SFML

* [Main Page](http://docs.google.com/index.htm)
* [Modules](http://docs.google.com/modules.htm)
* [Classes](http://docs.google.com/annotated.htm)
* [Files](http://docs.google.com/files.htm)
* [Class List](http://docs.google.com/annotated.htm)
* [Class Index](http://docs.google.com/classes.htm)
* [Class Hierarchy](http://docs.google.com/hierarchy.htm)
* [Class Members](http://docs.google.com/functions.htm)
* **sf**
* [SoundBuffer](http://docs.google.com/classsf_1_1SoundBuffer.htm)

[Public Member Functions](#_gjdgxs) | [Friends](#_30j0zll) | [List of all members](http://docs.google.com/classsf_1_1SoundBuffer-members.htm)

sf::SoundBuffer Class Reference

[Audio module](http://docs.google.com/group__audio.htm)

Storage for audio samples defining a sound. [More...](http://docs.google.com/classsf_1_1SoundBuffer.htm#details)

#include <[SoundBuffer.hpp](http://docs.google.com/SoundBuffer_8hpp_source.htm)>

| Public Member Functions | |
| --- | --- |
|  | [SoundBuffer](http://docs.google.com/classsf_1_1SoundBuffer.htm#a0cabfbfe19b831bf7d5c9592d92ef233) () |
|  | Default constructor. |
|  | |
|  | [SoundBuffer](http://docs.google.com/classsf_1_1SoundBuffer.htm#aaf000fc741ff27015907e8588263f4a6) (const [SoundBuffer](http://docs.google.com/classsf_1_1SoundBuffer.htm) &copy) |
|  | Copy constructor. |
|  | |
|  | [~SoundBuffer](http://docs.google.com/classsf_1_1SoundBuffer.htm#aea240161724ffba74a0d6a9e277d3cd5) () |
|  | Destructor. |
|  | |
| bool | [loadFromFile](http://docs.google.com/classsf_1_1SoundBuffer.htm#a2be6a8025c97eb622a7dff6cf2594394) (const std::string &filename) |
|  | Load the sound buffer from a file. |
|  | |
| bool | [loadFromMemory](http://docs.google.com/classsf_1_1SoundBuffer.htm#af8cfa5599739a7edae69c5cba273d33f) (const void \*data, std::size\_t sizeInBytes) |
|  | Load the sound buffer from a file in memory. |
|  | |
| bool | [loadFromStream](http://docs.google.com/classsf_1_1SoundBuffer.htm#ad292156b1e01f6dabd4c0c277d5e079e) ([InputStream](http://docs.google.com/classsf_1_1InputStream.htm) &stream) |
|  | Load the sound buffer from a custom stream. |
|  | |
| bool | [loadFromSamples](http://docs.google.com/classsf_1_1SoundBuffer.htm#a63da986e144b578135edd48e51c565e8) (const Int16 \*samples, std::size\_t sampleCount, unsigned int channelCount, unsigned int sampleRate) |
|  | Load the sound buffer from an array of audio samples. |
|  | |
| bool | [saveToFile](http://docs.google.com/classsf_1_1SoundBuffer.htm#ab2083dc1a934c64959d9e3f162328a76) (const std::string &filename) const |
|  | Save the sound buffer to an audio file. |
|  | |
| const Int16 \* | [getSamples](http://docs.google.com/classsf_1_1SoundBuffer.htm#a4ba0c1e5b5be500af42de30b1360eb2e) () const |
|  | Get the array of audio samples stored in the buffer. |
|  | |
| std::size\_t | [getSampleCount](http://docs.google.com/classsf_1_1SoundBuffer.htm#a2df1f1ae89213adee1494b95bb583f9d) () const |
|  | Get the number of samples stored in the buffer. |
|  | |
| unsigned int | [getSampleRate](http://docs.google.com/classsf_1_1SoundBuffer.htm#a8cdfaea2ad1d05f81fa67442566c166e) () const |
|  | Get the sample rate of the sound. |
|  | |
| unsigned int | [getChannelCount](http://docs.google.com/classsf_1_1SoundBuffer.htm#a0a2890747db3811fb8d969d3e3abd0d1) () const |
|  | Get the number of channels used by the sound. |
|  | |
| [Time](http://docs.google.com/classsf_1_1Time.htm) | [getDuration](http://docs.google.com/classsf_1_1SoundBuffer.htm#aee681c7a0f3dff4c4d0c9f8bbdb51bb0) () const |
|  | Get the total duration of the sound. |
|  | |
| [SoundBuffer](http://docs.google.com/classsf_1_1SoundBuffer.htm) & | [operator=](http://docs.google.com/classsf_1_1SoundBuffer.htm#adcc786b60bbd95be1551368fafd274a7) (const [SoundBuffer](http://docs.google.com/classsf_1_1SoundBuffer.htm) &right) |
|  | Overload of assignment operator. |
|  | |

| Friends | |
| --- | --- |
| class | **Sound** |
|  | |

## Detailed Description

Storage for audio samples defining a sound.

