USER'S MANUAL LSPB

CHANGE HISTORY

REV #	DATE	REASON OF MODIFICATION
1	01/08/2014	FIRST ISSUE - LSPB version 2.0 beta
2	11/08/2015	LSPB version 2.10 beta See change bars
3	23/03/2016	LSBP version 3.0 beta See change bars Loadable software transport media generation is an option (see ref [3] §3). The basic is to generate the loadable software parts (see ref [3] §2).

TABLE OF CONTENTS

1. INTRODUCTION

This document is the user's manual of the tool called LSPB.
LSPB is used to build ARINC 665 loads compatible with reports described in ref [1], [2] and [3].
LSPB run under JAVA and then necessitate the installation of the JRE. It has been tested with the following JRE version:
- 1.6.0 24

LSPB is provided as is, even if it has been tested intensively in order to check compliance with all ARINC reports. The tool supplier shall not be liable for the consequences of use of files produced by the tool.

2. REFERENCED DOCUMENTS

[1]   LOADABLE SOFTWARE STANDARDS ARINC REPORT 665-1
[2]   LOADABLE SOFTWARE STANDARDS ARINC REPORT 665-2
[3]   LOADABLE SOFTWARE STANDARDS ARINC REPORT 665-3

3. ABBREVIATIONS

ASCII   American Standard Code for Information Interchange
CRC   Cyclic Redundancy Code
DTD   Document Type Definition
GUI   Graphical User Interface
JRE   Java Runtime Environment
LSPB   Loadable Software Part Builder
XML   eXtended Markup Language

4. DEFINITIONS AND NOTATIONS

4.1   DEFINITIONS

Character refers to: any ASCII character from code 32_{10} to 127_{10}.
Letter refers to: from 'A' to 'Z' or 'a' to 'z'.
Number refers to: from '0' to '9'
Alphanumeric refers to: letter or number

4.2   NOTATIONS

X_{10} means that the number X is written in decimal base.

5. USAGE

LSPB can be used in command line or with a GUI.
The command line is commonly used in the executable automatic generation process, like makefile. Options are used to tune the generation process.
GUI is used, from time to time, when the load is only to be delivered to your customer.

5.1   COMMAND LINE

java –jar LSPB.jar <XML configuration file>

You may have to add the path before "java" in case of directory containing java.exe is not in the PATH variable.

In this mode, the tool runs automatically taking directives from the configuration file. It will generate the ARINC665 load with a minimum set of files, according to the load content and format (see ref [3] §2.2 and §3.2):

- Header file : file with an extension LUH (see ref [3] §2.2.3.1)
- Data files : files with an extension LUP (see ref [3] §2.2.3.2) in the sub directory whose name is given by the tag DIRECTORY of the configuration file

Optionally, batch file can be generated with the load (see ref [3], §2.3.1), using the tag BATCH of the configuration file.
Optionally, List-of-Files file (see ref [3] §3.2.3.2), List-of-Loads file (see ref [3] §3.2.3.1) and List-of-Batch file (see ref [3], §3.2.3.3) can be generated with the media (see ref [3], §3), using the tag MEDIA of the configuration file.

5.2   GUI

5.2.1   Main window

By launching the command "java –jar LSPB.jar", the main LSPB window appears:
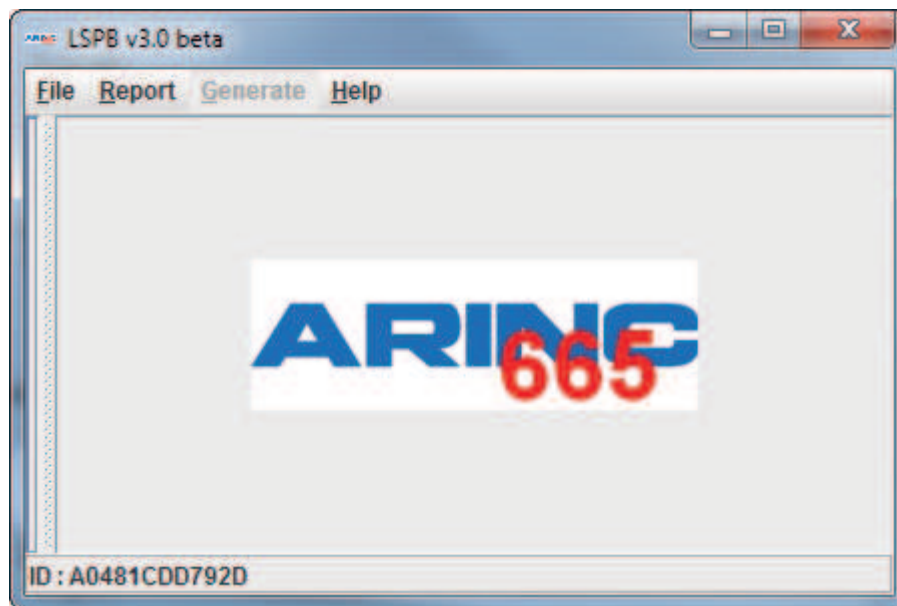
The first thing to do will be to open a configuration file. To do so, select the *File->Open* menu to choose the configuration file to open.

