



Instruktion

Datum	Diarienummer	Ärendetyp
2022-06-20	16FMV10512-5:3	3.1
	Dokumentnummer	Sida
	3.0.0	1(9)
Giltig t.o.m.	Upphäver	
t.v.	16FMV10512-5:2	

Beslutande
Gustaf Fahl
Teknisk Direktör

Föredragande
Jörgen Wremert
LogV Mdata TI

FMV S1000D IPD to PMF converter

Table of contents

	Page
References	2
1 Introduction	2
1.1 S1000D spare part catalogs	2
1.2 Reuse of an IPD data module structure.....	2
2 The FMV IPD to PMF converter.....	2
2.1 Aim	2
2.2 Functional scope.....	3
2.3 Converter components.....	3
2.4 Installation	3
3 How to execute a conversion.....	4
3.1 Invoking the converter.....	4
3.2 Validation of output before delivery to FMV	5
4 Requirements on S1000D IPD input.....	5
4.1 File based transfer package content.....	5
4.2 The IPD publication module.....	5
4.3 The IPD data module	6
4.4 The IPD illustration.....	8
4.5 CAGE code mapping	8
4.5.1 Using the remarks element.....	8
4.5.2 Using a separate mapping file	9

References

Table 1 References

Document No./Reference	Title
[1] Logistikportalen	Regelverk FMV Materielpublikationer (http://logistikportalen.fmv.se/): http://logistikportalen.fmv.se/tjansterprodukter/materielpublikationer/materielforteckningar/Sidor/Startsidemall.aspx
[2] S1000D Issue 4.1	International specification for technical publications (applicable issue below referred to as "S1000D")
[3] 16FMV10512-3:n ¹	FMV General rules for application of S1000D

1 Introduction

1.1 S1000D spare part catalogs

S1000D provides a data module type specifically furnished to contain spare parts information. A spare part catalog is usually composed of several catalog data modules where each data module consists of an illustration and the corresponding list of items.

A S1000D spare parts data module is referred to as an IPD data module, where IPD stands for Illustrated Parts Data.

A multi-figure catalog, e.g. covering an entire system, is achieved by use of the S1000D publication module which can contain a hierarchical structure referring to the various data modules. In principle this catalog is very similar to a normal "FMV/FM RDK".

1.2 Reuse of an IPD data module structure

Initially the IPD data module was specified to contain a subset of the data stored in an S2000M database. This is no longer the case, the IPD data module has been "neutralized" to also reflect environments that do not apply S2000M, but rather rely on completely different practices and tools.

Nevertheless, in practice the S1000D IPD concept differs a lot from the catalog structure that FMV/FM use to apply, that is the PMF based catalogs. This means that any useful information in an IPD data module needs to be extracted and converted to a corresponding PMF structure to be possible to import and make use of in the FMV/FM environment.

2 The FMV IPD to PMF converter

2.1 Aim

The converter is developed to facilitate creation of spare parts catalog information in accordance with PMF catalog structures from S1000D based catalogs without compromising a producer's S1000D environment used to create S1000D conformant IPD and publication module based catalogs.

The converter can be obtained from FMV Kluster 508 upon request to verksamhetsstod@fmv.se.

¹ Reference number :n in the reference list relates to the latest version according to "Beslut FMV Business Rules för S1000D", 16FMV10512-1:n.



Instruktion

Datum	Diarienummer	Ärendetyp
2022-06-20	16FMV10512-5:3	3.1
Dokumentnummer		Sida
3.0.0		3 (9)

2.2 Functional scope

The converter takes as input an S1000D spare parts catalog consisting of a publication module referring to a number of IPD data modules. Output is the corresponding PMF catalog structure.

The converter has been tested for IPD catalogs of S1000D Issues 2.3, 4.0/4.0.1 and 4.1. Since these IPD structures coincide with all issues from Issue 2.0 to 4.1, in the respects in focus, they should also be ok. Earlier S1000D Issues are not supported.

A presumption is that all information subject of conversion must be in XML. The converter does not support SGML.

It is also assumed that the input data is valid XML in accordance with the S1000D issue concerned. The converter will not parse any input data but is likely to simply end execution if such errors are encountered.

NOTE

It is highly recommended that the data is parsed against the S1000D issue 4.1 IPD XML-Schema and if possible BREX checked before conversion is taking place.

