



# Changelog MODIFY-IDM

## Kitchen/Bathroom Version 3.0.0

This changelog describes all innovations, enhancements and corrections that are made available with the current version **MODIFY-IDM 3.0.0** (compared to version MODIFY-IDM 2.8.1) both in the XML schema and in the documentation.

Version MODIFY-IDM 3.0.0 is published on 12.12.2022 and becomes valid from 01.02.2023.

### Content:

Decision of: Type Page  
(date)

Released	<b>1. Catalogue identification and versioning</b>				2
	1.1.	New element CATALOG_ID under CATALOG_IDENTIFICATION	2021-09-29	A	2
	1.2.	CATALOG_YEAR under CATALOG is deleted	2021-09-29	R	3
	1.3.	Move MAJOR_NO and FILE_RELEASE_DATE to CATALOG_IDENTIFICATION	2021-09-29	C	4
	1.4.	New element MINOR_NO under CATALOG_IDENTIFICATION	2021-09-29	A	5
	1.5.	CATALOG_NUMBER below CATALOG is deleted	2021-09-29	R	6
	<b>2. Classification according to ECLASS</b>				7
	2.1.	New optional complex Type CLASSIFICATION under CATALOG	2022-11-17	A	7
	2.2.	New optional complex Type CLASSIFICATION under SERIES	2022-11-17	A	10
	2.3.	New optional complex Type CLASSIFICATION under ITEM	2022-11-17	A	13
	2.4.	New optional complex type ECLASS_PROPERTY under FEATURE	2022-11-17	A	16
	2.5.	New optional choice between ECLASS_PROPERTY and ECLASS_VALUE under OPTION	2022-11-17	A	17
	<b>3. Regular expression for key elements</b>				20
	3.1.	Changed pattern in the TYPE_NO element under ITEM	2022-03-22	C	20
	<b>4. Textual changes in the documentation</b>				21
4.1.	Extended descriptions of the variant types according to CSV documentation	2022-03-22	C	21	
4.2.	Explanatory example for multi-language catalogues (e.g. Switzerland) on the elements COUNTRY_ID and ISO_LANGUAGE_ID	2022-03-22	C	22	
4.3.	Note on handling special colours in the OPTION element	2022-03-22	C	23	

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.

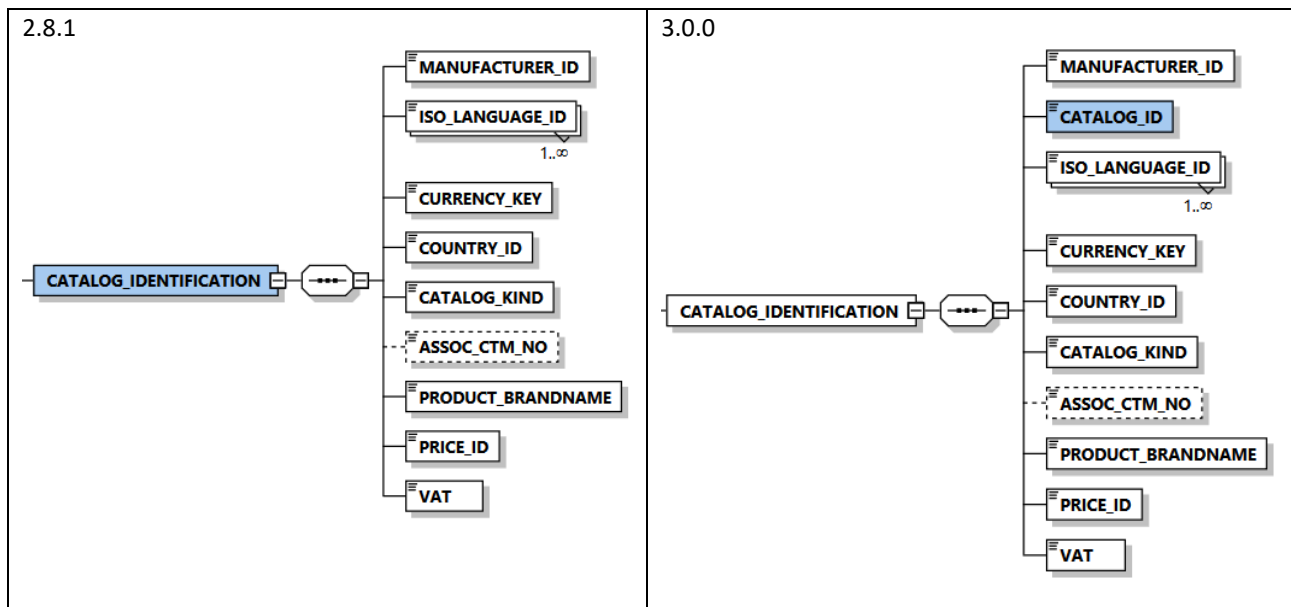


# Released

Display of all changes noted for version 3.0.0

## 1. Catalogue identification and versioning

1.1. A New element CATALOG\_ID under CATALOG\_IDENTIFICATION Decision : 2021-09-29



The element CATALOG\_ID under CATALOG\_IDENTIFICATION is a mandatory field with max. 64 characters and uniquely identifies the catalogue together with the MANUFACTURER\_ID.

Description in the documentation:

The manufacturer must enter a unique ID (e.g. GUID) that unequivocally identifies the catalogue here.

A manufacturer catalogue must always have the same ID for the same dealership.

