SOUND4 Stream.CL Library [2.0.1]

Generated by Doxygen 1.9.8

1 SOUND4 Stream.CL Library **1**

1.1 General usage . . . 1

1.2 Installation . . . 1

1.3 Integrators . . . 1

1.4 Setup Parameters . . . 2

1.5 Metadata . . . 2

1.6 Metadata XML format . . . 3

1.7 Ad break signaling . . . 3

1.7.1 Enabling it on a stream . . . 3

1.7.2 Starting and stopping a break . . . 4

1.7.3 Timing . . . 4

2 File Index **5**

2.1 File List . . . 5

3 File Documentation **7**

3.1 documentation.h File Reference . . . 7

3.2 documentation.h . . . 7

Index **9**

Chapter 1

SOUND4 Stream.CL Library

1.1 General usage

This library can be used with SOUND4 .CL HD and WEB processing families. This extension supports Live processing only.

1.2 Installation

You just have to put this library in the same directory of the process, and it will detect and use it, if possible.

You can check in the About window, it should appear with its version.

You will need a `Stream.CL` license activated by SOUND4.

You can put a folder named "SOUND4 Stream.CL Metaparsers" in the same directory, which contains some `Lua` files with predefined metadata parser scripts. For Linux/Darwin, this folder can also be in (libpath)/../share/sound4/.

1.3 Integrators

When integrating in your software, like radio automation software, you can automatically set the metadata from it, using its SetMetadata function.

You should set the `Metadata_no_ui` before startup to avoid that the user can manage it on his side (Setup Parameters).

1.4 Setup Parameters

Those parameters can be send to the process at startup by using its `SetParameter` function before creating the instance. They will configure what will be available to the user in the interface :

- `Metadata_no_ui` : If set, disable the user access to metadata (0/1)
- `Metadata_provided_path` : A directory were there are software provided metadata scripts (.lua). If not set, will search in directory "SOUND4 Stream.CL Metaparsers" located where the library is.
- `Metadata_no_auto_path` : If set, disable the search of scripts in "SOUND4 Stream.CL Metaparsers" (0/1)
- `Metadata_server_maxclient` : Set the maximum count of client to metadata TCP server (default: 1)
- `LICENSE_CERT_DIR` : optional certificates dirs for HTTPS verification
- `LICENSE_CERT_BUNDLE` : optional certificates bundle for HTTPS verification
- `LICENSE_CERT_CHECK` : if "false", do not check HTTPS certificates

Advanced runtime setup :

- `STREAM_THREAD_PRIO` : force the streaming threads priority.
  - For **Linux**, the thread will use SCHED_RR and priority should be in [1,99].
  - For **Windows**, this should be the numeric value of one of THREAD_PRIORITY_*.
  - For **Darwin**, this should be in [-20,20].

1.5 Metadata

Metadata can be used to set the stream artist/title/song/... information that will be passed to the streaming servers.

The following metadata can be passed :

- `title` : The title of the current song
- `artist` : The name of the artist of the current song
- `compact_song` : The compact_song sent to stream servers using single metadata info. Automatically built from artist and title if not set
- `xml_metadata` : The metadata in XML form. See Metadata XML format
- `adbreak` : Ad break: `"true"` or `"1"` starts it, any other value (e.g. `"false"`) stops it. Deleting the key changes nothing. See Ad break signaling

1.6 Metadata XML format

The metadata XML should have the following format (each is optional):

```
<?xml version="1.0" encoding="UTF-8" ?>
<metadata>
     <title>Strange Fruit</title>
     <artist>Billie Holiday</artist>
     <!-- compact_song will be made automatically with artist - title if not specified -->
     <compact_song>Billie Holiday - Strange Fruit</compact_song>
     <!-- adbreak: "true" starts an ad break, "false" stops it, without it nothing changes.
          Never sent to the servers. See the Ad break section -->
     <adbreak>false</adbreak>
     <_ice>
         <!-- optional Icecast specific -->
         <title>Strange Fruit</title>
         <artist>Billie Holiday</artist>
     </_ice>
     <_shout>
         <!-- optional Shoutcast specific -->
         <TPE1>Billie Holiday</TPE1>
         <TIT2>Strange Fruit</TIT2>
         <TCON v1="24">Subgenre</TCON>
         <extension>
             <soon>The Next Artist - The Next Title</soon>
         </extension>
     </_shout>
     <_id3>
         <!-- optional ID3 format (used in MPEG-TS) -->
         <TPE1>Billie Holiday</TPE1>
         <TIT2>Strange Fruit</TIT2>
         <TCON>Rock</TCON>
     </_id3>
     <_rtmp>
         <!-- optional RTMP specific -->
         <!-- this will add a field with key "mycommand" in the onMetadata notify AMF block -->
         <mycommand>Send command</mycommand>
     </_rtmp>
</metadata>
```

1.7 Ad break signaling

An ad break can be signaled inside the encoded streams, so that a downstream ad insertion system can detect it in-band: while a break is in progress, a flag is set in the header of every audio frame of the streams that enable it.

- AAC, HE-AAC, HE-AACv2 (ADTS frames): `copyright_identification_start`
- MP3, MP2 (MPEG audio frames): `copyright`

1.7.1 Enabling it on a stream

Each stream has its own option (`adbreak_copyright`, "Ad Break Flag"). It is offered only when the output keeps such a frame header:

- MP3 and MP2: every protocol offering them (the frame headers are kept, RTP included).
- AAC family: every protocol carrying ADTS (Icecast, Shoutcast, UDP, SRT, MPEG-TS, HLS). Not on RTMP, RTSP and RTP without MPEG-TS, which carry raw AAC frames without header.

Synchronized HLS streams follow their main stream. Opus, Vorbis, FLAC and PCM have no such flag.

1.7.2 Starting and stopping a break

A break is started and stopped by one of these sources, usually the radio automation:

- the `adbreak` metadata key (see Metadata), e.g. `SetMetadata("adbreak", "true");`
- the `adbreak` node of the XML metadata (see Metadata XML format), for instance set by a metadata parser script: `"true"` on the ad items, `"false"` on the first item after the break;
- the `adbreak` general setting, through the JSON API. It is a command, never saved in presets.

Stopping is always explicit: metadata without the `adbreak` node (e.g. a new title), or deleting the `adbreak` key, changes nothing. Each value given by the metadata applies; the setting applies when it changes. The last one from any source wins: any source can start or stop a break. Use one source only.

1.7.3 Timing

The change is taken with the audio being processed when it is received, and applied on all streams to the frames carrying that audio (encoder delay compensated), whatever each stream buffers. The resolution is one frame: 1024 samples for AAC, 2048 for HE-AAC, 1152 for MP3 and MP2 (about 21, 43 and 24 ms at 48 kHz). The latency of the source itself (automation, metadata polling...) is not compensated.

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

documentation.h 7

Chapter 3

File Documentation

3.1 documentation.h File Reference

3.2 documentation.h

Go to the documentation of this file.
```
00001
```

Index

documentation.h, 7

SOUND4 Stream.CL Library, 1