

This document is part of openLilyLib¹,
a collection of resources for the LilyPond notation software²
and the \LaTeX typesetting system.

Excerpt from:
The openLilyLib Tutorials

Songbooks with LilyPond and \LaTeX

Jeremy Boor

July 18, 2013

¹<http://www.openlilylib.org>

²<http://www.lilypond.org>

Copyright © 2012–13 Urs Liska and others

openLilyLib is a collaborative and free software and documentation project. All contributions are copyright by their attributed authors.

If not stated otherwise all creative content distributed by the openLilyLib project is licensed under the Creative Commons Attribution-ShareAlike 3.0 Unported License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-sa/3.0/>.

All software that is part of openLilyLib is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This explicitly also applies for any code examples that may be part of this manual.

This program or source code is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. To view a copy of this license, visit <http://www.gnu.org/licenses/gpl.html>.

Contents

1	Songbooks	5
1.1	Preparing Your LilyPond Scores	5
1.2	Using <code>\includepdf</code>	5
1.3	Getting Your Scores into the Table of Contents	6
1.4	Getting LilyPond Scores and \LaTeX Page-numbers out of Each Other's Way	7

Original author: Jeremy Boor | Contributors: None yet

Abstract

There are different ways to combine text and music to books. This tutorial describes step by step how to compile a songbook consisting of textual parts and complete scores using \LaTeX and LilyPond.

1 Songbooks

A songbook is a book that contains pieces of music, but also contains various bits of text. It may include a general introduction, and specific introductions or performance notes for each piece. It should have frontmatter, including a table of contents that lists all of the pieces of music included in the book.

LilyPond provides a python script called `lilypond-book` that is designed to incorporate LilyPond output into \LaTeX documents. However, `lilypond-book` is best suited for musicological documents, wherein musical examples are presented for consideration rather than for performance. `lilypond-book` is not designed to preserve all of the format and layout decisions that make LilyPond scores excellent for musical use, such as system length and between-system spacing.

In a songbook, the musical scores should be given an optimal presentation for use as musical scores, while the text should be given an optimal presentation for reading. Attempts to use `lilypond-book` for songbook production result in an unfruitful opposition between \LaTeX 's layout procedures and those of LilyPond.

The best situation for producing a songbook with LilyPond and \LaTeX is one in which \LaTeX 's layout and formatting are simply suspended while the musical score is inserted into the final output. To this end we will use `\includepdf`.

1.1 Preparing Your LilyPond Scores

- LilyPond scores must be compiled and the resulting PDFs must be to hand before we can do anything with \LaTeX to turn them into a songbook. It will be easiest if we keep our finished LilyPond scores in the same directory as our \LaTeX source, so as not to have to type long paths when we call the PDFs into \LaTeX .
- Since we want our LilyPond scores to be separable within the songbook, we should compile them separately, producing one PDF for each piece of music.
- Since the page-numbering must be done by \LaTeX , we should turn off page numbering in our LilyPond source. This is done in the `\paper` block:

```
\paper{ print-page-number = ##f }
```

1.2 Using `\includepdf`

The `\includepdf` command is part of the `pdfpages` package. You must have this package installed and have `\usepackagepdfpages` in the preamble of your \LaTeX document.

By default, `\includepdf` includes only the first page of the PDF. To include all of your PDF score, your basic use of `\includepdf` should look like this:

```
\includepdf[pages=-]{path/to/myScore.pdf}
```

This use of `\includepdf` brings your LilyPond scores into the flow of your \LaTeX output. Each score will begin on its own page, and regular \LaTeX business will follow on the the page that follows the last page of your PDF.

Also by default, `\includepdf` removes all headers and footers supplied by \LaTeX . This includes page-numbering. In order to restore page-numbering to your music pages, include this line in the body of your \LaTeX document:

```
\includepdfset{pagecommand=\thispagestyle{plain}}
```

This sets the `pagestyle` to `plain` for all PDF pages. The `plain` `pagestyle` provides a page number at the bottom center of each page, but no headers. It is likely that the placement of the page number will interfere with your LilyPond score. We shall deal with this issue later.