The conversion only includes the S1000D catalog structure information, such as this is represented by an S1000D publication module and the referenced IPD data modules. Details about the catalog items are not extracted and reused.

The converter requires Java to be installed.

2.3 Converter components

The converter consists of five components

NOTE

XXX-XX is the version indicator.

- An xslt script, IPD2PMF_XXX-XX.xsl, which reads the IPD structure input and generates the output (the script included in this package)
- A java application, xsltgui_XXX-XX.jar, that provides a simple GUI interface to support invocation of the xslt script (and xsl processor), that is Java has to be installed on the processing computer (the java application included in this package)
- An xsl processor MS XSL, msxsl.exe, downloadable from the web (not included in this package)
- A file, cage2fkod.xml, defining the mapping of CAGE codes to PMF “company codes” (incl in the package, see below)
- XML Schema, S1000D_to_PMF_R_vXX.xsd, for validation of output before delivery to FMV. (Where XX is the version indicator.)

2.4 Installation

The package comes without any specific installation procedure. However, installation is fairly straight-forward.

Thus, to install the converter do the following:

1. Create an installation folder where the checking tool(s) will be kept, e.g. c:\ipd2pmf.
2. Unzip the file “FMV-S1000DIPD2PMF-tool_XXX-XX.zip” into the installation folder. Make sure the following needed files are now present in the folder:
 - xsltgui_XXX-XX.jar
 - IPD2PMF_XXX-XX.xsl
 - cage2fkod.xml (see section 4.5 below for description of use)

3. Install an XML parser, for example:
 - msxsl.exe, which can be downloaded free from Microsoft official website

3 How to execute a conversion

3.1 Invoking the converter

The converter runs in one pass.

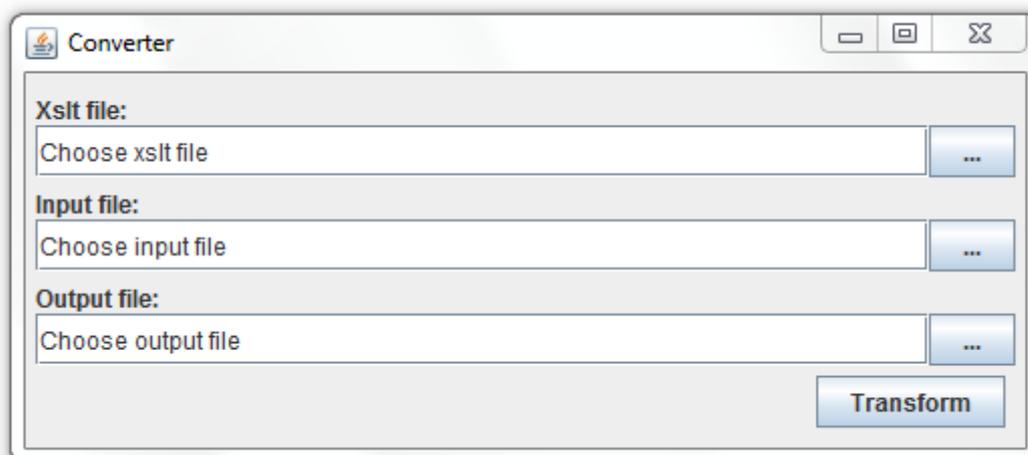
Apart from the application files listed above, make sure that the following are available in the folder from where the converter is being run, that is the converter installation folder:

- The publication module representing the spare parts catalog subject of conversion.
- The data modules (from XML schema ipd.xsd, <dmModule> containing <dmCode infoCode="941" ... />) referenced by that publication module.
- The code mapping file, cage2fkod.xml (if needed updated as described below).

The requirements on the input files are described in section 4 below.

Provided that Java is properly installed, the converter is started by double-clicking the java jar file [xsltgui_XXX-XX.jar](#).

The following pop-up window will show:



Provide the input asked for by filling in, or browsing for, file names/paths:

- The xslt script file name [IPD2PMF_XXX-XX.xsl](#).
- The input file name, that is the S1000D publication module representing the IPD based catalog subject of conversion.

NOTE

IPD data modules and the publication module must be located in the folder where the converter is executed (see below for details on format).

- The output file name of the resulting PMF file.