The Report menu allows selecting manually the version of ARINC665 to use to build the load. If you select it after having loaded a configuration file, it surpasses the version indicated in the configuration file.

Then, click on the menu Generate to generate the ARINC665 load.
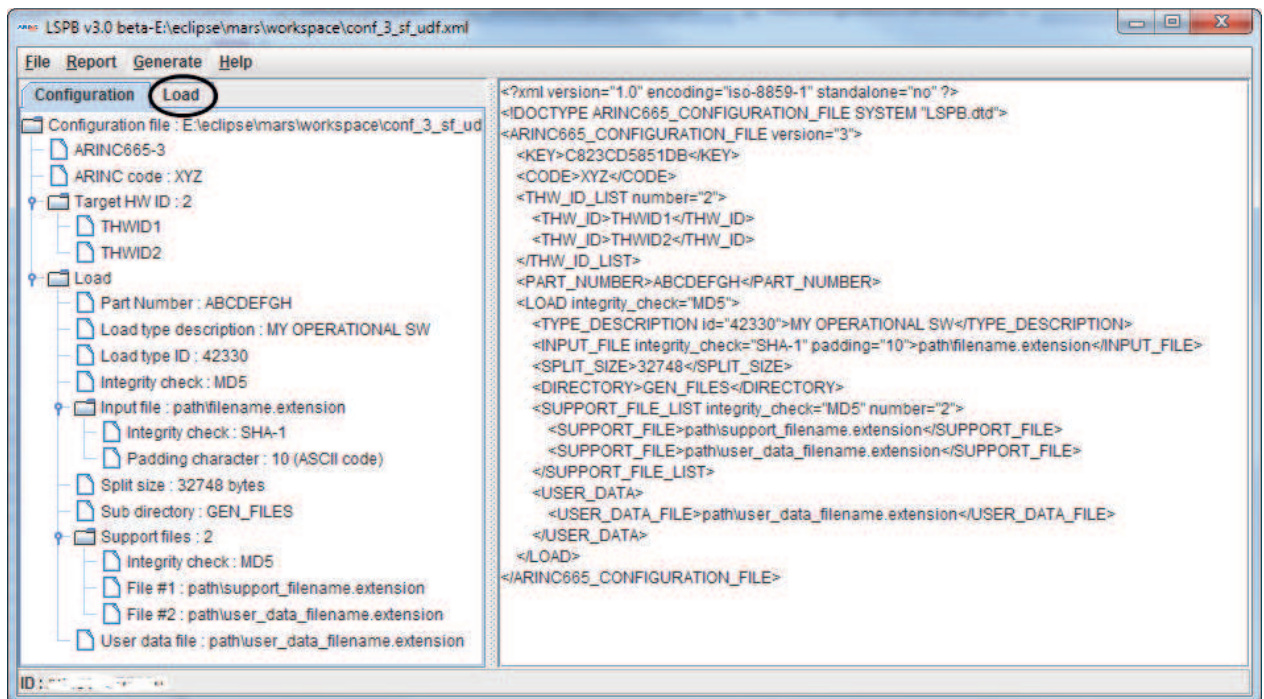
Clicking on the Load tab, you can display detailed information of the load.
