

libsound.so Documentation

Michail Andreas Michalopoulos
m.a.michalopoulos@student.tue.nl

May 28, 2024

Contents

1	Introduction	1
2	Library Compilation	1
2.1	Generating Object Files	1
2.2	Creating the Dynamic Library	2
3	Library Linking	2
4	Library Usage	2
5	Example	3

1 Introduction

This document provides a concise guide on how to compile and link the libsound.so Sound library in a C++ program. The Sound library is a custom library that provides audio functionality. It is a simple script that passes terminal commands using the ALSA library, which is built into most Linux distributions.

2 Library Compilation

2.1 Generating Object Files

First, we need to compile the Sound library source files into object files. This is done using the `g++` compiler with the `-c` option. The `-fPIC` option is used to generate position-independent code, which is necessary for dynamic libraries. For example, to compile ‘Sound.cpp’ into an object file, use the following command:

```
g++ -c -fPIC Sound.cpp -o Sound.o
```

2.2 Creating the Dynamic Library

Next, we create a dynamic library from the object files. This is done using the `g++` compiler with the `-shared` option. The `-o` option is used to specify the output file name. For example, to create a dynamic library named `libsound.so` from `Sound.o`, use the following command:

```
g++ -shared -o libsound.so Sound.o
```

3 Library Linking

To link the Sound library to your C++ program, use the `g++` compiler with the `-L` option followed by the directory where the library is located and the `-l` option followed by the library name (without the `'lib'` prefix and the `'.so'` extension). The `-o` option is used to specify the output file name. For example, to compile a program in a file named `main.cpp` and link the Sound library, use the following command:

```
g++ filename.cpp path\libsound.so -o filename
```

4 Library Usage

To use the Sound library in your C++ program, include the `'Sound.h'` header file at the top of your source file:

```
#include "Sound.h"
```

The library provides the following functions:

- `'playSound(soundfile)'`: Plays the sound file specified by `'soundfile'`, which is a string containing the path to the sound file.
- `'alsahelp()'`: Displays help information for the ALSA sound system.
- `'showsounddevice()'`: Displays information about the sound device.
- `'showdocumentation()'`: Displays this documentation.

You can then call the functions from the Sound library in your program. For example, if the Sound library has a function named `'playSound'`, you can call it like this:

```
playSound("example.wav");
```

Sound file selection should be handled in the main code since the library is just for playing the sound. The recommended method would be to make a configuration file with the paths to the different sound files and just call the `'playSound'` function with the path to the sound file.

5 Example

Here's a simple example of how you can use the library:

```
#include "Sound.h"

int main() {
    playSound("/path/to/soundfile.wav");
    return 0;
}
```