3.2 Validation of output before delivery to FMV

To validate the output, before delivery to FMV, according to the XML Schema (S1000D_to_PMF_R_v01.xsd) perform these steps:

1. Add a path to the XML schema.

Example:

```
<ImportKatalog
```

```
  xsi:noNamespaceSchemaLocation="S1000D_to_PMF_R_v02.xsd"
```

```
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">.
```

This may be done by changing the variable "noNamespaceSchemaLocation" from "No" to "Yes" in;

```
<xsl:variable name="noNamespaceSchemaLocation">Yes</xsl:variable>
```

in "IPD2PMF_XXX-XX.xsl". The variable "path_name_pm" handle the path and file name of the XML Schema.

2. Open the output file in a XML Editor (like XML Spy, XMetaL or ArborText Editor).
3. Perform a validation.
4. Correct any errors (in the PM or in the DM's [IPD]) and validate again.
5. Remove any folder path to the XML schema in the element <ImportKatalog> before delivery. It's only allowed with the correct file name in the attribute

`xsi:noNamespaceSchemaLocation`.

Note: The attribute `xsi:noNamespaceSchemaLocation` does not have to be included in the delivery.

4 Requirements on S1000D IPD input

All these requirements below are also included in the FMV general BREQ.

4.1 File based transfer package content

The converter assumes that the catalog to convert exists as an S1000D transfer package, including a complete DDN delivery list.

Apart from the DDN object, the transfer package must contain the following objects:

- One publication module (PM), representing the structure of the catalog.
- The IPD data modules (DM) referred by the PM.
- The illustrations/information objects referred to by the various DM.

4.2 The IPD publication module

The IPD publication module, constituting the overall structure, must be in accordance with the following description:

- The IPD publication module can consist of one or more sections, each section represented by a <pmEntry>.
- Sections must have a title using element <pmEntryTitle> in the IPD publication module entry <pmEntry> element.
- Each section is comprised of one or more IPD figures, each of these represented by an IPD data module, which is referenced from within the <pmEntry> of the IPD publication module by use of a <dmRef>.
- Nesting of <pmEntry> must not be used.



Instruktion

Datum	Diarienummer	Ärendetyp
2022-06-20	16FMV10512-5:3	3.1
Dokumentnummer	Sida	
3.0.0	6 (9)	

- The IPD publication module must contain a `<remarks>/<simplePara>` element storing the "M/F-number" (aka FBET "Förrådsbeteckning") of the FMV/FM RDK (refer to FMV project POC for advice). The remarks element must contain the following:
`<simplePara>FBET:Fxxxx-xxxxxx</simplePara>` or `<simplePara>FBET:Mxxxx-xxxxxx</simplePara>`
where xxxx-xxxxxx is the M/F-number assigned to the FMV/FM RDK.

4.3 The IPD data module

The IPD data modules, each representing a catalog figure (that is one illustration accompanied by the corresponding catalog items), must be in accordance with the following description:

- Each IPD data module includes an illustration and one or more corresponding CSN entries, where the illustration includes a title that will be used as the title of the catalog figure.
- If there isn't a drawing in the S1000D source a "blank drawing" (ICN-A196N-178640-001-01.CAL) can be obtained from FMV.
The PMF element `<Bildidentitet>` will be populated with the information from the IPD data module code attribute `disassyCode`.
- Over all applicability information for each IPD data module is populated in the status information for the data module. As decided in FMV General Rules [3], for applicability, BRDP 02-20-00-000 states that the human readable annotations must be provided.
The element `<displayText>/<simplePara>` must be used and to include the human readable IPD data module applicability annotations.
- S1000D can contain more than one element `<figure>` and more than one child element `<graphic>` to element `<figure>`.
- Each CSN (`<catalogSeqNumber>`) entry specifies:
 - item/call-out number using attribute "item".
 - indenture using attribute "indenture".
- Each CSN entry includes one or more ISN (`<itemSeqNumber>`) entries.
 - When the CSN only has one ISN entry the attribute "itemSeqNumberValue" must be set to "00A".
 - When the CSN requires more than one ISN entry the following rules applies:
 - The attribute "itemSeqNumberValue" will be used for identification.
 - The one character of three that will be converted to PMF will be the second character for the attribute `itemSeqNumberValue`.

