

MMA Technical Standards Board/ AMEI MIDI Committee

Letter of Agreement for Recommend Practice

XMF Compression Definition for ZLIB

Primary Authors: XMF Working Group

Editors: Chris Grigg

Abstract:

This RP defines a new XMF Standard UnpackerID for "ZLIB" compression.

Background:

During preparation of the initial versions of the XMF specifications, no Standard UnpackerIDs were defined. At that time, the XMF Working Group found it prohibitive to designate any Unpackers as standard, given that most of the available methods for the purpose are either proprietary, non-portable, and/or legally encumbered.

ZLIB claims that it is designed to be a free, general-purpose, legally unencumbered -- that is, not covered by any patents -- lossless data-compression library for use on virtually any computer hardware and operating system. The MMA is not warranting such claims, and is including a disclaimer as part of the proposal.

Details:

DISCLAIMER

The fact that the MMA XMF specification has a field that allows specification of certain compression algorithms is not in any way a representation, warranty, or endorsement regarding any compression algorithm, nor is it a representation or warranty that the vendor of any such compression algorithm is the owner of the intellectual or other property rights involved therein. Rather, the use of any compression algorithm to which a value has been assigned in the appropriate field in the XMF specification is solely at the user's risk.

1. XMF Standard UnpackerID for zlib

A Standard UnpackerID of 0x01 will indicate the Node is packed using zlib compression (refer to section 5.1 of the Specification for XMF Meta File Format).

Example:

```
0x00 // Standard UnpackerID follows
0x01 // zlib Standard UnpackerID
```

2. zlib Usage in XMF

The intent of zlib as an XMF Unpacker is for the data compression (size reduction) of a File Node. Refer to the Specification for XMF Meta File Format, Sections 2.2.1.1.1 and 5.1 for the prescribed usage of an Unpacker.

Note: In XMF Files, FolderNodes appearing in the Tree must not use any unpackers that would render the folder's contained Nodes illegible. This rule arises because otherwise we'd need to invent a way to describe how the Tree continues on the inside of that 'black box'. If you wish to 'black box' encode a folder, however, it is possible to store it outside the Tree, as a detached Node (see section 2.2.1.2.1 of the XMF Meta File Format), and point the FolderNode's ContentReference at the detached Node.

Along with the freely available zlib code library and efforts to create code libraries and projects for nearly all available operating systems, zlib should be fairly easy for most computer based XMF parsers to implement. The zlib source code is available in C language, which should make it quite possible for implementation in embedded systems that support C compilation.

It should be noted that zlib uses a lossless general-purpose compression algorithm, and is not optimized for audio files. For example, it will typically not perform well on .WAV files. The expected usage for zlib within XMF would be to compress SMF files and other types of included data, such as text files, that would benefit from zlib's lossless type of compression.

3. zlib Resources

All documentation, source code, libraries, and projects for zlib can be found on the zlib website:
<http://www.zlib.org>

4. zlib License

Please refer to http://www.zlib.org/zlib_license.html for the latest version of the zlib license. For MMA members' reference at balloting time, the current license text is attached.

ZLIB LICENSE TEXT FROM http://www.zlib.org/zlib_license.html AS OF DEC 05 2002

```
/*      zlib.h -- interface of the 'zlib' general purpose compression library
version 1.1.3, July 9th, 1998
Copyright (C) 1995-1998 Jean-loup Gailly and Mark Adler
This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any
damages arising from the use of this software.
Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and
redistribute it freely, subject to the following restrictions:
1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you
use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original
software.
3. This notice may not be removed or altered from any source distribution.
```

Jean-loup Gailly Mark Adler
jloup@zip.org madler@alumni.caltech.edu

The data format used by the zlib library is described by RFCs (Request for Comments) 1950 to 1952 in the files
<ftp://ds.internic.net/rfc/rfc1950.txt> (zlib format), [rfc1951.txt](ftp://ds.internic.net/rfc/rfc1951.txt) (deflate format) and [rfc1952.txt](ftp://ds.internic.net/rfc/rfc1952.txt) (gzip format).
*/

5. Publication Plan:

1. Parties who have received copies of previous versions of **XMF Specification Incorporating all Recommended Practices** should be notified that a Standard UnpackerID for ZLIB has been approved.
2. Notification of approval of the Standard UnpackerID definition for ZLIB should be posted on the MMA web site, including all technical details contained in this document.
3. The MMA may wish to issue a press release to announce the existence of the Standard UnpackerID for ZLIB.
4. Parties known to be producing XMF content, tools, or technology should be notified of the existence of the Standard UnpackerID for ZLIB, if possible.

First Date of Issue: May 10, 2003Originated by: MMA AMEI

Revision Date: _____

Reference TSB Item #: 178Reference TSBB Volume #: 27Title: XMF Compression Definition for ZLIB RP#: 40Related item(s): XMF Meta File Format SpecificationApproved by MMA: Date: 5/10/03

[Jan 19 2003]

Approved by AMEI: Sechi GatsDate: 6/17/03