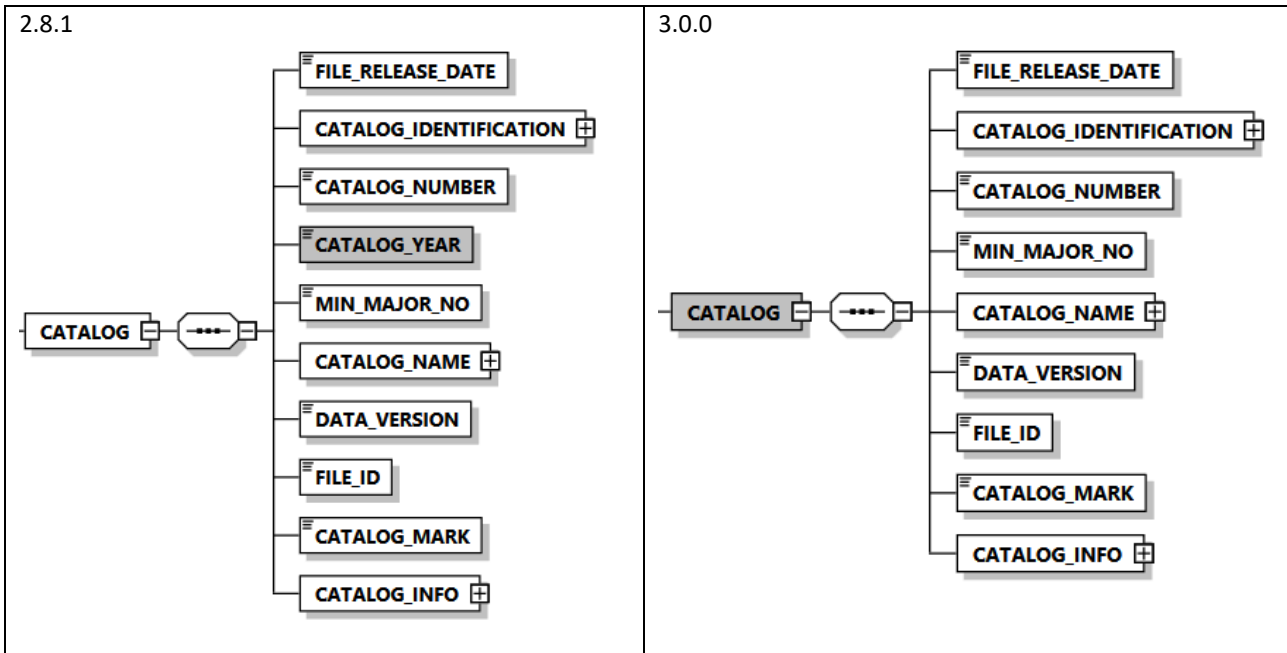
```
<xs:element name="CATALOG_ID">
  <xs:annotation>...</xs:annotation>
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:minLength value="1"/>
      <xs:maxLength value="64"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

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.

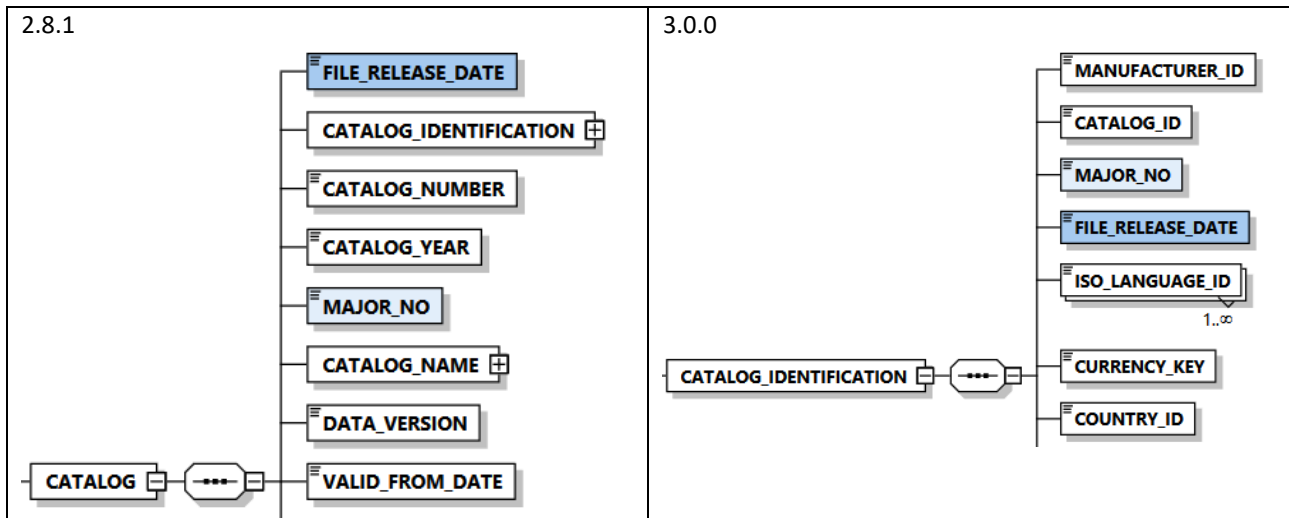


The element CATALOG\_YEAR under CATALOG is omitted. It is replaced by the element MAJOR\_NO under CATALOG\_IDENTIFICATION.



1.3. C Move MAJOR\_NO and FILE\_RELEASE\_DATE to CATALOG\_IDENTIFICATION

Decision : 2021-09-29



The elements MAJOR\_NO and FILE\_RELEASE\_DATE version the catalogue uniquely together with the new element MINOR\_NO.

The FILE\_RELEASE\_DATE remains unchanged.

### MAJOR\_NO:

The existing element MAJOR\_NO is changed in its function. It now indicates the year of entry of the main catalogue.

Description in the documentation:

The catalogue version number MAJOR\_NO indicates the year in which the main catalogue was created. The complete specification of the catalogue version in which a dataset is delivered consists of MAJOR and MINOR.

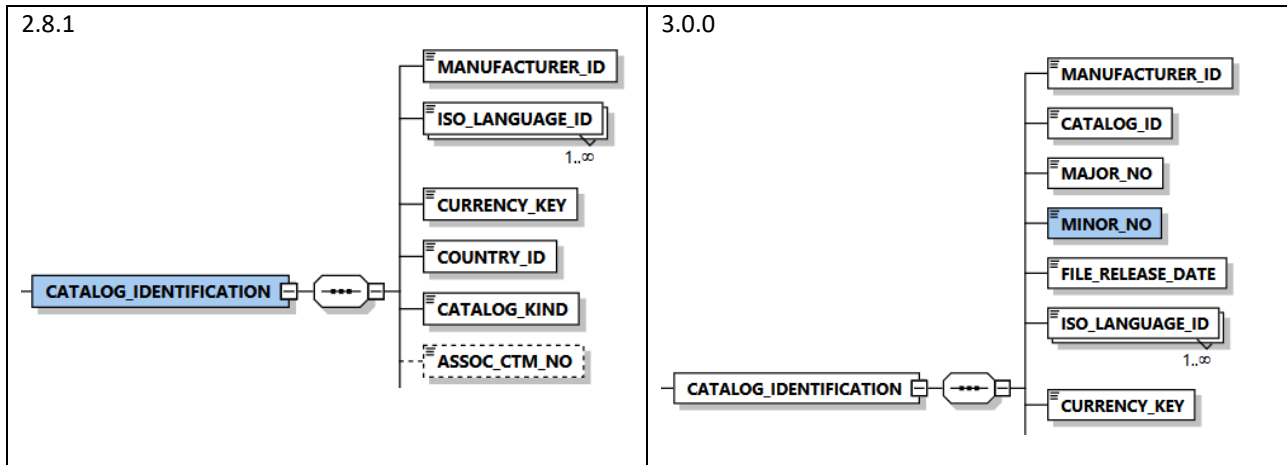
```
<xs:element name="MAJOR_NO">
  <xs:annotation>...</xs:annotation>
  <xs:simpleType>
    <xs:restriction base="xs:gYear"/>
  </xs:simpleType>
</xs:element>
<xs:element name="MINOR_NO">...</xs:element>
<xs:element name="FILE_RELEASE_DATE" type="xs:dateTime">
  <xs:annotation>...</xs:annotation>
</xs:element>
```

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.



The element MINOR\_NO is a mandatory field and contains consecutive numbers as a sub-version number to each MAJOR\_NO.

Description in the documentation:

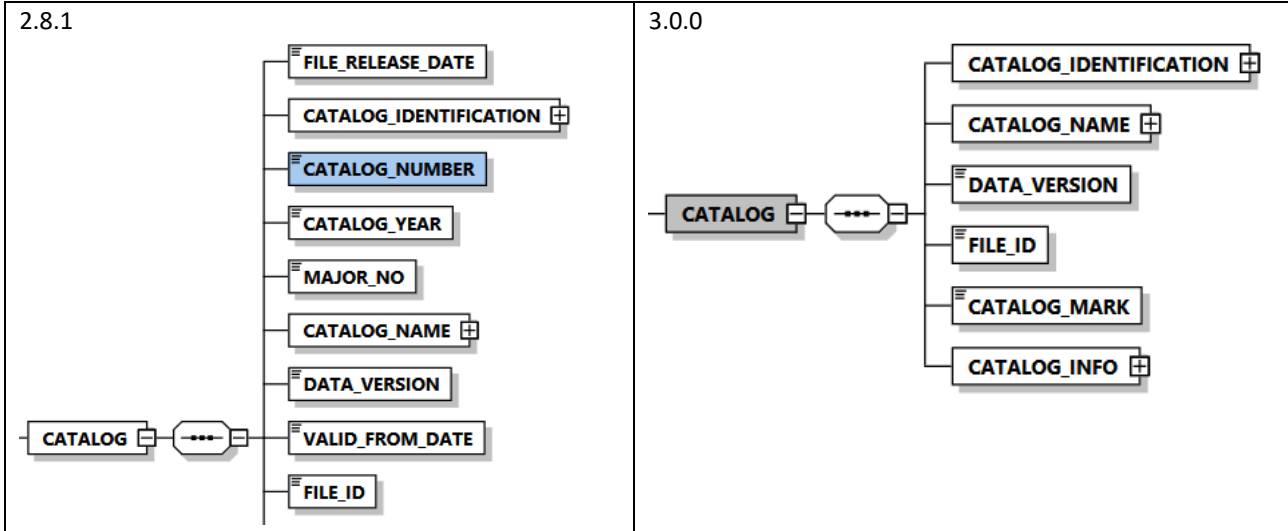
The sub-catalogue version number MINOR\_NO is given in consecutive numbers during the year or the period of validity of the main catalogue. The complete specification of the catalogue version in which a dataset is delivered consists of MAJOR and MINOR. If the catalogue version MAJOR is changed, the sub-catalogue version is reset to 0.

