

MusicFormats API guide

<https://github.com/jacques-menu/musicformats>

vv0.9.68 – May 2, 2023

Jacques Menu

Minimal score



```
1 S_lpsrScore translateMsrToLpsr (  
2     const S_msrScore&          originalMsrScore,  
3     const S_msrOahGroup&       msrOpts,  
4     const S_lpsrOahGroup&      lpsrOpts,  
5     mfPassIDKind               passIDKind,  
6     std::string                passDescription,  
7     const S_mfcMultiComponent& multiComponent)  
8 {  
9     // ... ..  
10  
11     // create an msr2lpsrTranslator  
12     msr2lpsrTranslator  
13         translator (  
14             originalMsrScore);  
15  
16     // build the LPSR score  
17     S_lpsrScore  
18         resultingLpsr =  
19         translator.translateMsrToLpsr (  
20             originalMsrScore,  
21             multiComponent);  
22  
23     // ... ..  
24  
25     return resultingLpsr;  
26 }
```

Contents

| | | |
|------------|---|-----------|
| 1 | Acknowledgements | 3 |
| I | MusicFormats API principles | 4 |
| 2 | API principles | 5 |
| II | The MXSR API | 6 |
| 3 | Creating scores with the MXSR API. | 7 |
| III | The MSR API | 8 |
| 4 | Creating scores with the MSR API | 9 |
| IV | The LPSR API | 10 |
| 5 | Creating scores with the LPSR API | 11 |
| V | The BSR API | 12 |
| 6 | Creating scores with the BSR API. | 13 |
| VI | Indexes | 14 |

List of Figures

Chapter 1

Acknowledgements

Many thanks to Dominique Fober, the designer and maintainer of the `libmusicxml2` library. This author would not have attempted to work on a MusicXML to LilyPond converter without his work being already available.

In particular, the conversion of MusicXML data to a tree is extremely well done directly from the MusicXML DTD, and that was a necessary step to produce LilyPond code. Dominique also provided a nice way to browse this tree with a two-phase visitor design pattern, which this author uses extensively in his own code. The interested reader can find information about that in [libmusicxml2.pdf](#), and more technical details in [MusicFormatsMaintenanceGuide.pdf](#).

`xml2ly` and some of the specific examples presented in this document started as this author's contribution to `libmusicxml2`, and was later moved to a separate GitHub repository for practical reasons.

Part I

MusicFormats API principles

Chapter 2

API principles

Part II

The MXSR API

Chapter 3

Creating scores with the MXSR API

Part III

The MSR API

Chapter 4

Creating scores with the MSR API

Part IV

The LPSR API

Chapter 5

Creating scores with the LPSR API

Part V

The BSR API

Chapter 6

Creating scores with the BSR API

Part VI

Indexes

Main index

| | | |
|-----------------------|--|-------------------|
| A | | |
| API | | 1 |
| D | | |
| Dominique Fober | | 3 |
| DTD | | 3 |
| L | | |
| libmusicxml2 | | 3 |
| LilyPond | | 3 |
| M | | |
| MusicFormats | | 1 |
| MusicXML | | 3 |