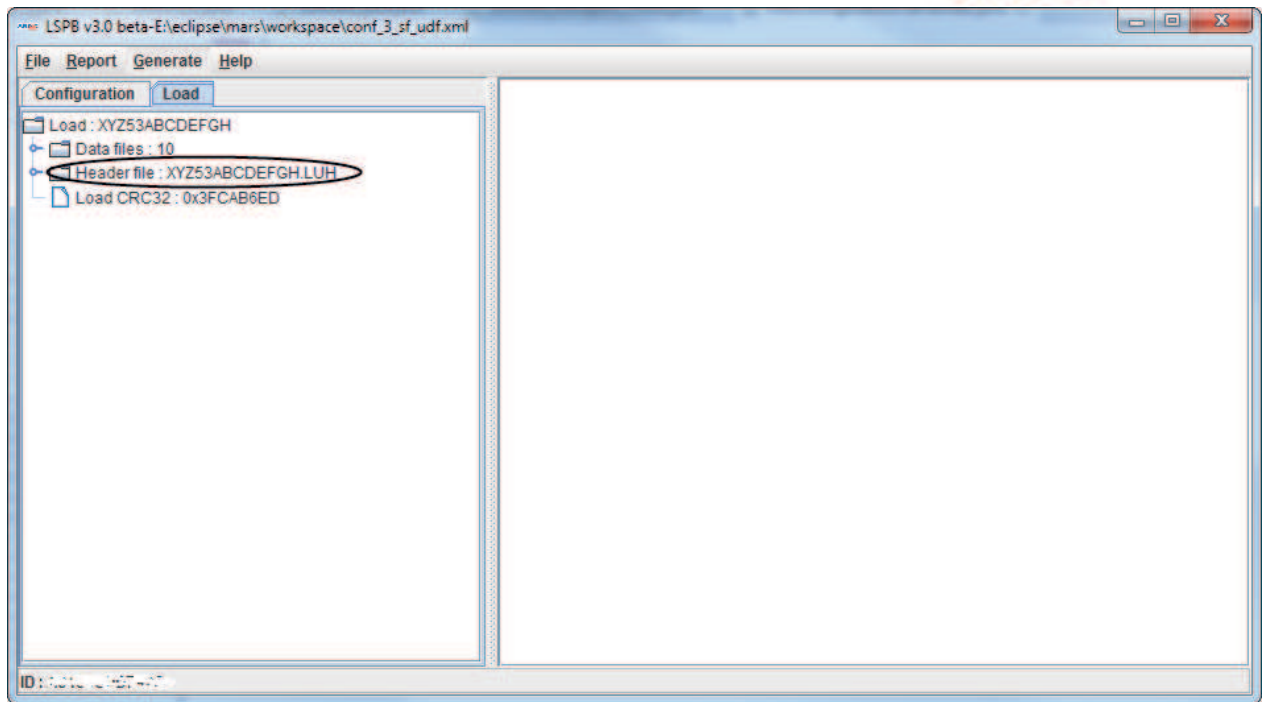
Clicking on the header file, you can display the hexadecimal dump of this file.
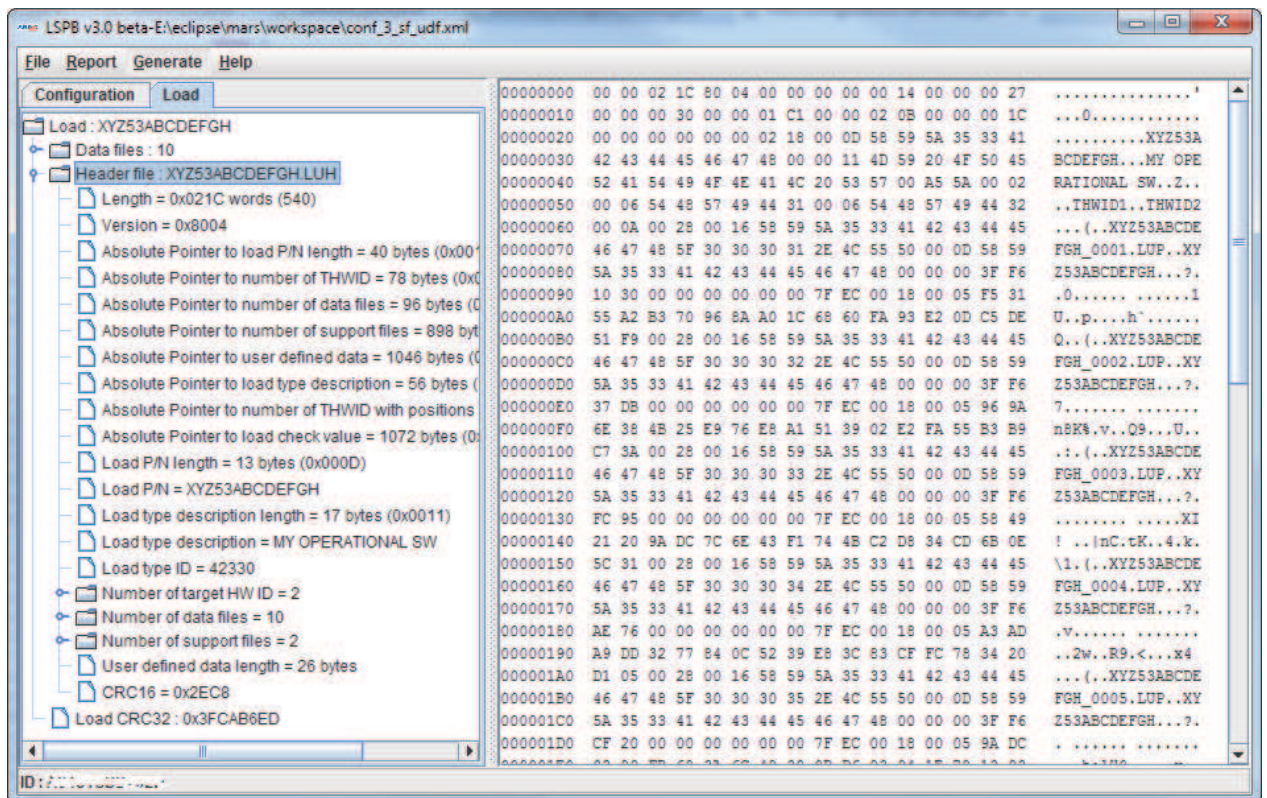
You can also display the hexadecimal dump of data files, List-of-Loads and List-of-files files by clicking on each of them in the tree.
Note: for the data files, the latency to display it could be several seconds, depending on the size of the data file.