```
<xs:element name="MINOR_NO">
  <xs:annotation> ... </xs:annotation>
  <xs:simpleType>
    <xs:restriction base="xs:nonNegativeInteger">
      <xs:minInclusive value="1"/>
      <xs:maxInclusive value="999"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```



1.5. R CATALOG\_NUMBER below CATALOG is deleted

Decision : 2021-09-29

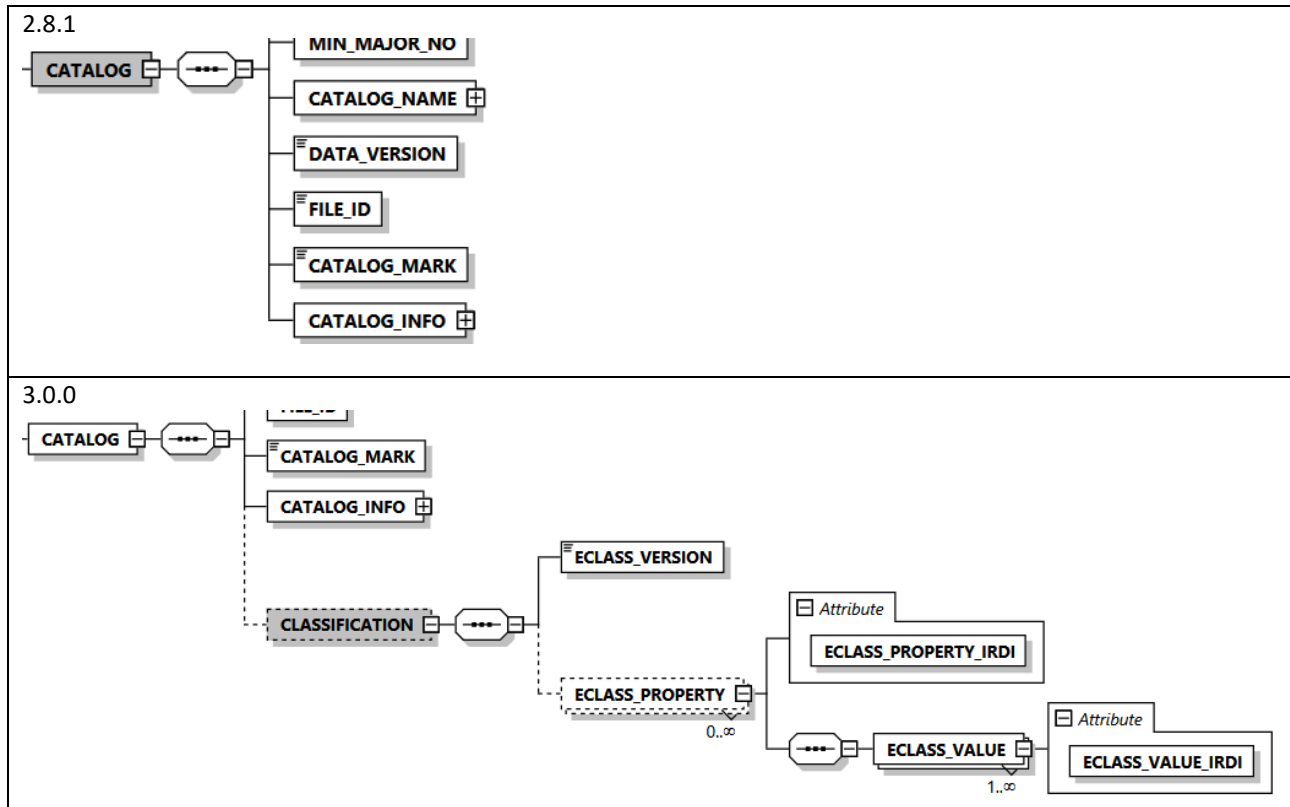


The element CATALOG\_NUMBER under CATALOG is omitted.



