Converting Noteworthy songs to LilyPond in Windows

I have adapted the NWC2LY program originally written by Mike Wiering (see http://nwc2ly.sourceforge.net/) by porting it into C# and then adding some new features and correcting some bugs. My program can be used in 2 ways:

1. As a Noteworthy "User Tool" to convert a single stave of Noteworthy music to LilyPond, and
2. As a standalone program that converts an entire Noteworthy score to a complete LilyPond score.

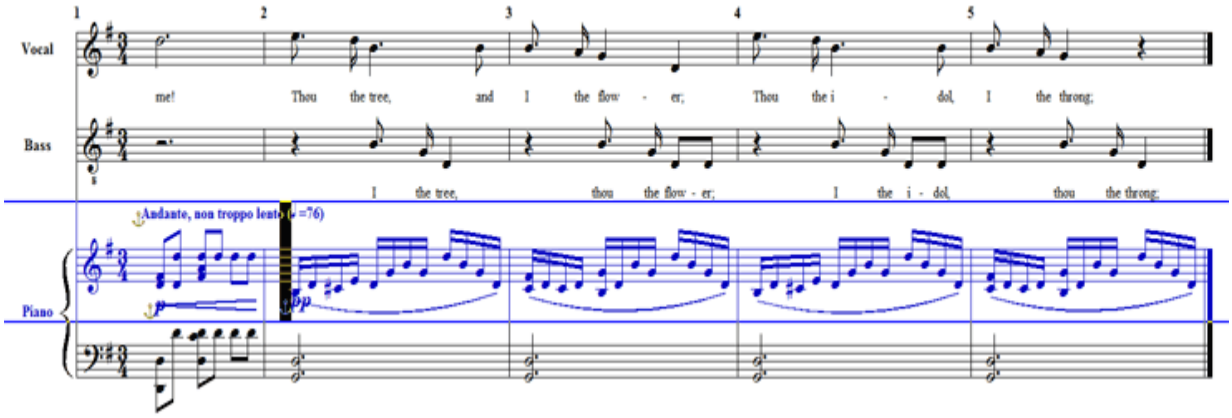
Please let me know of bugs/enhancement requests at the Noteworthy Forum (http://my.noteworthysoftware.com/).

All the files mentioned in these instructions can be found at http://www.holmessoft.co.uk/homepage/software/nwc2ly/

This brief manual concentrates on the 2nd mode, although all of the features that work on individual staves of Noteworthy will also work in the 1st mode.

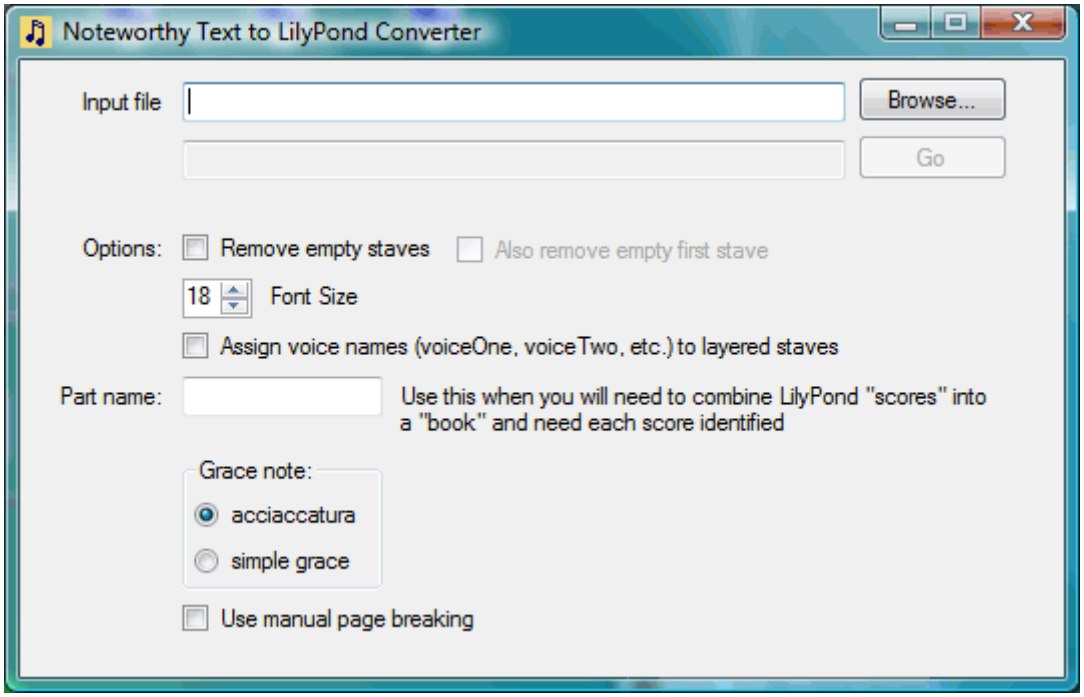
Simple Use

At its simplest, we can use the program to convert a song for 2 people and piano to LilyPond source that will typeset the page directly. I've used a small extract from a Gilbert and Sullivan show called Iolanthe to show this.



