

MMA Technical Standards Board/ AMEI MIDI Committee

Confirmation of Approval of New MIDI Message

File Reference System Exclusive Message (CA-018)

This message allows computer based MIDI devices to access files on a shared file system or network using a URL-style file specification. It allows the sounds in these files to be played using simple MIDI messages.

Because of the nature of computer-based MIDI devices and their ability to directly (or indirectly, through software drivers) access files on a shared file system or network, a file reference mechanism using URL-style file names allows these devices to optimally load sounds without transferring the file contents over MIDI. Additionally, the existence of standard URL managers on newer operating systems allows MIDI device manufacturers to implement this file reference mechanism easily to enable such features as DLS file loading and WAV file loading using existing MIDI sequencers and software.

The DLS Level 1 specification does not provide an API definition, and currently no operating system vendors have implemented one. Even though DLS APIs have been promised by OS vendors, their delivery has been repeatedly delayed and their use will require existing MIDI software to be rewritten. Proprietary APIs have been developed by some manufacturers, but these do not address the broad needs of the industry by providing a consistent delivery mechanism for downloadable sounds. Because DLS file loading is really a subset of the generic concept of file loading, it is simpler to provide a file reference mechanism that can be used for DLS file loading as well as other file types like WAV files, SoundFont files or other file types.

This solution adds a new MIDI system exclusive non-realtime message to the MIDI specification. This message is intended to be used for the purpose of loading sound files from a shared file system or network, and playing them using standard MIDI channel events (program change, note events, continuous controllers, etc.). The message consists of a URL-based file name and a set of file type parameters specified using the following MIDI system exclusive data format:

FILE REFERENCE MESSAGE

This message has a variable length. The Sub-ID #2 is used to specify the file operation.

```
F0 7E <device ID> 0B <sub-ID#2> <ctx> <len> <data> F7
```

```
F0 7E      Universal Non-Real Time SysEx header
<device ID> Target device ID (suggest using 7F: broadcast)
0B        New sub-id#1
<sub-ID#2> File Reference Message command code:
          00 - reserved
          01 - Open file
          02 - Select or reselect contents
          03 - Open file and select contents
          04 - Close file
          05 to 7F - reserved
<ctx>     Two 7-bit bytes: command context.
```

This context is used to distinguish multiple file operations to the same device. This value can be anything the sender wants, as long as multiple active operations are differentiated.

```
<len>     Two 7-bit bytes: byte count of <data>, LSB first.
```

This limits all <data> blocks to 16383 bytes or less. This count is the number of bytes remaining in the message before the end-of-exclusive byte.

```
<data>    Command specific data. See below.
```

OPEN FILE

This message specifies the type and location of a sound file so it can be loaded. It does not cause the sound to play. A SELECT CONTENTS message, followed by the appropriate MIDI channel events, must be sent for the sound to play.

A file can be opened multiple times using different <ctx> values if it needs to be processed using multiple SELECT CONTENTS messages. If a <ctx> value is used that matches one already in use by an open file, that file will be closed.

```
F0 7E <device ID> 0B 01 <ctx> <len> <type> <url> F7
```

```
<device ID> Target device ID
<ctx>       Two 7-bit bytes: command context.
<len>       Two 7-bit bytes: combined byte count of <type>
            and <url>, LSB first.
```

```
Command specific data:
```

```
<type>     Four 7-bit ASCII bytes: case insensitive file type.
```

Type	Hex value and Description
----	-----
"DLS "	44 4C 53 20
"dls "	64 6C 73 20

DLS format files; if DLS files are supported, the minimum requirement is to support DLS Level 1 formatted files.

Type	Hex value and Description
----	-----
"SF2 "	53 46 32 20
"sf2 "	73 66 32 20

SoundFont format files; if SoundFont files are supported, the requirement is to support SoundFont 2 formatted files.

Type	Hex value and Description
----	-----
"WAV "	57 41 56 20
"wav "	77 61 76 20

Microsoft Wave (WAV) format files; if WAV files are supported, they must contain the following chunks: <fmt >, <data>. Optional chunks are: <inst>, <smpl>

<url> Variable length, NULL terminated, 7-bit ASCII bytes:
URL format file name.

Maximum URL length including terminator is 261 bytes.
(People familiar with Microsoft Windows will recognize this as the value MAX_PATH+1)
The URL prefix can be any valid prefix. The "http://" and "file://" prefixes should be used whenever possible.

See the notes on using the URL in the Comments section below.