## 2. Classification according to ECLASS

2.1. A New optional complex Type CLASSIFICATION under CATALOG Decision : 2022-11-17



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 properties that apply to all items 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.

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.



### **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\_PROPERTY:**

The optional element ECLASS\_PROPERTY below CLASSIFICATION is a complexType and can be created as often as desired.

#### Description in the documentation:

In this element the ECLASS properties are stored on catalog level.

Specifications at lower levels for the same properties overwrite the value specified at catalog level.

### **ECLASS\_PROPERTY\_IRDI:**

The attribute ECLASS\_PROPERTY\_IRDI at the ECLASS\_PROPERTY is of type string and mandatory. The pattern restricts to the IRDI values possible for characteristics.

#### Description in the documentation:

The attribute identifies the ECLASS property via the IRDI.

### **ECLASS\_VALUE:**

The element ECLASS\_VALUE below ECLASS\_PROPERTY is a complexType, mandatory and can be created multiple times for MultipleChoice characteristics.

#### Description in the documentation:

The values of the ECLASS property are stored in this element.

### **ECLASS\_VALUE\_IRDI:**

The attribute ECLASS\_VALUE\_IRDI on ECLASS\_VALUE is of type string and mandatory. The pattern restricts to the IRDI values possible for characteristic values.

#### Description in the documentation:

The attribute identifies the value of the ECLASS property via the IRDI.

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.