A sound buffer holds the data of a sound, which is an array of audio samples.

A sample is a 16 bits signed integer that defines the amplitude of the sound at a given time. The sound is then restituted by playing these samples at a high rate (for example, 44100 samples per second is the standard rate used for playing CDs). In short, audio samples are like texture pixels, and a [sf::SoundBuffer](http://docs.google.com/classsf_1_1SoundBuffer.htm) is similar to a [sf::Texture](http://docs.google.com/classsf_1_1Texture.htm).

A sound buffer can be loaded from a file (see [loadFromFile()](http://docs.google.com/classsf_1_1SoundBuffer.htm#a2be6a8025c97eb622a7dff6cf2594394) for the complete list of supported formats), from memory, from a custom stream (see [sf::InputStream](http://docs.google.com/classsf_1_1InputStream.htm)) or directly from an array of samples. It can also be saved back to a file.

[Sound](http://docs.google.com/classsf_1_1Sound.htm) buffers alone are not very useful: they hold the audio data but cannot be played. To do so, you need to use the [sf::Sound](http://docs.google.com/classsf_1_1Sound.htm) class, which provides functions to play/pause/stop the sound as well as changing the way it is outputted (volume, pitch, 3D position, ...). This separation allows more flexibility and better performances: indeed a [sf::SoundBuffer](http://docs.google.com/classsf_1_1SoundBuffer.htm) is a heavy resource, and any operation on it is slow (often too slow for real-time applications). On the other side, a [sf::Sound](http://docs.google.com/classsf_1_1Sound.htm) is a lightweight object, which can use the audio data of a sound buffer and change the way it is played without actually modifying that data. Note that it is also possible to bind several [sf::Sound](http://docs.google.com/classsf_1_1Sound.htm) instances to the same [sf::SoundBuffer](http://docs.google.com/classsf_1_1SoundBuffer.htm).

It is important to note that the [sf::Sound](http://docs.google.com/classsf_1_1Sound.htm) instance doesn't copy the buffer that it uses, it only keeps a reference to it. Thus, a [sf::SoundBuffer](http://docs.google.com/classsf_1_1SoundBuffer.htm) must not be destructed while it is used by a [sf::Sound](http://docs.google.com/classsf_1_1Sound.htm) (i.e. never write a function that uses a local [sf::SoundBuffer](http://docs.google.com/classsf_1_1SoundBuffer.htm) instance for loading a sound).

Usage example:

// Declare a new sound buffer

[sf::SoundBuffer](http://docs.google.com/classsf_1_1SoundBuffer.htm) buffer;

// Load it from a file

if (!buffer.[loadFromFile](http://docs.google.com/classsf_1_1SoundBuffer.htm#a2be6a8025c97eb622a7dff6cf2594394)("sound.wav"))

{

// error...

}

// Create a sound source and bind it to the buffer

[sf::Sound](http://docs.google.com/classsf_1_1Sound.htm) sound1;

sound1.[setBuffer](http://docs.google.com/classsf_1_1Sound.htm#a8b395e9713d0efa48a18628c8ec1972e)(buffer);

// Play the sound

sound1.[play](http://docs.google.com/classsf_1_1Sound.htm#a2953ffe632536e72e696fd880ced2532)();

// Create another sound source bound to the same buffer

[sf::Sound](http://docs.google.com/classsf_1_1Sound.htm) sound2;

sound2.[setBuffer](http://docs.google.com/classsf_1_1Sound.htm#a8b395e9713d0efa48a18628c8ec1972e)(buffer);

// Play it with a higher pitch -- the first sound remains unchanged

sound2.[setPitch](http://docs.google.com/classsf_1_1SoundSource.htm#a72a13695ed48b7f7b55e7cd4431f4bb6)(2);

sound2.[play](http://docs.google.com/classsf_1_1Sound.htm#a2953ffe632536e72e696fd880ced2532)();

See Also[sf::Sound](http://docs.google.com/classsf_1_1Sound.htm), [sf::SoundBufferRecorder](http://docs.google.com/classsf_1_1SoundBufferRecorder.htm)

Definition at line [52](http://docs.google.com/SoundBuffer_8hpp_source.htm#l00052) of file [SoundBuffer.hpp](http://docs.google.com/SoundBuffer_8hpp_source.htm).

## Constructor & Destructor Documentation

| sf::SoundBuffer::SoundBuffer | ( |  | ) |  |
| --- | --- | --- | --- | --- |

Default constructor.

| sf::SoundBuffer::SoundBuffer | ( | const [SoundBuffer](http://docs.google.com/classsf_1_1SoundBuffer.htm) & | *copy* | ) |  |
| --- | --- | --- | --- | --- | --- |

Copy constructor.