SELECT CONTENTS

This message specifies the information necessary to play sounds in the file using subsequent MIDI channel messages.

If a loaded sound overlaps a built-in (default or ROM) sound, i.e., if it has the same program and variation number and covers the same key range, the loaded sound will replace (not layer with) the built-in sound. If two or more loaded sounds conflict, the most recently loaded sound will play.

If additional SELECT CONTENTS messages are sent using the same context value, only the most recent selection is used. If a sound is to be played using multiple selects, it must be opened using a different context value for each select.

The format of this message depends on the file type specified in the OPEN FILE message.

Select data for a DLS or SoundFont file:

F0 7E <device ID> 0B 02 <ctx> <len> <count> ([<dst-bank> <dst-prog> <src-bank> <src-prog> <flags> <vol>] repeated <count> times) [<ext-data>] F7

<device ID> Target device ID
 <ctx> Two 7-bit bytes: command context used with OPEN FILE.
 <len> Two 7-bit bytes: byte count of data, LSB first.
 This value is minimally equal to 1 + (count * 8). If it is larger, <ext-data> is present.

Command specific data:

<count> 7-bit byte: Number of maps.
 If <count> is zero, use mapping information provided in the DLS file. Each instrument select map contains:

<dst-bank> Two 7-bit bytes: MIDI bank number, MSB first (MIDI CC0 followed by CC32), required to select sound for playing (use 00 00 for default).
 For more information, see the Comments section below.

<dst-prog> 7-bit byte: MIDI program number required to select sound for playing.

<src-bank> Two 7-bit bytes: MIDI bank number stored in the instrument header in the file, MSB first (MIDI CC#0 followed by CC#32).
 For DLS files, this value selects the 'ins ' with an 'insh' subchunk having a matching value in bits <8..14> and <0..6> of the Locale.ulBank field.
 For SoundFont files, this value selects the instrument with a 'phdr' subchunk having a matching value in the wBank field.

<src-prog> 7-bit byte: MIDI program number stored in the instrument header in the file.
 For DLS files, this value selects the 'ins ' with an 'insh' subchunk having a matching value in bits <0..6> of the Locale.ulInstrument field.
 For SoundFont files, this value selects the instrument with a 'phdr' subchunk having a matching value in the wPreset field.

<flags> 7-bit byte: Bit field containing the following flags:

Bit	Flag
---	----
0	Source Drum flag. This flag indicates the selected instrument is a drum instrument. For DLS files, this bit is used to match bit <31> of the Locale.ulBank field.

MMA/AMEI Confirmation of Approval — CA-018

- 1 Destination Drum flag. This flag indicates the selected instrument is a drum instrument. When set, this flag indicates that the instrument should be loaded as a drum instrument.
- 2..6 Reserved, must be zero.

<vol> 7-bit byte: Initial volume (use 7F for default).

<ext-data> Optional extension data. This is provided for future expansion of the message, and shall be ignored by any devices that do not support it.

Select data for a WAV file:

F0 7E <device ID> 0B 02 <ctx> <len> <dst-bank> <dst-prog> <base> <lokey>
<hikey> <fine> <vol> [<ext-data>] F7

<device ID> Target device ID

<ctx> Two 7-bit bytes: command context used with OPEN FILE.

<len> Two 7-bit bytes: byte count of data, LSB first.
This value is minimally equal to 9. If it is larger, <ext-data> is present.

Command specific data:

<dst-bank> Two 7-bit bytes: MIDI bank number, MSB first (MIDI CC#0 followed by CC#32), required to select sound for playing (use 00 00 for default).
For more information see the Comments section below.

<dst-prog> 7-bit byte: MIDI program number required to select sound for playing.

<base> 7-bit byte: MIDI note where sound plays at original pitch.

<lokey> 7-bit byte: Lowest MIDI note that plays sound (use 00 for default).

<hikey> 7-bit byte: Highest MIDI note that plays sound (use 7F for default).

<fine> Two 7-bit bytes: Fine tuning offset, LSB first. The format of this value is similar to that for RPN 01 - FINE TUNING as follows:

LSB	MSB	Tuning offset
---	---	-----
00	00	-100.0 cents
00	40	0.0 cents
7F	7F	+100.0 cents

MMA/AMEI Confirmation of Approval — CA-018

<vol> 7-bit byte: Initial volume (use 7F for default). These values override any values specified in the <smpl> or <inst> RIFF chunks of the WAV file.

<ext-data> Optional extension data. This is provided for future expansion of the message, and shall be ignored by any devices that do not support it.