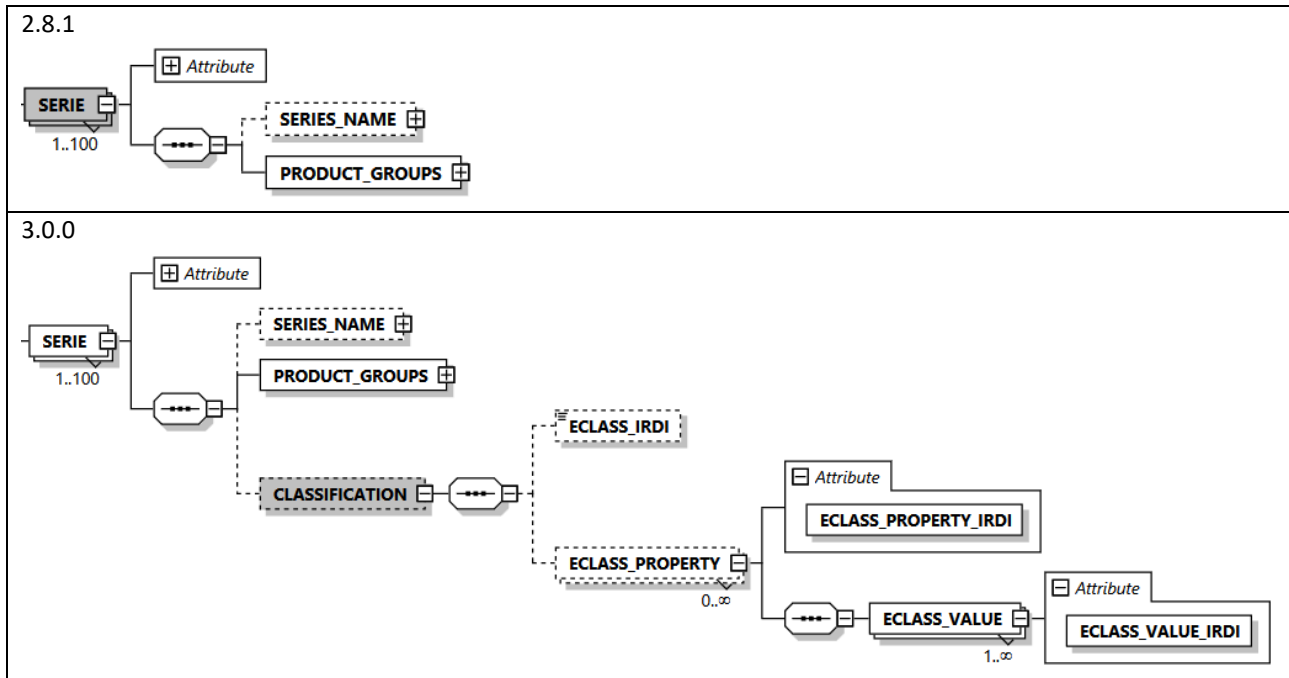
```
<xs:element name="CLASSIFICATION" minOccurs="0">
  <xs:annotation>...</xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element name="ECLASS_VERSION">
        <xs:annotation>...</xs:annotation>
        <xs:simpleType>
          <xs:restriction base="xs:integer">
            <xs:pattern value=""/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="ECLASS_PROPERTY" minOccurs="0" maxOccurs="unbounded">
        <xs:annotation>...</xs:annotation>
        <xs:complexType>
          <xs:sequence>
            <xs:element name="ECLASS_VALUE" maxOccurs="unbounded">
              <xs:annotation>...</xs:annotation>
              <xs:complexType>
                <xs:attribute name="ECLASS_VALUE_IRDI" use="required">
                  <xs:annotation>...</xs:annotation>
                  <xs:simpleType>
                    <xs:restriction base="xs:string">
                      <xs:pattern value="0173-1#07-[A-Z]{3}[0-9]{3}#[0-9]{39}"/>
                    </xs:restriction>
                  </xs:simpleType>
                </xs:attribute>
              </xs:complexType>
            </xs:element>
          </xs:sequence>
          <xs:attribute name="ECLASS_PROPERTY_IRDI" use="required">
            <xs:annotation>...</xs:annotation>
            <xs:simpleType>
              <xs:restriction base="xs:string">
                <xs:pattern value="0173-1#02-[A-Z]{3}[0-9]{3}#[0-9]{3}"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:attribute>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

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.



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 properties, which apply to all items in the series, are stored on the series.

#### CLASSIFICATION:

The optional element CLASSIFICATION below SERIE 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.

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.



### **ECLASS\_PROPERTY:**

The optional element ECLASS\_PROPERTY below CLASSIFICATION is a complexType and can be created as often as desired.

#### Description in the documentation:

In this element the ECLASS properties are stored on series level.

### **ECLASS\_PROPERTY\_IRDI:**

The attribute ECLASS\_PROPERTY\_IRDI at the ECLASS\_PROPERTY is of type string and mandatory. The pattern restricts to the IRDI values possible for characteristics.

#### Description in the documentation:

The attribute identifies the ECLASS property via the IRDI.

### **ECLASS\_VALUE:**

The element ECLASS\_VALUE below ECLASS\_PROPERTY is a complexType, mandatory and can be created multiple times for MultipleChoice characteristics.

#### Description in the documentation:

The values of the ECLASS property are stored in this element.

### **ECLASS\_VALUE\_IRDI:**

The attribute ECLASS\_VALUE\_IRDI on ECLASS\_VALUE is of type string and mandatory. The pattern restricts to the IRDI values possible for characteristic values.

#### Description in the documentation:

The attribute identifies the value of the ECLASS property via the IRDI.



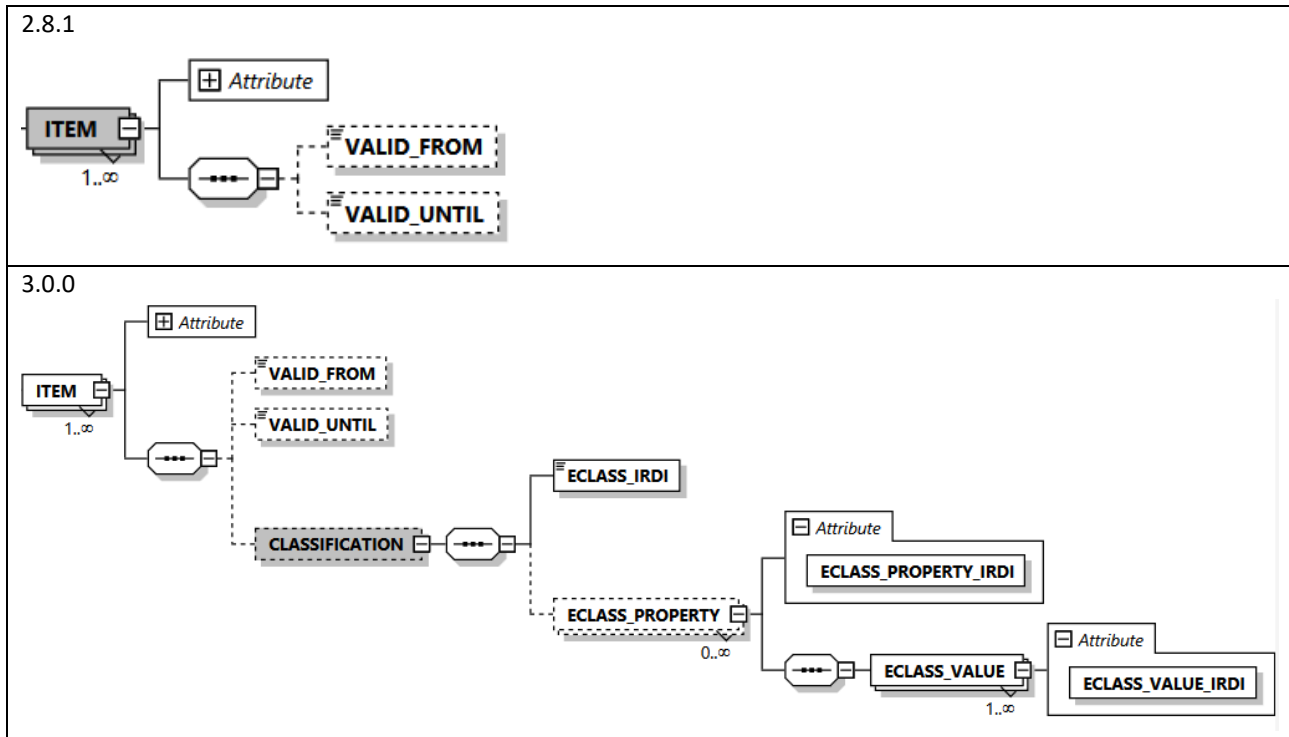
```
<xs:element name="CLASSIFICATION" minOccurs="0">
  <xs:annotation>[...]</xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element name="ECLASS_IRDI" minOccurs="0">
        <xs:annotation>[...]</xs:annotation>
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <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_PROPERTY" minOccurs="0" maxOccurs="unbounded">
        <xs:annotation>[...]</xs:annotation>
        <xs:complexType>
          <xs:sequence>
            <xs:element name="ECLASS_VALUE" maxOccurs="unbounded">
              <xs:annotation>[...]</xs:annotation>
              <xs:complexType>
                <xs:attribute name="ECLASS_VALUE_IRDI" use="required">
                  <xs:annotation>[...]</xs:annotation>
                  <xs:simpleType>
                    <xs:restriction base="xs:string">
                      <xs:pattern value="0173-1#07-[A-Z]{3}[0-9]{3}#[0-9]{39}"/>
                    </xs:restriction>
                  </xs:simpleType>
                </xs:attribute>
              </xs:complexType>
            </xs:element>
          </xs:sequence>
          <xs:attribute name="ECLASS_PROPERTY_IRDI" use="required">
            <xs:annotation>[...]</xs:annotation>
            <xs:simpleType>
              <xs:restriction base="xs:string">
                <xs:pattern value="0173-1#02-[A-Z]{3}[0-9]{3}#[0-9]{3}"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:attribute>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

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.



As the only classification scheme in the future, ECLASS will be maintained at various nodes in IDM format. The ECLASS class and any number of ECLASS properties are stored on the item.

#### CLASSIFICATION:

The optional element CLASSIFICATION below ITEM is a complexType.

##### Description in the documentation:

In this element, the classification information is stored at item level.

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

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.



### **ECLASS\_PROPERTY:**

The optional element ECLASS\_PROPERTY below CLASSIFICATION is a complexType and can be created as often as desired.

#### Description in the documentation:

In this element the ECLASS properties are stored on item level.

