:element>
            <xs:element name="ECLASS_VALUE" type="xs:string" minOccurs="0" maxOccurs="unbounded">
                <xs:annotation>...</xs:annotation>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
</xs:element>
```

4.3. A New elements under CLASSIFICATION below ITEM Decision : 2023-02-09

3.1.0



4.0.0



The previous structure for maintaining classification information of several schemas is replaced by purely ECLASS-related and thus completely new elements. The complex type CLASSIFICATIONS is omitted. The remaining CLASSIFICATION now contains the ECLASS class and any number of ECLASS features.

CLASSIFICATION:

The optional element CLASSIFICATION below ITEM is a complexType.

Description in the documentation:

In this element, the classification information and categorisations are stored at item level.

Information on lower levels for the same ECLASS properties overwrites the value specified at item level.

ECLASS_IRDI:

The optional element ECLASS_IRDI below CLASSIFICATION is of type string. The pattern restricts to the IRDI values possible for classes.

Description in the documentation:

This element stores the IRDI of the 4th level of the hierarchical ECLASS class structure.

ECLASS_VALUE:

The optional element ECLASS_VALUE below CLASSIFICATION is of type string and can be created as often as desired.

Description in the documentation:

The values of the ECLASS property are stored in this element. For this purpose, the path along the multi-level property structure is specified. The underscore (_) serves as a separator.

5. TYPE_NO without spaces at the beginning and end

5.1.	F	New pattern in the TYPE_NO element	Decision : 2022-04-27

3.1.0

```
<xs:attribute name="TYPE_NO" use="required">
    <xs:annotation>...</xs:annotation>
    <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:minLength value="1"/>
            <xs:maxLength value="30"/>
        </xs:restriction>
    </xs:simpleType>
</xs:attribute>
```

4.0.0

```
<xs:attribute name="TYPE_NO" use="required">
    <xs:annotation>...</xs:annotation>
    <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:minLength value="1"/>
            <xs:maxLength value="30"/>
            <xs:pattern value="\S(.{0,28}\S)?"/>
        </xs:restriction>
    </xs:simpleType>
</xs:attribute>
```

The TYPE_NO attribute previously allowed all characters. With the new pattern, spaces at the beginning and end are excluded.

6. Textual changes in the documentation

6.1.	C	Replace IDMP with IDML in the element T_MODIFY_CONTENT_CATALOG	Decision :

Element: T_MODIFY_CONTENT_CATALOG

3.1.0 This element offers an entry point for modified texts and designations, trade-specific media or additional classifications with respect to a catalogue in IDMP format. This data can be created by the trade as well as associations and similar, who would like to transfer texts, media and/or classifications that were adjusted for items and versions. These texts and media are only relevant for sales. For orders, the texts and designations of the original format must be used in IDMP format. If an element in the data pool references media, the reference must specify all relevant media, including those to be used by the base catalogue. This in particular applies to magnet planner information.
4.0.0 This element offers an entry point for modified texts and designations, trade-specific media or additional classifications with respect to a catalogue in IDML format. This data can be created by the trade as well as associations and similar, who would like to transfer texts, media and/or classifications that were adjusted for items and versions. These texts and media are only relevant for sales. For orders, the texts and designations of the original format must be used in IDML format. If an element in the data pool references media, the reference must specify all relevant media, including those to be used by the base catalogue. This in particular applies to magnet planner information.

6.2.	C	Concretisation of the changes in the referenced node under basic structure	Decision : 2022-03-09

Introduction -> Basic structure

3.1.0

The basic structure of the IDMP schema

The main element T_MODIFY_CONTENT_CATALOG defines that a new and complete catalogue data pool is made available.
The IDM schema is subdivided in data ranges that are specified in the subsequent child-elements (see figure 1) in T_MODIFY_CONTENT_CATALOG.

4.0.0

The basic structure of the IDML scheme

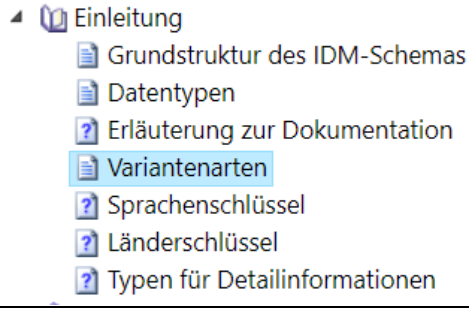
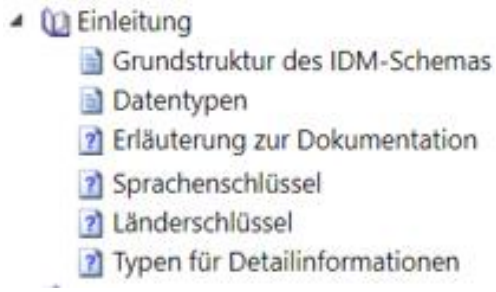
The main element T_MODIFY_CONTENT_CATALOG defines that a new, complete catalogue dataset is provided.

The MODIFY_CONTENT_IDMP completely replaces the content of the basic catalogue in the respective referenced node. If, for example, DETAIL_INFOS from the basic catalogue are to be retained at a node, they must also be contained in the MODIFY_CONTENT_IDMP. The subordinate structures in the hierarchy remain unaffected by the change.