Finally, you find all the generated files (header file and data files) in the requested directory (as per configuration file).

```
XYZ53ABCDEFGH.LUH
XYZ53ABCDEFGH_0001.LUP
XYZ53ABCDEFGH_0002.LUP
XYZ53ABCDEFGH_0003.LUP
XYZ53ABCDEFGH_0004.LUP
XYZ53ABCDEFGH_0005.LUP
XYZ53ABCDEFGH_0006.LUP
XYZ53ABCDEFGH_0007.LUP
XYZ53ABCDEFGH_0008.LUP
XYZ53ABCDEFGH_0009.LUP
XYZ53ABCDEFGH_0010.LUP
```

5.3   CONFIGURATION FILE

The configuration file is an XML file. This file shall be valid vs the following DTD:

LSPB.dtd

```xml
<?xml version="1.0" encoding="iso-8859-1" standalone="no" ?>
<!DOCTYPE ARINC665_CONFIGURATION_FILE SYSTEM "LSPB.dtd">
<ARINC665_CONFIGURATION_FILE version="X">
  <KEY>X</KEY>
  <CODE>X</CODE>
  <THW_ID_LIST number="X">
    <THW_ID>X</THW_ID>
  </THW_ID_LIST>
  <PART_NUMBER>X</PART_NUMBER>
  <LOAD integrity_check="X">
    <BATCH>X</BATCH>
    <TYPE_DESCRIPTION id="X">X</TYPE_DESCRIPTION>
    <INPUT_FILE integrity_check="X" padding="X">X</INPUT_FILE>
    <SPLIT_SIZE>X</SPLIT_SIZE>
    <DIRECTORY>X</DIRECTORY>
    <SUPPORT_FILE_LIST integrity_check="X" number="X">
      <SUPPORT_FILE>X</SUPPORT_FILE>
    </SUPPORT_FILE_LIST>
    <USER_DATA>
      <USER_DATA_FILE>X</USER_DATA_FILE>
      <USER_DATA_TEXT>X</USER_DATA_TEXT>
    </USER_DATA>
  </LOAD>
  <MEDIA>
  </MEDIA>
</ARINC665_CONFIGURATION_FILE>
```

All the tags shall be written in uppercase. All the attributes shall be written in lowercase and enclosed with double quotes.