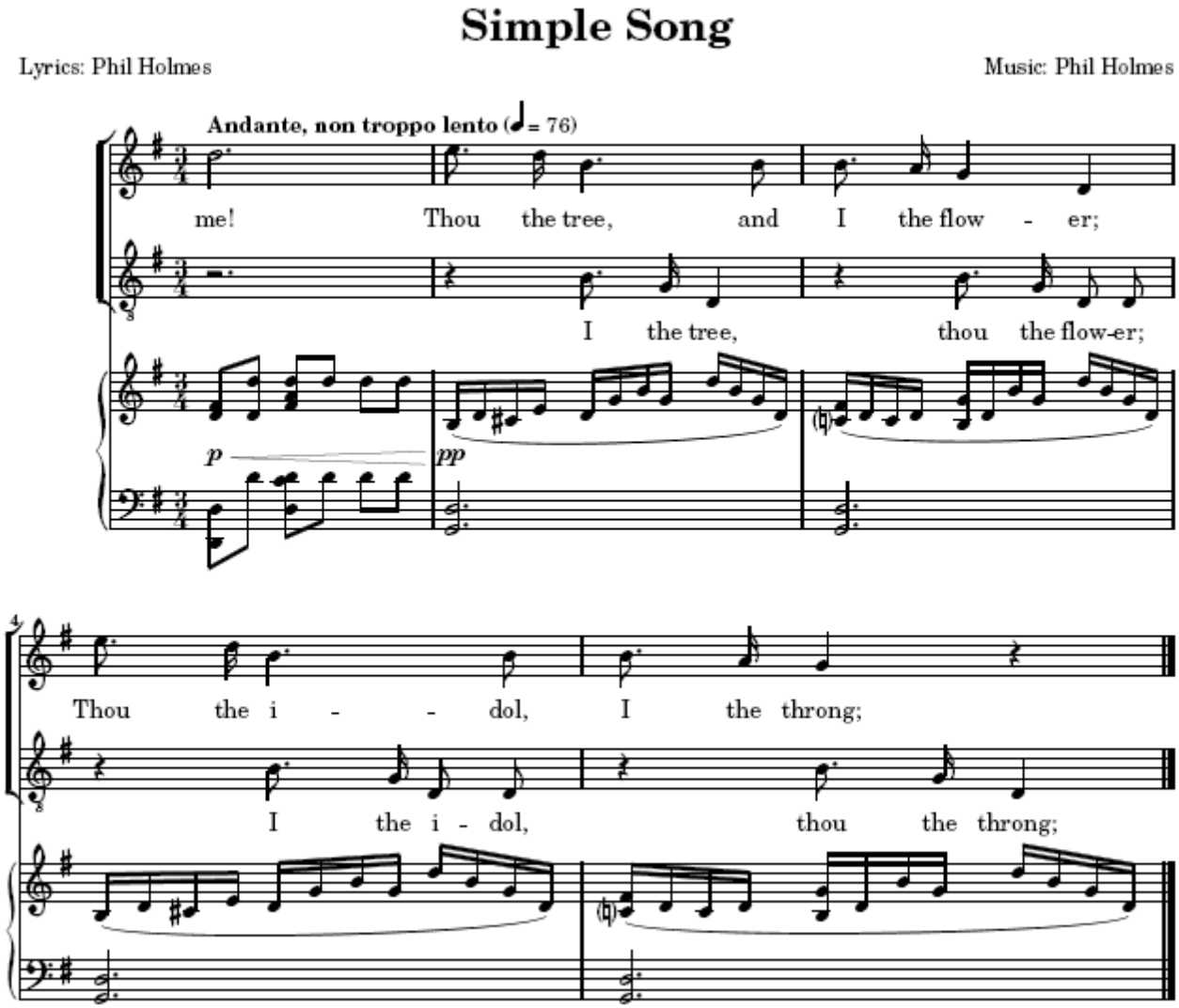
Then Export it as a Noteworthy Text file. If you've not already downloaded my program, you need to download two separate executable files - NWCTXT2Ly.exe and nwc2ly.exe. It doesn't require installation - just save the files somewhere where you can access them easily (both in the same directory) and run the NWCTxt2Ly file to run the program that converts a complete score..

Running NWCTXT2Ly should produce a window similar to the one below:



Click the Browse button and locate the NWCTXT file you've just exported. Ignore the other options for now and just click the Go button. It should tell you that you've written 4 staves with 4 voices. If you now look in the folder/directory where the NWCTXT file was stored, you'll see about 11 new files - some with a .ly extension and others with a .nwcextract extension. The .ly files are LilyPond files and are used to create the score. The .nwcextract files are just used by my program during its processing. If you're keen on housekeeping, you can delete them.