1.3 Getting Your Scores into the Table of Contents

Your songbook probably needs a table of contents, and you will want your scores to be listed. The regular \LaTeX `\section` and `\chapter` commands provide a TOC listing, but they also print a numbered title above the section to which they refer. This will not be appropriate for scores, since the scores already have their own titles produced by LilyPond. To provide a TOC listing without printing a title, you can use:

```
\addcontentsline{toc}{section}{The Name of a Piece of Music}
```

This should be immediately before the `\includepdf` command. So now each inclusion of a piece of music in your book looks something like this:

```
\addcontentsline{toc}{section}{The Name of a Piece of Music}
\includepdf[pages=-]{path/to/thePDFscoreOfThatPiece.pdf}
```

Note that `\section` is used when your \LaTeX document uses the `article` class, while `\chapter` should be substituted if you are making a book.

These PDF-inclusions can be interspersed with regular `\section` or `\chapter` commands. This is useful if you want to provide introductions to your pieces or other textual interludes. Of course, if the introduction has the same title as your piece, then you should omit the `\addcontentsline` for that piece.

By default, `\section` and `\chapter` are numbered, both in their titles and in their TOC entries. `\addcontentsline` entries are not numbered. To turn off numbering for all TOC entries for the sake of uniformity, include this line the body of your \LaTeX document:

```
\setcounter{secnumdepth}{-1}
```

With `\chapter/section` numbering turned off, you can freely intersperse textual sections with scores however you please:

```
[frame=single]
\section{Excellent Song}
The introduction to ‘‘Excellent Song’’ goes like this.
It elaborates upon the excellence of ‘‘Excellent Song.’’

% We don't need a second TOC entry
% after the introduction, so straight to:

\includepdf[pages=-]{path/to/ExcellentSong.pdf}

% The next song doesn't need an introduction, so:
\addcontentsline{toc}{section}{Mediocre Song}
\includepdf[pages=-]{path/to/MediocreSong.pdf}
\addcontentsline{toc}{section}{Another Song}
\includepdf[pages=-]{path/to/anotherSong.pdf}

\section{Verbose Preamble to the Next Bit}
```

Please note that \LaTeX must compile your file at least *twice* in order to get the TOC right.

1.4 Getting LilyPond Scores and \LaTeX Page-numbers out of Each Other's Way

We have page numbers on our scores, but they're in the wrong place with regard to the score. The way that we fix this will depend on the final dimensions of the book we intend to produce, and will be affected by our manipulation of layout variables in both LilyPond and \LaTeX . Basically, we need enough room for everything. We need the scores to stop in time to leave enough room for the page number, and we also need the page number to be low (or high) enough that the score doesn't need to stop (or start) in a silly-looking place. Fortunately, the page-numbers can be moved around a bit in \LaTeX without mucking up the rest of the layout.

In \LaTeX we can use both the geometry package (which will be necessary anyway if our book is an odd size) and the `\setlength` command. We will be primarily interested in the length of the `\footskip` variable, which determines how far away the footer is from the body of the text.

In LilyPond, we need to work with the `\paper` block. Assuming that we want to leave out page numbers on the bottom, we will most likely be interested in adjusting the bottom-margin and the foot-separation.

If the book is limited to a standard paper size like A4 or US-letter, the desired affect can be achieved simply by making LilyPond's bottom margin slightly wider, and \LaTeX 's `\footskip` value somewhat larger.

If the dimensions of the book allow for it, we might also be interested in moving the page numbers to the top of the page on alternate sides, as happens by default in LilyPond. This can be done with the help of the `fancyhdr` package, which allows the redefinition of pagestyles.

Using `fancyhdr`, we can also apply a different pagestyle to the text-only pages of our book. This can be useful if we want the headers of our pages to have fancy things on them like head-rules and chapter names. To do this, we set the pagestyle of the document to something other than style given to the score pages in the `\includepdfset{pagecommand=\thispagestyle{plain}}` command. So the body of our \LaTeX document might contain the command `\pagestyle{fancy}`. This setting will be overridden and the pagestyle replaced whenever `\includepdf` is invoked.