Level	Tag / Attributes	Attributes	References
1	ARINC665_CONFIGURATION_FILE	version	
2	KEY		
2	CODE		See ref [3] §2.1.2
2	THW_ID_LIST	number	
3	THW_ID		See ref [3] §1.5, §2.2.3.1.22
2	PART_NUMBER		See ref [3] §2.1
2	LOAD	integrity_check	See ref [3] §5
3	BATCH		See ref [3] §2.3.1, 3.2.3.3
3	TYPE_DESCRIPTION	id	
3	INPUT_FILE	integrity_check, padding	
3	SPLIT_SIZE		
4	DIRECTORY		
3	SUPPORT_FILE_LIST	integrity_check, number	
4	SUPPORT_FILE		See ref [3] §2.2.3.1.44
3	USER_DATA		See ref [3] §2.2.3.1.57
4	USER_DATA_FILE		
4	USER_DATA_TEXT		
4	USER_DATA_BCC		
2	MEDIA		

5.3.1   Tag ARINC665_CONFIGURATION_FILE

This is the main tag of the configuration file.
The content shall be the tags CODE, LOAD, INPUT_FILE, SPLIT_SIZE, DIRECTORY and THW_ID_LIST.
This tag is mandatory. If omitted, LSPB will generate an exception.
The attribute version is used to indicate the report version. The possible values are 1_{10}, 2_{10} or 3_{10}.
This attribute is not mandatory. If omitted, version is set to 2_{10}.

Example:
<ARINC665_CONFIGURATION_FILE
version="1">…</ARINC665_CONFIGURATION_FILE>

5.3.2   Tag KEY

This tag is used to check if LSPB is authorized to run on the computer.
The content shall be a list of alphanumeric. It will be provided by LSPB manufacturer and is based on the ID displayed at the bottom of the main LSPB window.
This tag is not mandatory. If omitted or in case of invalid content, LSPB will not generate the load CRC32.

Example:
<KEY>A1B2C3D4E5F6G7H8</KEY>

5.3.3   Tag BATCH

This tag is used to generate batch file (see ref [3] §2.3.1).
The content shall be a list of characters. The following values (no case sensitive) are accepted:
-   NO : no files will be generated

This document is the property of Assistance Informatique Toulouse and must not be disclosed nor reproduced without express authorization
mailto : assistanceinformatiquetoulouse@gmail.com

Page 10

- YES : batch file will be generated with COMMENT field empty (see ref [3] 2.3.1.9 and 2.3.1.10)
- Any other : batch file will be generated with COMMENT field filled with this list of characters (see ref [3] 2.3.1.9 and 2.3.1.10)

This tag is not mandatory. If omitted, no file will be generated.

Examples:
<BATCH>NO</BATCH>
<BATCH>YES</BATCH>
<BATCH>ONE COMMENT</BATCH>

5.3.4 Tag CODE

This tag is used to indicate the supplier code MMM.
The content shall be a list of 3 letters.
This tag is mandatory. If omitted, LSPB will raise an exception.

Example:
<CODE>AIF</CODE>

5.3.5 Tag LOAD

This tag is used to include the part number.
The content shall be the tags PART_NUMBER and TYPE_DESCRIPTION.
This tag is mandatory.

The attribute integrity_check is used to indicate if load check value (see ref [3] § 2.2.3.1.59, 2.2.3.1.60 and 2.2.3.1.61) has to be included into the header file (see ref [3] §2.2.3.1) and which integrity check algorithm (see ref [3] §5) has to be used. The possible values are:
- MD5
- SHA-1
- Any other : no integrity check

The attribute is not mandatory. If omitted, the load check value length field is set to 0_{10} in the header file.