OPEN FILE AND SELECT CONTENTS

This message is a combination of the OPEN FILE and SELECT CONTENTS messages (for convenience).

F0 7E <device ID> 0B 03 <ctx> <len> <type> <url> <select data> F7

<device ID> Target device ID
<ctx> Two 7-bit bytes: command context.
<len> Two 7-bit bytes: byte count of data, LSB first.
<type> Same as in OPEN FILE message above.
<url> Same as in OPEN FILE message above.
<select data> Command specific data for the SELECT CONTENTS message above.

CLOSE FILE

This message closes an open file and deallocates any memory allocated for the file's contents. Sounds from the file will no longer play where selected, and any built-in or previously loaded sounds disabled by the sounds selected from this file will be re-enabled.

F0 7E <device ID> 0B 04 <ctx> 00 00 F7

<device ID> Target device ID
<ctx> Two 7-bit bytes: command context used with OPEN FILE (or OPEN FILE AND SELECT CONTENTS).

Comments:

Bank number

Loaded sounds can be played on a MIDI channel by sending the bank number as MIDI CC #0 and CC #32, immediately followed by the MIDI program number. Subsequent MIDI note events that match the key range selected will play the loaded sound only; built-in or previously loaded sounds that could have been selected by the same MIDI note event will not sound. This allows the sound designer to replace built-in sounds or provide missing sounds or drum instruments.

Because of conflicting interpretations of MIDI bank select by various manufacturers (Roland GS uses CC #0 for variation select, and CC #32 for mode select in some devices, while Yamaha XG uses CC #0 for mode select and CC #32 for variation select), it is recommended that the `<dst-bank>` field be specified using MMA recommended practices.

The Source Drum flag represents the bit 31 of the `Locale.ulBank` field in the DLS file. This bit should be combined with the 14 bits of `<src-bank>` information as described in the DLS document to form a complete bank address when locating instrument data in a DLS file.

The Destination Drum flag represents the desired mode of operation of the instrument when it is loaded into the DLS synthesizer. This bit should be combined with the 14-bits of `<dst-bank>` information to form a complete bank address. This allows for a drum instrument to be loaded as a melodic instrument or vice versa. For example, if a drum instrument is loaded from a DLS file and the Destination Drum flag is clear, the instrument will be available as a melodic instrument on channels 1-9 and 11-16. This allows for multiple drum kits to be loaded simultaneously.

IMPORTANT NOTE FOR DLS-1 SYNTHESIZERS: DLS-1 synthesizers are not required to support global articulation data for drum instruments and are not required to support local articulation data for melodic instruments. Therefore, attempts to load a drum instrument into a melodic bank, or a melodic instrument into a drum bank, may fail on DLS-1 synthesizers.

URL handling

URL file names are platform independent, and provide the ability for SMFs to reference files they depend on without regard to the platform or relative location they are played from. The processing of URL format file names can be handled by APIs provided with most modern operating systems, so any MIDI device manufacturer that provides a proprietary API for downloading sounds on these operating systems can implement this SysEx message handler relatively easily.

Manufacturers of external MIDI devices can be supported generically by the development of an external MIDI device driver that intercepts `FILE REFERENCE` messages and translates them to `FILE DUMP` messages. For example, under Win95, using DirectX 3.0 or later, there is a DLL called `URLMON.DLL` (located in the system directory) that provides complete management of URL resources. This includes network and dialup service access, file streaming and caching, local "file://" protocol name translation, etc. Because of the potential delays associated with downloading sound files from the World Wide Web, `FILE REFERENCE` messages should precede their use by the maximum possible time. Under no circumstances should the reception of MIDI events be delayed while waiting for file downloads over a network. The MIDI device should simply not play the sound until it is loaded. OS-provided URL managers will only cause a delay the first time a file is loaded; once files are cached, file access is as fast as accessing files on the local machine.

Extension data

MMA/AMEI Confirmation of Approval — CA-018

Care should be taken when adding fields to the <ext-data> of the SELECT CONTENTS message. It is recommended that new fields use a header consisting of a two-byte extension ID (provided by the MMA upon request) and a two-byte field size (LSB first). This will allow devices to skip unsupported fields while retaining the ability to locate supported fields within the extension area.

Date of issue: 11/4/98 (Revised 10/27/99)

Originated by: MMA

Reference TSBB Item #: 144

TSBB Volume #: 22 (revised)

Title: File Reference SysEx Message

CA#: 018

Related item(s): System Exclusive Messages; Universal Exclusive ID