### **ECLASS\_PROPERTY\_IRDI:**

The attribute ECLASS\_PROPERTY\_IRDI at the ECLASS\_PROPERTY is of type string and mandatory. The pattern restricts to the IRDI values possible for characteristics.

#### Description in the documentation:

The attribute identifies the ECLASS property via the IRDI.

### **ECLASS\_VALUE:**

The element ECLASS\_VALUE below ECLASS\_PROPERTY is a complexType, mandatory and can be created multiple times for MultipleChoice characteristics.

#### Description in the documentation:

The values of the ECLASS property are stored in this element.

### **ECLASS\_VALUE\_IRDI:**

The attribute ECLASS\_VALUE\_IRDI on ECLASS\_VALUE is of type string and mandatory. The pattern restricts to the IRDI values possible for characteristic values.

#### Description in the documentation:

The attribute identifies the value of the ECLASS property via the IRDI.



```
<xs:element name="CLASSIFICATION" minOccurs="0">
  <xs:annotation>[...]</xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element name="ECLASS_IRDI">
        <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_PROPERTY" minOccurs="0" maxOccurs="unbounded">
        <xs:annotation>[...]</xs:annotation>
        <xs:complexType>
          <xs:sequence>
            <xs:element name="ECLASS_VALUE" maxOccurs="unbounded">
              <xs:annotation>[...]</xs:annotation>
              <xs:complexType>
                <xs:attribute name="ECLASS_VALUE_IRDI" use="required">
                  <xs:annotation>[...]</xs:annotation>
                  <xs:simpleType>
                    <xs:restriction base="xs:string">
                      <xs:pattern value="0173-1#07-[A-Z]{3}[0-9]{3}#[0-9]{39}"/>
                    </xs:restriction>
                  </xs:simpleType>
                </xs:attribute>
              </xs:complexType>
            </xs:element>
          </xs:sequence>
          <xs:attribute name="ECLASS_PROPERTY_IRDI" use="required">
            <xs:annotation>[...]</xs:annotation>
            <xs:simpleType>
              <xs:restriction base="xs:string">
                <xs:pattern value="0173-1#02-[A-Z]{3}[0-9]{3}#[0-9]{3}"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:attribute>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

A = Added = Adding new elements or attributes

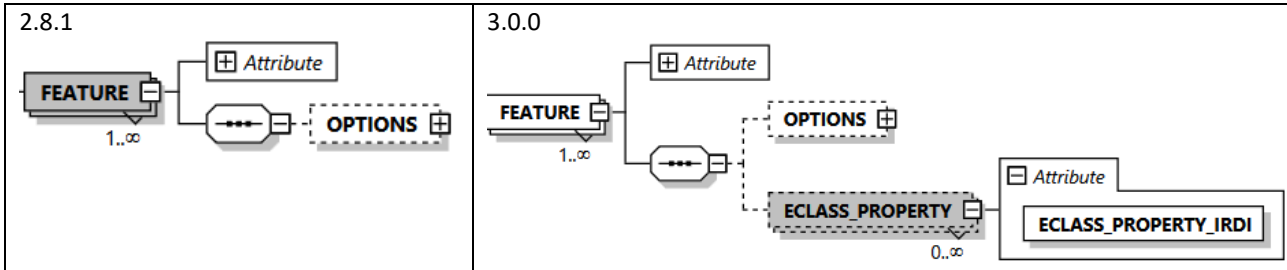
C = Changed = Changes to existing elements, attributes or descriptive texts.

R = Removed = Deletion of elements or attributes

F = Fixed = Error corrections to existing elements, attributes or descriptive texts.



2.4. A New optional complex type ECLASS\_PROPERTY under FEATURE Decision : 2022-11-17



As the only classification scheme in the future, ECLASS is maintained at various nodes in IDM format. Any number of ECLASS features are stored at the feature.

### ECLASS\_PROPERTY:

The optional element ECLASS\_PROPERTY below FEATURE is a complexType and can be created as often as desired.

Description in the documentation:

In this element the ECLASS properties are stored at the feature.

### ECLASS\_PROPERTY\_IRDI:

The attribute ECLASS\_PROPERTY\_IRDI at the ECLASS\_PROPERTY is of type string and mandatory. The pattern restricts to the IRDI values possible for characteristics.

Description in the documentation:

The attribute identifies the ECLASS property via the IRDI.

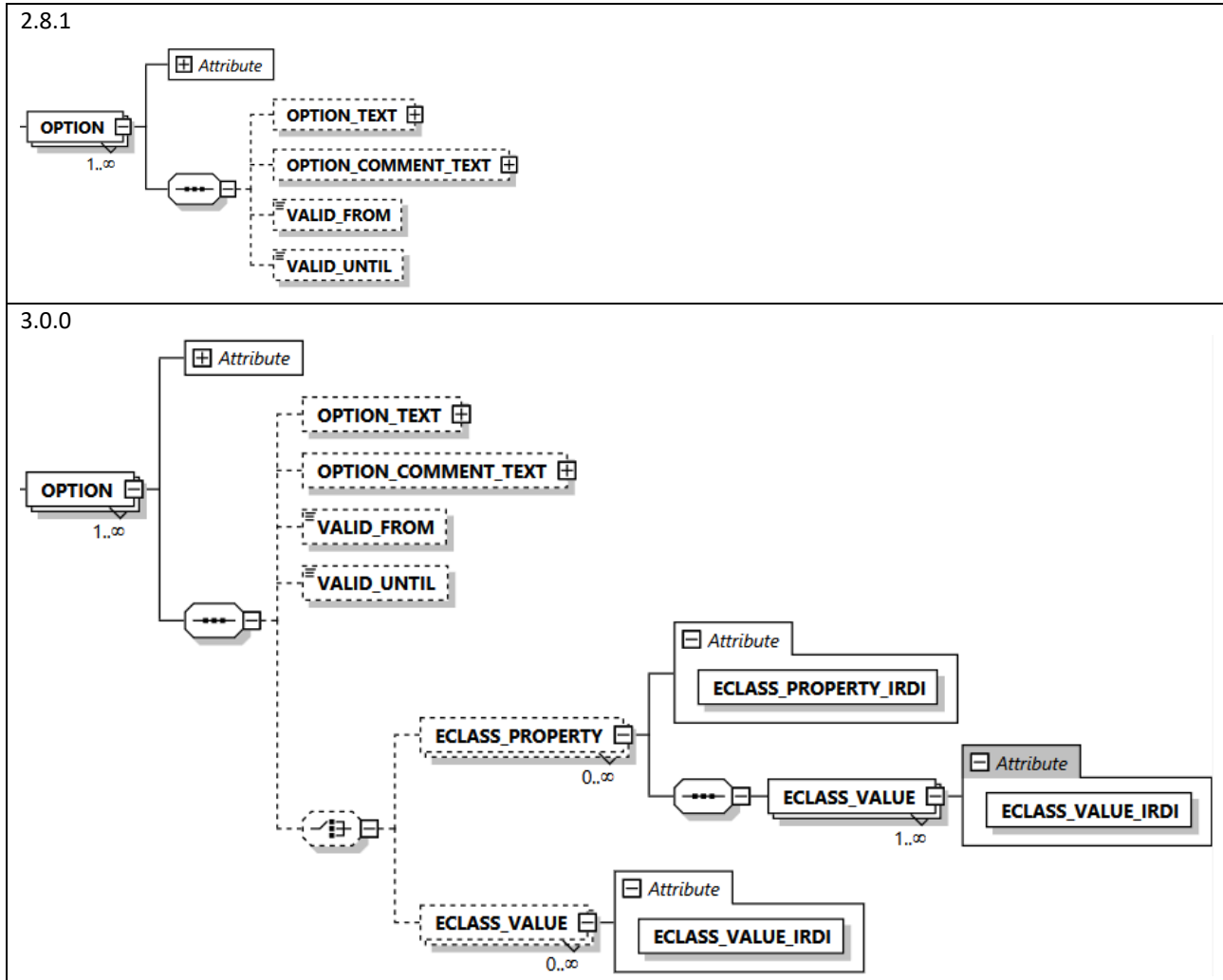
```
<xs:element name="ECLASS_PROPERTY" minOccurs="0" maxOccurs="unbounded">
  <xs:annotation>...</xs:annotation>
  <xs:complexType>
    <xs:attribute name="ECLASS_PROPERTY_IRDI" use="required">
      <xs:annotation>...</xs:annotation>
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:pattern value="0173-1#02-[A-Z]{3}[0-9]{3}#[0-9]{3}"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
  </xs:complexType>
</xs:element>
```





2.5. A New optional choice between ECLASS\_PROPERTY and ECLASS\_VALUE under OPTION

Decision : 2022-11-17



As the only classification scheme in the future, ECLASS is maintained at various nodes in IDM format. Any number of ECLASS properties and ECLASS property values can be stored at the option.

#### **ECLASS\_PROPERTY:**

The optional element ECLASS\_PROPERTY under the Choice below OPTION is a complexType and can be created as often as desired.

#### Description in the documentation:

In this element the ECLASS properties are stored at the option.

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.



### **ECLASS\_PROPERTY\_IRDI:**

The attribute ECLASS\_PROPERTY\_IRDI at the ECLASS\_PROPERTY is of type string and mandatory. The pattern restricts to the IRDI values possible for characteristics.

#### Description in the documentation:

The attribute identifies the ECLASS property via the IRDI.

### **ECLASS\_VALUE:**

The element ECLASS\_VALUE below ECLASS\_PROPERTY is a complexType, mandatory and can be created multiple times for MultipleChoice characteristics.

#### Description in the documentation:

The values of the ECLASS property are stored in this element.

### **ECLASS\_VALUE\_IRDI:**

The attribute ECLASS\_VALUE\_IRDI at ECLASS\_VALUE below ECLASS\_PROPERTY is of type string and mandatory. The pattern restricts to the IRDI values possible for characteristic values.

#### Description in the documentation:

The attribute identifies the value of the ECLASS property via the IRDI.

### **ECLASS\_VALUE:**

The optional element ECLASS\_VALUE under the Choice below OPTION is a complexType and can be created as often as desired.

#### Description in the documentation:

The values of an ECLASS property are stored in this element.

### **ECLASS\_VALUE\_IRDI:**

The attribute ECLASS\_VALUE\_IRDI at ECLASS\_VALUE below OPTION is of type string and mandatory. The pattern restricts to the IRDI values possible for characteristic values.

#### Description in the documentation:

The attribute identifies the value of an ECLASS property via the IRDI.