Parameters

| copy | Instance to copy |
| --- | --- |

| sf::SoundBuffer::~SoundBuffer | ( |  | ) |  |
| --- | --- | --- | --- | --- |

Destructor.

## Member Function Documentation

| unsigned int sf::SoundBuffer::getChannelCount | ( |  | ) | const |
| --- | --- | --- | --- | --- |

Get the number of channels used by the sound.

If the sound is mono then the number of channels will be 1, 2 for stereo, etc.

ReturnsNumber of channels See Also[getSampleRate](http://docs.google.com/classsf_1_1SoundBuffer.htm#a8cdfaea2ad1d05f81fa67442566c166e), [getDuration](http://docs.google.com/classsf_1_1SoundBuffer.htm#aee681c7a0f3dff4c4d0c9f8bbdb51bb0)

| [Time](http://docs.google.com/classsf_1_1Time.htm) sf::SoundBuffer::getDuration | ( |  | ) | const |
| --- | --- | --- | --- | --- |

Get the total duration of the sound.

Returns[Sound](http://docs.google.com/classsf_1_1Sound.htm) duration See Also[getSampleRate](http://docs.google.com/classsf_1_1SoundBuffer.htm#a8cdfaea2ad1d05f81fa67442566c166e), [getChannelCount](http://docs.google.com/classsf_1_1SoundBuffer.htm#a0a2890747db3811fb8d969d3e3abd0d1)

| std::size\_t sf::SoundBuffer::getSampleCount | ( |  | ) | const |
| --- | --- | --- | --- | --- |

Get the number of samples stored in the buffer.

The array of samples can be accessed with the [getSamples()](http://docs.google.com/classsf_1_1SoundBuffer.htm#a4ba0c1e5b5be500af42de30b1360eb2e) function.

ReturnsNumber of samples See Also[getSamples](http://docs.google.com/classsf_1_1SoundBuffer.htm#a4ba0c1e5b5be500af42de30b1360eb2e)

| unsigned int sf::SoundBuffer::getSampleRate | ( |  | ) | const |
| --- | --- | --- | --- | --- |

Get the sample rate of the sound.

The sample rate is the number of samples played per second. The higher, the better the quality (for example, 44100 samples/s is CD quality).

ReturnsSample rate (number of samples per second) See Also[getChannelCount](http://docs.google.com/classsf_1_1SoundBuffer.htm#a0a2890747db3811fb8d969d3e3abd0d1), [getDuration](http://docs.google.com/classsf_1_1SoundBuffer.htm#aee681c7a0f3dff4c4d0c9f8bbdb51bb0)

| const Int16\* sf::SoundBuffer::getSamples | ( |  | ) | const |
| --- | --- | --- | --- | --- |

Get the array of audio samples stored in the buffer.

The format of the returned samples is 16 bits signed integer (sf::Int16). The total number of samples in this array is given by the [getSampleCount()](http://docs.google.com/classsf_1_1SoundBuffer.htm#a2df1f1ae89213adee1494b95bb583f9d) function.

ReturnsRead-only pointer to the array of sound samples See Also[getSampleCount](http://docs.google.com/classsf_1_1SoundBuffer.htm#a2df1f1ae89213adee1494b95bb583f9d)

| bool sf::SoundBuffer::loadFromFile | ( | const std::string & | *filename* | ) |  |
| --- | --- | --- | --- | --- | --- |

Load the sound buffer from a file.

Here is a complete list of all the supported audio formats: ogg, wav, flac, aiff, au, raw, paf, svx, nist, voc, ircam, w64, mat4, mat5 pvf, htk, sds, avr, sd2, caf, wve, mpc2k, rf64.

Parameters

| filename | Path of the sound file to load |
| --- | --- |

ReturnsTrue if loading succeeded, false if it failed See Also[loadFromMemory](http://docs.google.com/classsf_1_1SoundBuffer.htm#af8cfa5599739a7edae69c5cba273d33f), [loadFromStream](http://docs.google.com/classsf_1_1SoundBuffer.htm#ad292156b1e01f6dabd4c0c277d5e079e), [loadFromSamples](http://docs.google.com/classsf_1_1SoundBuffer.htm#a63da986e144b578135edd48e51c565e8), [saveToFile](http://docs.google.com/classsf_1_1SoundBuffer.htm#ab2083dc1a934c64959d9e3f162328a76)

| bool sf::SoundBuffer::loadFromMemory | ( | const void \* | *data*, |
| --- | --- | --- | --- |
|  |  | std::size\_t | *sizeInBytes* |
|  | ) |  |  |

Load the sound buffer from a file in memory.