Note:
The attribute value for "itemSeqNumberValue" according S1000D must be three characters. PMF can only have one character to indicate item sequence numbers.

 - ISN (`<itemSeqNumber>`) example (the converted character shows in bold):
 - `<itemSeqNumber itemSeqNumberValue="03A">`
 - `<itemSeqNumber itemSeqNumberValue="07A">`
 - `<itemSeqNumber itemSeqNumberValue="0AA">`
- Each ISN entry includes data about the item concerned, where each ISN entry specifies:
 - The attributes "manufacturerCodeValue" and "partNumberValue" in the sub-element `<partRef>` must be used to uniquely identify the item.
 - There are two different structures to identify the FMV unique FBET.
 - By using the "manufacturerCodeValue" and "partNumberValue" in the subelement `<partRef>`.
If this structure is used the value of attribute "manufacturerCodeValue" must be the FMV CAGE code, "A196N". (For older FBET, CAGE codes "S4418" or "SZ003" will also be valid.)
 - Or, by using the element `<customerStockNumber>`
 - If the ISN isn't a valid FMV FBET the following is required in the subelement `<partRef>`:
 - By using the attribute "manufacturerCodeValue"; with the CAGE code (not "A196N", "S4418" or "SZ003").



Instruktion

Datum	Diarienummer	Ärendetyp
2022-06-20	16FMV10512-5:3	3.1
Dokumentnummer		Sida
3.0.0		7 (9)

- And the attribute `partNumberValue`: with the "Ritnr" (drawing number/producers reference number/artikelnummer). With a maximum of 27 characters.

Note:

The element `<customerStockNumber>` FBET will take presence over the element `<partRef>` FBET if both element have specified a FBET.

Note:

Preferable structure for the FBET is to use the element `<customerStockNumber>`.

Note:

This element combination will be the reference to the unique part as stored in the FMV databases (PMF)! If incorrect, FMV catalog generator will not be able to resolve the link, thus not able to populate the item correctly in the generated catalog.

- Item description/name using element `<descrForPart>` (in substructure `<partSegment>/<identItemData>`).
If the item description is more than 19 characters (including white space, tab, etc), the element `<shortName>` must be used. The definition in the element `<shortName>` must not be more than 19 characters. If the element `<shortName>` exists this element will be used.
- Quantity using element `<quantityPerNextHigherAssembly>` is used to give the actual quantity for the item. To identify the quantity "As Required" (erfordrigt antal), the characters "AR" or "REF" must be used.
- When an IPD data module are updated, an item can be change or new. When using change indication for updates, the following rules applies:
 - The attribute "issueType" for sub element `<dmStatus>` must be set to "changed"
 - To indicate a change item, the attribute "changeMark" must be set to "1" and the attribute "changeType" must be set to "modify" in the sub element `<catalogSeqNumber>`
 - To indicate a new item, the attribute "changeMark" must be set to "1" and the attribute "changeType" must be set to "add" in the sub element `<catalogSeqNumber>`
 - The value "delete" for the attribute "changeType" must not be used.
- PMF have the possibility to populate additional part information. The additional information can be populated in four additional rows in the PMF layout. To populate the additional item information for PMF (this is used for "Övriga kompletterande uppgifter" (Other supplementary information) in the resulting PMF file) the following rules applies:
 - The element `<genericPartData>` (parent `<genericPartDataGroup>`) must be used and for the attribute `genericPartDataName` the following rules applies.
 - To populate additional row 1 the attribute `genericPartDataName` value must be "Rad03".
 - To populate additional row 2 the attribute `genericPartDataName` value must be "Rad04".
 - To populate additional row 3 the attribute `genericPartDataName` value must be "Rad05".
 - To populate additional row 4 the attribute `genericPartDataName` value must be "Rad06".
 - The child element `<genericPartDataValue>` must contain a minimum of 1 character and a maximum of 40 characters .
 - The `<genericPartData>` can only be used if the ISN isn't a valid FMV FBET or for additional technical info that is not inducted in GoF REG (the FMV system for FBET).
- The DM must be named with the following syntax:
 - DMC-modellIdentCode-systemDiffCode-systemCode-subSystemCode+subSubSystemCode-assyCode-disassyCode+disassyCodeVariant-infoCode+infoCodeVariant-itemLocationCode_issueNumber-inWork_languagelsoCode-countryIsoCode.xml
 - Example; DMC-TRACKFIRE-B-63-70-TGF1-00A-941A-A_001-00_SV-SE.xml