```
<xs:choice minOccurs="0">
  <xs:element name="ECLASS_PROPERTY" minOccurs="0" maxOccurs="unbounded">
    <xs:annotation>...</xs:annotation>
    <xs:complexType>
      <xs:sequence>
        <xs:element name="ECLASS_VALUE" maxOccurs="unbounded">
          <xs:annotation>...</xs:annotation>
          <xs:complexType>
            <xs:attribute name="ECLASS_VALUE_IRDI" use="required">
              <xs:annotation>...</xs:annotation>
              <xs:simpleType>
                <xs:restriction base="xs:string">
                  <xs:pattern value="0173-1#07-[A-Z]{3}[0-9]{3}#[0-9]{39}"/>
                </xs:restriction>
              </xs:simpleType>
            </xs:attribute>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
      <xs:attribute name="ECLASS_PROPERTY_IRDI" use="required">
        <xs:annotation>...</xs:annotation>
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:pattern value="0173-1#02-[A-Z]{3}[0-9]{3}#[0-9]{3}"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
    </xs:complexType>
  </xs:element>
  <xs:element name="ECLASS_VALUE" minOccurs="0" maxOccurs="unbounded">
    <xs:annotation>...</xs:annotation>
    <xs:complexType>
      <xs:attribute name="ECLASS_VALUE_IRDI" use="required">
        <xs:annotation>...</xs:annotation>
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:pattern value="0173-1#07-[A-Z]{3}[0-9]{3}#[0-9]{39}"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
    </xs:complexType>
  </xs:element>
</xs:choice>
```

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.



### 3. Regular expression for key elements

7.1. C Changed pattern in the TYPE\_NO element under ITEM

Decision : 2022-03-22

2.8.1

```
<xs:element name="ITEM" maxOccurs="unbounded">
  <xs:annotation>...</xs:annotation>
  <xs:complexType>
    <xs:sequence>...</xs:sequence>
    <xs:attribute name="SERIE_NO" use="required">...</xs:attribute>
    <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="15"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
  </xs:complexType>
</xs:element>
```

3.0.0