Here is a complete list of all the supported audio formats: ogg, wav, flac, aiff, au, raw, paf, svx, nist, voc, ircam, w64, mat4, mat5 pvf, htk, sds, avr, sd2, caf, wve, mpc2k, rf64.

Parameters

| data | Pointer to the file data in memory |
| --- | --- |
| sizeInBytes | Size of the data to load, in bytes |

ReturnsTrue if loading succeeded, false if it failed See Also[loadFromFile](http://docs.google.com/classsf_1_1SoundBuffer.htm#a2be6a8025c97eb622a7dff6cf2594394), [loadFromStream](http://docs.google.com/classsf_1_1SoundBuffer.htm#ad292156b1e01f6dabd4c0c277d5e079e), [loadFromSamples](http://docs.google.com/classsf_1_1SoundBuffer.htm#a63da986e144b578135edd48e51c565e8)

| bool sf::SoundBuffer::loadFromSamples | ( | const Int16 \* | *samples*, |
| --- | --- | --- | --- |
|  |  | std::size\_t | *sampleCount*, |
|  |  | unsigned int | *channelCount*, |
|  |  | unsigned int | *sampleRate* |
|  | ) |  |  |

Load the sound buffer from an array of audio samples.

The assumed format of the audio samples is 16 bits signed integer (sf::Int16).

Parameters

| samples | Pointer to the array of samples in memory |
| --- | --- |
| sampleCount | Number of samples in the array |
| channelCount | Number of channels (1 = mono, 2 = stereo, ...) |
| sampleRate | Sample rate (number of samples to play per second) |

ReturnsTrue if loading succeeded, false if it failed See Also[loadFromFile](http://docs.google.com/classsf_1_1SoundBuffer.htm#a2be6a8025c97eb622a7dff6cf2594394), [loadFromMemory](http://docs.google.com/classsf_1_1SoundBuffer.htm#af8cfa5599739a7edae69c5cba273d33f), [saveToFile](http://docs.google.com/classsf_1_1SoundBuffer.htm#ab2083dc1a934c64959d9e3f162328a76)

| bool sf::SoundBuffer::loadFromStream | ( | [InputStream](http://docs.google.com/classsf_1_1InputStream.htm) & | *stream* | ) |  |
| --- | --- | --- | --- | --- | --- |

Load the sound buffer from a custom stream.

Here is a complete list of all the supported audio formats: ogg, wav, flac, aiff, au, raw, paf, svx, nist, voc, ircam, w64, mat4, mat5 pvf, htk, sds, avr, sd2, caf, wve, mpc2k, rf64.

Parameters

| stream | Source stream to read from |
| --- | --- |

ReturnsTrue if loading succeeded, false if it failed See Also[loadFromFile](http://docs.google.com/classsf_1_1SoundBuffer.htm#a2be6a8025c97eb622a7dff6cf2594394), [loadFromMemory](http://docs.google.com/classsf_1_1SoundBuffer.htm#af8cfa5599739a7edae69c5cba273d33f), [loadFromSamples](http://docs.google.com/classsf_1_1SoundBuffer.htm#a63da986e144b578135edd48e51c565e8)

| [SoundBuffer](http://docs.google.com/classsf_1_1SoundBuffer.htm)& sf::SoundBuffer::operator= | ( | const [SoundBuffer](http://docs.google.com/classsf_1_1SoundBuffer.htm) & | *right* | ) |  |
| --- | --- | --- | --- | --- | --- |

Overload of assignment operator.

Parameters

| right | Instance to assign |
| --- | --- |

ReturnsReference to self

| bool sf::SoundBuffer::saveToFile | ( | const std::string & | *filename* | ) | const |
| --- | --- | --- | --- | --- | --- |

Save the sound buffer to an audio file.

Here is a complete list of all the supported audio formats: ogg, wav, flac, aiff, au, raw, paf, svx, nist, voc, ircam, w64, mat4, mat5 pvf, htk, sds, avr, sd2, caf, wve, mpc2k, rf64.

Parameters

| filename | Path of the sound file to write |
| --- | --- |

ReturnsTrue if saving succeeded, false if it failed See Also[loadFromFile](http://docs.google.com/classsf_1_1SoundBuffer.htm#a2be6a8025c97eb622a7dff6cf2594394), [loadFromMemory](http://docs.google.com/classsf_1_1SoundBuffer.htm#af8cfa5599739a7edae69c5cba273d33f), [loadFromSamples](http://docs.google.com/classsf_1_1SoundBuffer.htm#a63da986e144b578135edd48e51c565e8)

The documentation for this class was generated from the following file:

* [SoundBuffer.hpp](http://docs.google.com/SoundBuffer_8hpp_source.htm)

Copyright � Laurent Gomila  ::  Documentation generated by [doxygen](http://www.doxygen.org/)  ::