Example:
<LOAD integrity_check="MD5">…</LOAD>

5.3.6 Tag PART_NUMBER

This tag is used to indicate the load part number and media set part number without the MMM and CC (see ref [3] §2.1.3) parts. LSPB will add the MMM using the content of tag CODE and then calculate and add the CRC8.
The content shall be a list of alphanumeric. There is no check vs load part number format (see ref [3] §2.1.1).
This tag is mandatory. If omitted, LSPB will raise an exception.

Example:
<PART_NUMBER>12345678</PART_NUMBER>

5.3.7 Tag TYPE_DESCRIPTION

This tag is used to indicate the load type description (see ref [3] § 2.2.3.1.16, 2.2.3.1.17 and 2.2.3.1.18).

The content shall be a list of letters.

This tag is not mandatory. If omitted, the type description fields are not added to the header file.

The attribute id is used to indicate the identifier (see ref [3] §2.2.3.1.18).

The content shall be a list of numbers. The possible values are from 0_{10} to 65535_{10}.

The attribute is not mandatory. If omitted, identifier is set to 0_{10}.

Example:
<TYPE_DESCRIPTION id="42330">A DESCRIPTION<TYPE_DESCRIPTION>

5.3.8   Tag INPUT_FILE

This tag is used to indicate the path and the name to access the input file. It can be an absolute path or a relative path. In case of relative path, the path where LSPB is invoked is added to the relative path.

The content shall be a list of characters. If the file is not readable, LSPB will generate an exception.

This tag is mandatory. If omitted, LSPB will raise an exception.

The attribute integrity_check is used to indicate if data file check value (see ref [3] § 2.2.3.1.40, 2.2.3.1.41 and 2.2.3.1.42) has to be included into the header file (see ref [3] §2.2.3.1) and which integrity check algorithm (see ref [3] §5) has to be used. The possible values are:
-   MD5
-   SHA-1
-   Any other : no integrity check

The attribute is not mandatory. If omitted, no integrity check.

The attribute padding is used to indicate the ASCII of padding character to use for the last data file if necessary (when size is odd).

The content is a list of numbers. The possible values are from 0_{10} to 255_{10}.

The attribute is not mandatory. If omitted, ASCII of padding character will be set to 0_{10}.

Examples:
<INPUT_FILE>c:\temp\a.exe</INPUT_FILE>[1]
<INPUT_FILE integrity_check="SHA-1" padding="10">temp\a.exe</INPUT_FILE>[2]

5.3.9   Tag SPLIT_SIZE

This tag is to indicate the size in bytes of the generated data files (see ref [3] §2.2.3.2).

The content shall be a list of numbers. The size shall not be lower than 2_{10}, or greater or equal than 2^{31}_{10}, or an odd number.

This tag is not mandatory. If omitted, size is set to 32748_{10}.

Example:
<SPLIT_SIZE>10000</SPLIT_SIZE>

---

[1] Absolute path
[2] Relative path

5.3.10 Tag DIRECTORY

This tag is used to indicate the directory where to write header and data files.
The content shall be a writeable, relative or absolute path.
This tag is mandatory. If omitted, LSPB will generate an exception.
If the directory does not exist, it shall be created. If the directory exists and contains already data files, there will be erased by clicking the menu Generate.

Example:
<DIRECTORY>FILES</DIRECTORY>

5.3.11 Tag THW_ID_LIST

This tag is used to include the target hardware identifiers.
The content shall be the at least one tag THW_ID.
This tag is mandatory.

The attribute number is used to indicate the amount of target hardware identifiers.
The content shall be a list of numbers. The value shall be greater than 0.
The attribute is mandatory. If omitted, LSPB will generate an exception.

Example:
<THW_ID_LIST number="2">…<THW_ID_LIST >

5.3.12 Tag THW_ID

This tag is used to indicate at least one target hardware identifier (see ref [3] §2.2.3.1.21 and 2.2.3.1.22).
The content shall be a list of characters.
This tag is mandatory. There shall have as many tags THW_ID as value of attribute number in tag THW_ID_LIST.

Examples:
<THW_ID_LIST number="1">
        <THW_ID>THW ID1</THW_ID>
<THW_ID_LIST >[3]
<THW_ID_LIST number="2">
        <THW_ID>THW ID1</THW_ID>
        <THW_ID>THW ID2</THW_ID>
<THW_ID_LIST >[4]

5.3.13 Tag SUPPORT_FILE_LIST

This tag is used to include the support files.
The content shall be the at least one tag SUPPORT_FILE.
This tag is not mandatory.

