

Maintainer's guide to xml2ly

Jacques Menu

December 21, 2019 version

Abstract

This document presents the design principles behind xml2ly, as well as the way to use it.

All the examples mentioned can be downloaded from <https://github.com/grame-cncm/libmusicxml/tree/lilypond/files/samples/musicxml>. They are grouped by subject in sub-directories, such as [basic/HelloWorld.xml](#).

1 Acknowledgements

The scores fragments shown in this document have been produced by translating the '.xml' files to LilyPond syntax, and then creating the graphical score with LilyPond.

The translations have been done by xml2ly, a prototype tool developed by this author. xml2ly and some of the specific examples presented in this document are this author's contribution to libmusicxml2, an open-source C++ library created and maintained by Dominique Fober at Grame, Lyon, France. The home page to libmusicxml2 is <https://github.com/grame-cncm/libmusicxml>.

The reader is invited to handle the '.xml' file examples with their own software tools to compare the results with the ones herein.

Tests with other score editing applications are mentioned in this document, namely SibeliusTM, FinaleTM and MuseScore, which is open-source. musicxml2ly is mentioned too: this translator is supplied with LilyPond. This author doesn't own licenses for other commercial applications such as DoricoTM or CapellaTM.

2 Overview of xml2ly

2.1 Why xml2ly?

MusicXML (*Music eXtended Markup Language*) is a specification language meant to represent music scores by texts, readable both by humans and computers. It has been designed by the W3C Music Notation Community Group (<https://www.w3.org/community/music-notation/>) to help sharing music score files between applications, through export and import mechanisms.

The homepage to MusicXML is <https://www.musicxml.com>.

MusicXML data contains very detailed information about the music score, and it is quite verbose by nature. This makes creating such data by hand quite difficult, and this is done by applications actually.

2.2 What xml2ly does

Listings

Contents