```
<xs:element name="ITEM" maxOccurs="unbounded">
  <xs:annotation>...</xs:annotation>
  <xs:complexType>
    <xs:sequence>...</xs:sequence>
    <xs:attribute name="SERIE_NO" use="required">...</xs:attribute>
    <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="15"/>
          <xs:pattern value="\S{0,28}\S?" />
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
  </xs:complexType>
</xs:element>
```

To minimise erroneous type numbers, the new pattern in the TYPE\_NO element under ITEM excludes spaces at the beginning and end.

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.



## 4. Textual changes in the documentation

### 4.1. C Extended descriptions of the variant types according to CSV documentation

Decision : 2022-03-22

2.8.1			
Variation type	Variation text	Option type	Remark
105	Front combination	K	If only one key is used for different front information.
201	Carcase combination	K	If only one key is used for several – interior /exterior - carcase colours.
303	Combination of handles	K	If only one key is defined for several handle conditions
999	Fixed material assignment	F	

3.0.0			
Variation type	Variation text	Option type	Remark
105	Front combination	K	If only one key is given for several front information. If variant type 105 is filled, it overrides variant types 100-104 to the extent that they are only for internal use by the software companies (e.g. internal colour control). These are then set as invisible header variants ("U") in the element HEADER_POS_VARIATION_TYPE. For external use as a selection for the user (e.g. query as header and order information), only the elements under EDI_INFO are used.
201	Carcase combination	K	If there is only one key for several carcase colours - inside/outside. If variant type 201 is filled, it overrides variant types 200, 202-205 to the extent that they are only for internal use by the software companies (e.g. internal colour control). These are then set as invisible header variants ("U") in the element HEADER_POS_VARIATION_TYPE. For external use as a selection for the user (e.g. query as header and order information), only the elements under EDI_INFO are used.
303	Combination of handles	K	If only one key is given for several handle conditions. If variant type 303 is filled, it overrides variant types 300-302 to the extent that they are only for internal use by the software companies (e.g. internal colour control). These are then set as invisible header variants ("U") in the element HEADER_POS_VARIATION_TYPE. For external use as a selection for the user (e.g. query as header and order information), only the elements under EDI_INFO are used.
999	Fixed material assignment	F	This variant type is defined globally, but may only be used in the material properties.

It was noticed that the descriptions of 4 variant types were only partially transferred during the transformation of the CSV into the XML documentation. Missing information has now been added again.

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.



4.2. C Explanatory example for multi-language catalogues (e.g. Switzerland) on the elements COUNTRY\_ID and ISO\_LANGUAGE\_ID

Decision : 2022-03-22

ISA\_LANGUAGE\_ID:

2.8.1

The language code defines the language in which the texts in the data pool were created. The ISO language code must be used.

Multiple language codes indicate that the catalogue is used in multiple languages. The catalogue texts must only use the language codes entered in these elements.

3.0.0

The language key defines the language in which the texts of the dataset were created. The ISO language key is to be used for this purpose.

In the case of multilingual catalogues, all languages occurring in the catalogue must be entered here. Only the language keys entered in these elements may be used for the texts of the catalogue.

For example, in a Swiss catalogue, the ISO\_LANGUAGE\_ID's DE (German), FR (French) and IT (Italian) can be stored here.

COUNTRY\_ID:

2.8.1

This element is used to specify the correct country allocation where a language is spoken in multiple countries and the identical currency is specified. This requires entry of the ISO country code.

3.0.0

The ISO country code of the target market is entered in this element.

For countries with several languages but the same currency, make sure to use the correct ISO country code.

For example, a Swiss catalogue may contain the 3 ISO\_LANGUAGE\_ID's DE (German), FR (French) and IT (Italian). However, the COUNTRY\_ID must be indicated as CH (Switzerland).

To avoid misleading information in the COUNTRY\_ID and ISO\_LANGUAGE\_ID elements, an explanatory example has been added to the description of both elements.



2.8.1

This element defines the features of a feature class. All features of the complete database are defined in this element. In case of a finish option (cf. table standard variations, flag "A") neither texture (TEXTUR) nor colour values (COLOR) are permitted. The assignment of brightness and transparency is optional. In case they are defined they have priority over the colour variations.

3.0.0

In this element, the options for a feature are defined and all options to be modified in the dataset are stored. If these are special colours, separate OPTIONS must be created for each colour scheme (e.g. RAL, NCS, etc...). (see also: Best Practices/Special Colours)

In order to avoid incorrect handling of special colours in the future, the description in the OPTION element has been supplemented and a best practice has been added to the documentation.

Excerpt Best Practice:

### Special colours

Not all possible spot colours can be created individually in an IDM catalogue.

Therefore, it is necessary to define a structure that enables the planning systems to show the applicable special colours.

For this purpose, an option is to be created for each standardised colour scheme, e.g. RAL, NCS, SIKKENS, etc.

Example 1:

The supplier offers a desired colour in the RAL and NCS colour schemes for various design elements.

**Correct:**

```

<OPTION FEATURE_NO="101" OPTION_KEY="NCS">
  <OPTION_TEXT>
    <LANGUAGE ISO_LANGUAGE_ID="DE">
      <TEXT>
        NCS-Lackierung
    </TEXT>
  </LANGUAGE>
  <OPTION_COMMENT_TEXT>
  <MINIMUM_DELIVERY_TIME>
</OPTION_TEXT>
</OPTION>
<OPTION FEATURE_NO="101" OPTION_KEY="RAL">
  <OPTION_TEXT>
    <LANGUAGE ISO_LANGUAGE_ID="DE">
      <TEXT>
        RAL-Lackierung
    </TEXT>
  </LANGUAGE>
  <OPTION_COMMENT_TEXT>
  <MINIMUM_DELIVERY_TIME>
</OPTION_TEXT>
</OPTION>

```

**Wrong:**

```

<OPTION FEATURE_NO="501" OPTION_KEY="MSLS">
  <OPTION_TEXT>
    <LANGUAGE ISO_LANGUAGE_ID="DE">
      <TEXT>
        RAL/NCS-Farben
    </TEXT>
  </LANGUAGE>
</OPTION_TEXT>
</OPTION>

```

[...]



## Contact

### Daten Competence Center e. V.

Goebenstraße 4-10

32052 Herford

Info: [www.dcc-moebel.org](http://www.dcc-moebel.org)

Dr.-Ing. Olaf Plümer

E-mail: [pluemer@dcc-moebel.org](mailto:pluemer@dcc-moebel.org)

Phone: +49 52 21 / 12 65-37

Anika Degenhard

E-mail: [degenhard@dcc-moebel.org](mailto:degenhard@dcc-moebel.org)

Phone: +49 52 21 / 12 65 - 38