The attribute integrity_check is used to indicate if support file check value (see ref [3] §2.2.3.1.52, 2.2.3.1.53 and 2.2.3.1.54) has to be included into the header file (see ref [3]

---

[3] Only one target hardware identifier
[4] Two target hardware identifiers

§2.2.3.1) and which integrity check algorithm (see ref [3] §5) has to be used. The possible values are:

- MD5
- SHA-1
- Any other : no integrity check

The attribute is not mandatory. If omitted, no integrity check.

The attribute number is used to indicate the amount of support files.
The content shall be a list of numbers.
The attribute is mandatory. If omitted, LSPB will generate an exception.

Example:
<SUPPORT_FILE_LIST number="2">…<SUPPORT_FILE _LIST >

5.3.14 Tag SUPPORT_FILE

This tag is used to indicate at least one support file (see ref [3] §2.2.3.1.44 through 2.2.3.1.51).
The content shall be a list of characters.
There shall have as many tags SUPPORT_FILE as value of attribute number in tag SUPPORT_FILE_LIST.

Examples:
<SUPPORT_FILE_LIST number="1">
      <SUPPORT_FILE>file1</SUPPORT_FILE>
<SUPPORT_FILE_LIST >[5]
<SUPPORT_FILE_LIST number="2">
      <SUPPORT_FILE>file1</SUPPORT_FILE>
      <SUPPORT_FILE>c:\temp\file1</SUPPORT_FILE>
<SUPPORT_FILE_LIST >[6]

5.3.15 Tag USER_DATA

This tag is used to include the user defined data (see ref [3] § 2.2.3.1.57).
The content shall be the tag USER_DATA_FILE or the tag USER_DATA_TEXT.
This tag is not mandatory.

5.3.16 Tag USER_DATA_FILE

This tag is used to indicate the path and the name to access the user data file. It can be an absolute path or a relative path. In case of relative path, the path where LSPB is invoked is added to the relative path.
The content shall be a list of characters. If the file is not readable, LSPB will generate an exception. The content of the file is written to the header file in user defined data field (see ref [3] §2.2.3.1.57).
This tag is not mandatory.

Examples:
<USER_DATA_FILE>c:\temp\user_data.txt</USER_DATA_FILE>[7]

---

[5] Only one support file
[6] Two support files
[7] Absolute path

<USER_DATA_FILE>user_data.bin</USER_DATA_FILE>[8]

5.3.17 Tag USER_DATA_TEXT

This tag is used to indicate text.
The content shall be a list of characters and is written to the header file in user defined data field (see ref [3] §2.2.3.1.57).
This tag is not mandatory.

Example:
<USER_DATA_TEXT>This is a text</USER_DATA_TEXT>

5.3.18 Tag USER_DATA_BCC

Not documented

5.3.19 Tag MEDIA

This tag is used to generate the following files:
- LOADS.LUM file (see ref [3], §3.2.3.1)
- FILES.LUM file (see ref [3], §3.2.3.2)
- BATCHES.LUM file (see ref [3] §3.2.3.3)

This tag is not mandatory. If omitted, no files will be generated.

Example:
<MEDIA></MEDIA>

6. LIMITATIONS

The limitations are sorted by report version and files.

6.1 ARINC665-1

This report is not yet supported.

6.2 ARINC665-2

6.2.1 General

- If you are using the media set generation, it's not possible to have a different part number for the load and the media set.
- It's not possible to have several integrity checks in case of several support files. The selected integrity check will apply to all support files.

6.2.2 Data files

- The maximum number of data files is 9999_{10}

6.2.3 List-of-Files file

- User Defined Data fields (ref [3] §3.2.3.2.26) not supported

---

[8] Relative path

- Media Sequence Number field (see ref [3] §3.2.3.2.11) is always set to 1
- Number Of Media Set Member field (see ref [3] §3.2.3.2.12) is always set to 1

6.2.4   List-of-Loads file

- User Defined Data fields (ref [3] §3.2.3.2.26) not supported
- Media Sequence Number field (see ref [3] §3.2.3.1.10) is always set to 1
- Number of Media Set Member field (see ref [3] §3.2.3.1.11) is always set to 1
- Number of Loads field (see ref [3] §3.2.3.1.12) is always set to 1.