The overriding information from the MODIFY_CONTENT_IDML is not intended for order processing. The origin IDs and designations from the basic catalogue must be specified there.

The IDM schema is divided into data areas, which are mapped by the following child elements (see Figure 1) of T_MODIFY_CONTENT_CATALOG.

6.3. R Deleting the variant types from the introduction Decision :

3.1.0	4.0.0
	

Since the schema no longer contains any features, the version types can also be deleted from the introduction.

6.4. C Rename the detailed information 5 and 8 Decision : 2022-01-27

3.1.0	4.0.0
The following info types can be used to describe the type of detail information: 1 = Milieu photo 2 = Item pictogram (type/item overview) 3 = Magnetic planner drawing (SVGs in top view) 4 = Version photo 5 = Version texture (leather type, ...) 6 = Manufacturer logo/product brand 7 = Colour consultancy (the versions are defined in an assessment procedure) 8 = Fabric sample book (PDF for fabric groups) 9 = Model description sheets (PDF below series) 10 = Installation instructions 11 = Marketing 12 = Miscellaneous 13 = Colour consultancy wall 14 = Colour consultancy floor 15 = Description of function 16 = Planning advice 17 = Catalogue image Only info types 1, 9, 10 and 17 are permitted in the MODIFY-CONTENT-IDMP.	The following info types are available for describing the type of detailed information: 1 = Milieu photo 2 = Item pictogram (type/item overview) 3 = Magnetic planner drawing (SVGs in top view) 4 = Variant photo 5 = Variant texture 6 = Manufacturer logo/product brand 7 = Colour consultancy (the versions are defined in an assessment procedure) 8 = Material passport (PDF) 9 = Model description sheets (PDF below series) 10 = Installation instructions 11 = Marketing 12 = Miscellaneous 13 = Colour consultancy wall 14 = Colour consultancy floor 15 = Description of function 16 = Planning advice 17 = Catalogue image In the MODIFY-CONTENT-IDML, only info types 1, 9, 10 and 17 are allowed.

The terms have been standardised for upholstery and home furniture.

6.5. C Addition of the recommended image formats in the INFO_TYPE element Decision : 2022-04-27

INFO_TYPE:

3.1.0 This element is here to describe the type of detailed information. The values of the available info types can be found in the corresponding table in the introduction. Media references for descriptions of function are possible both on the ITEM and on the SERIES. Format MP4 is recommended for videos.
4.0.0 This element describes the type of detailed information. The values of the possible info types can be found in the corresponding table from the introduction. Media referencing for function descriptions is possible on both the ITEM and the SERIES. The MP4 format is recommended for videos. For images, an image size of 2048x2048px and the following media formats are recommended: - PNG - TIF/TIFF - JPG/JPEG - SVG

The image size is now recommended for all images and there is a recommendation on the common image formats that can be processed by the software houses.

6.6. C Rename SERIE_NO to SERIE_ID in the ITEM Decision : 2022-09-06

3.1.0 In this element, articles are stored according to the basic catalogue. An article is identified by the attributes SERIE_NO (path 1) and TYPE_NO (path 2). Path 1: SERIES/SERIES_NO Path 2: SERIES/SERIES/PRODUCT_GROUPS/PRODUCT_GROUP/ITEMS/ITEM/TYPE_NO
4.0.0 In this element, articles are stored according to the basic catalogue. An article is identified by the attributes SERIE_ID (path 1) and TYPE_NO (path 2). Path 1: SERIES/SERIES_ID Path 2: SERIES/SERIES/PRODUCT_GROUPS/PRODUCT_GROUP/ITEMS/ITEM/TYPE_NO

Due to the renaming of the attributes SERIE_NO to SERIE_ID, the description in the element ITEM was adjusted accordingly.

6.7. C Case insensitivity for _KEY, _ID and _NO elements Decision : 2023-02-09

Add the sentence: "The element/attribute must be unique, regardless of upper and lower case", in the descriptions of the following elements/attributes:

CATALOG_ID:

- T_MODIFY_CONTENT_CATALOG/CATALOG/CATALOG_IDENTIFICATION/CATALOG_ID
- T_MODIFY_CONTENT_CATALOG/REF_CATALOG/@CATALOG_ID

RECEIVER_ID :

- T_MODIFY_CONTENT_CATALOG/CATALOG/RECEIVER_ID

SERIES_ID:

- T_MODIFY_CONTENT_CATALOG/SERIES/SERIES/@SERIES_ID

TYPE_NO:

- T_MODIFY_CONTENT_CATALOG/SERIES/PRODUCT_GROUPS/PRODUCT_GROUP/ITEMS/ITEM/@TYPE_NO

Contact

Data Competence Center e. V.

Goebenstraße 4-10

32052 Herford

Info: www.dcc-moebel.org

Dr.-Ing. Olaf Plümer

E-mail: pluemer@dcc-moebel.org

Phone +49 52 21 / 12 65-37

Anika Degenhard

E-mail: degenhard@dcc-moebel.org

Tel.: +49 52 21 / 12 65 - 38