4.4 The IPD illustration

The figures used in the IPD data modules must be in accordance with the following description:

- A figure must be in a graphical format that has been agreed with FMV (e.g. refer to FMV General and/or any applicable project specific BR).
- If nothing else has been agreed this is how illustrations should be:
 - In PMF, it is preferable to import drawings in CALS compression type CCITT Group 4.
 - Valid graphic formats are:
 - tif/tiff (This format is difficult to import, in PMF, as there are many ways to create TIF files. If possible, use "png", "svg" or "jpg/jpeg".)
 - cal
 - png
 - jpg
 - jpeg
 - svg (width="500px" height="653px" viewBox="0 0 500 653")
 - If a line in SVG is set to be less than one (1) pixel (for example stroke-width = "0.25") then that line will not be displayed in PMF. However, there is only a cosmetic limitation in the PMF itself as these lines are shown correctly in the preview and in the production of a PDF.
 - If you use text elements in SVG, it is recommended to use the font "Helvetica". There are problems with certain fonts that prevent SVG from appearing in PMF at all, examples of such font are "Arial" and "ArialMT". The selected font should be checked in the SVG file as some programs can change the font when creating SVG. Text elements can also be converted to outlines, which means they no longer need to rely on fonts.
 - The size shall be 245 mm in height and 176 mm in width.
 - The resolution shall be 300 dpi.
 - There is a possibility to add the correct file format (suffix) with "IPD2PMFXXX-XX.xml" (e.g. The normal file format *.CGM from S1000D is not allowed as input format to PMF.). In that case change this in "IPD2PMF_XXX-XX.xml" before running the converter:
 - Change <xsl:variable name="change_graphic">No</xsl:variable> from "No" to "Yes".
 - Add the correct file format in the variable "output_graphic_cgm" (<xsl:variable name="output_graphic_cgm">TIF</xsl:variable>). In this case the value is set to "TIF". Change this, if necessary, to the correct file format.

4.5 CAGE code mapping

FMV's systems do not always use CAGE codes, as used in S1000D, for identification of part manufacturer/supplier but employs the specific corresponding PMF codes (FKOD's/firmakoder) for the identification purpose. This implies that in conversion from IPD data modules the CAGE codes in the data modules must be mapped into FKOD's in the resulting PMF files. The mapping information must be provided in <remarks> elements in the IPD data modules or in a separate file.

4.5.1 Using the remarks element

When mapping of manufacturer codes is provided in <remarks> elements in the identification and status section of the IPD data modules concerned, the <remarks> elements must contain two sub elements in accordance with the following:

```
<simplePara>FKODER:</simplePara>
<simplePara>CAGE1=FKA|CAGE2=FKB|CAGEN=FKN</simplePara>
```

where CAGE1...CAGEN means a manufacturer CAGE code and FKA ... FKN means a "firmabokstavskod" iaw PMF principles.

Note:

In this case, all CAGE codes used in an IPD data module concerned must be listed as above!

To get the valid values of "firmabokstavskod" contact FMV (refer to FMV project POC or verksamhetsstod@fmv.se for advice).



Instruktion

Datum	Diarienummer	Ärendetyp
2022-06-20	16FMV10512-5:3	3.1
Dokumentnummer		Sida
3.0.0		9 (9)

4.5.2 Using a separate mapping file

When mapping of manufacturer is provided in a separate mapping file the file must fulfill the following requirements:

The file must be named `cage2fkod.xml` and during conversion it must reside in the same folder as the application and the publication/data modules constituting the catalog, that is the modules being subject of conversion.

Note:

This file cannot be an integral part of a conformant S1000D transfer package but can be delivered in connection to a compliant transfer.

The overall structure of the file must be:

```
<?xml version="1.0" encoding="utf-8"?>
<param>
<fkoder>CAGE1=FKA|CAGE2=FKB|CAGEN=FKN</fkoder>
</param>
```

where CAGE1...CAGEN means a manufacturer CAGE code and FKA ... FKN means a "firmabokstavskod" in accordance with PMF specifications (refer to FMV project POC for advice).