6.2.5   List-of-Batch file

- User Defined Data fields (ref [3] §3.2.3.3.21) not supported
- Media Sequence Number field (see ref [3] § 3.2.3.3.10) is always set to 1
- Number of Media Set Member field (see ref [3] § 3.2.3.3.11) is always set to 1
- Number of Batches field (see ref [3] §3.2.3.3.12) always set to 1.
- Member Sequence Number field (see ref [3] §3.2.3.3.18) is always set to 1

6.2.6   Batch file

- Number of Loads for the Target HW ID POS (see ref [3] §2.3.1.15) always set to 1

6.3   ARINC665-3

6.3.1   General

- If you are using the media set generation, it's not possible to have a different part number for the load and the media set.
- It's not possible to have several integrity checks in case of several support files. The selected integrity check will apply to all support files.

6.3.2   Header file

- Only MD5 and SHA-1 integrity check algorithms are supported
- Integrity check of load (see ref [3] § 2.2.3.1.59, 2.2.3.1.60 and 2.2.3.1.61) not supported (Load Check Value Length forced to 0)
- Number of Target HW ID with Positions fields (see ref [3] §2.2.3.1.24 through 2.2.3.1.29) not supported

6.3.3   Data files

- Only MD5 and SHA-1 integrity check algorithms are supported
- The maximum number of data files is 9999_{10}

6.3.4   List-of-Files file

- Only MD5 and SHA-1 integrity check algorithms are supported
- User Defined Data fields (ref [3] §3.2.3.2.26) not supported
- Media Sequence Number field (see ref [3] §3.2.3.2.11) is always set to 1
- Number of Media Set Member field (see ref [3] §3.2.3.2.12) is always set to 1
- File Check Value Length filed for Media Set Files (see ref [3] §3.2.3.2.21) is always set to 0

6.3.5 List-of-Loads file

- Only MD5 and SHA-1 integrity check algorithms are supported
- User Defined Data fields (ref [3] §3.2.3.2.26) not supported
- Media Sequence Number field (see ref [3] §3.2.3.1.10) is always set to 1
- Number of Media Set Member field (see ref [3] §3.2.3.1.11) is always set to 1
- Number of Loads field (see ref [3] §3.2.3.1.12) is always set to 1.

6.3.6 List-of-Batch file

- User Defined Data fields (ref [3] §3.2.3.3.21) not supported
- Media Sequence Number field (see ref [3] § 3.2.3.3.10) is always set to 1
- Number of Media Set Member field (see ref [3] § 3.2.3.3.11) is always set to 1
- Number of Batches field (see ref [3] §3.2.3.3.12) always set to 1.
- Member Sequence Number field (see ref [3] §3.2.3.3.18) is always set to 1

6.3.7 Batch file

- Number of Loads for the Target HW ID POS (see ref [3] §2.3.1.15) always set to 1

7. LOAD STRUCTURE

The load structure and files are the following:
- FILES.LUM
- LOADS.LUM
- BATCHES.LUM (if required by tag BATCH)
- Batch file (.LUB, if required by tag BATCH)
- Sub directory (see tag DIRECTORY)
  - Header file (.LUH)
  - Data files (.LUP)

7.1 FILENAMES

7.1.1 Header file

The name of the header file is built with the part number of the load (see tag PART_NUMBER) including MMM and CC and then adding the extension LUH (see ref [3] § 2.2.2.1).

7.1.2 Data files

The name of the data files is built with the part number of the load (see tag PART_NUMBER) including MMM and CC, then adding an underscore, then adding a 4 characters number (from 0000 to 9999) and finally adding the extension LUP (see ref [3] § 2.2.2.2).
The amount of data files depends on the size of input file and the split size. If the split size is greater or equal than size of input file, there will be only on data file.

7.1.3 Batch file

The name of the batch file is built with the part number of the load (see tag PART_NUMBER) including MMM and CC and then adding the extension LUB (see ref [3] § 2.2.2.1).