Assuming you have already installed LilyPond from its download site (http://lilypond.org/install/), double clicking the Song1Iolanthe.ly file (the same name as the NWCTXT file, but a different extension) should mean that nothing appears to happen for a while as LilyPond does its processing, then a DOS window will open briefly and you should see a new log file, a PostScript file and a PDF file in the directory. Double click the PDF file to open it, and you should have your first LilyPond score, looking like this:



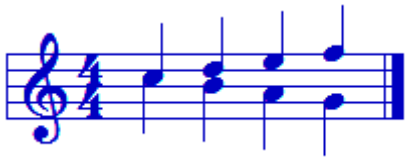
Some things to note from this:

- The piano part is shown as piano staves. The program assigns piano staves to any 2 staves which have the visual style "Upper Grand Staff" and "Lower Grand Staff". The singers have Choir staves assigned to them - this is the default for any stave with the visual style "Standard". Note that the bar lines are not joined. If the staves had the visual style "Orchestral" they would be shown as an orchestral part with joined bar lines.
- The title and copyright information is taken from the Noteworthy Info - I suggest you change the copyright notice, since the © character does not translate well into PDF.
- All the dynamic marks and changes are correctly shown - for the piano they sit below the Right Hand part, and for the score to look best any dynamics on other piano staves should be set not to display. Ditto tempo marks.
- Barring and slurring is carried over correctly - in fact - everything in this file is rendered accurately with no further work required.

So that's the very simplest song. Let's look at what happens when there are different voices represented.

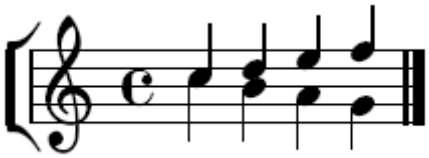
Voices

Here is a very simple tune with 2 voices:



Download the Noteworthy file, then export it as an NWCTXT file.

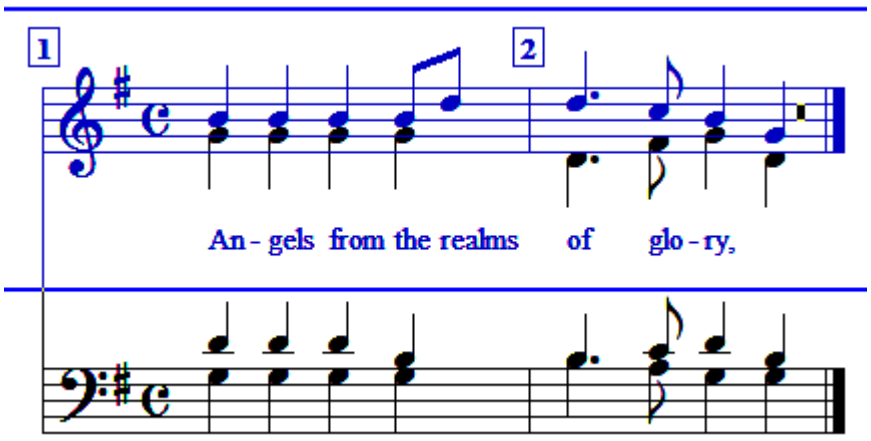
The two voices are created by forcing up-stems for the upper part, and down-stems for the lower part, and entering the notes as chords. Running NWCTXT2Ly against the exported file, and then double clicking the simplevoice.ly file as before, again gives a PDF file which looks like this:



So that worked without any special work either.

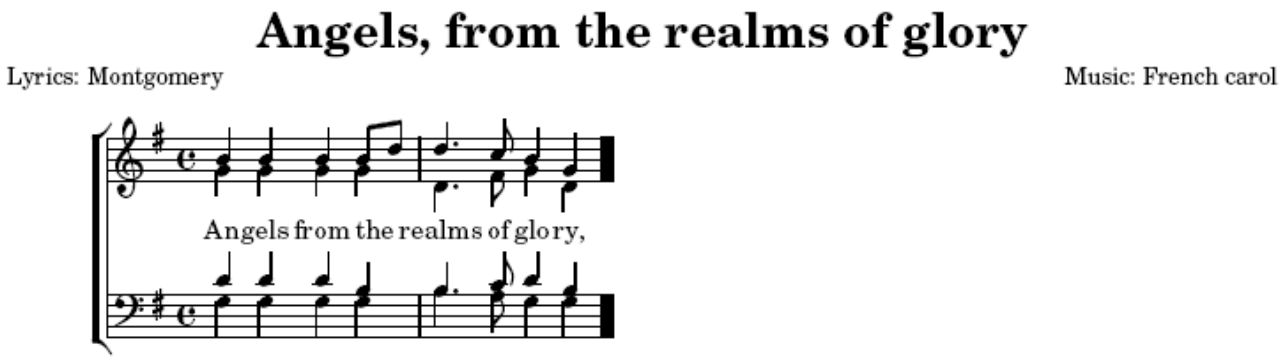
Mostly, Noteworthy uses layering to create multiple voice scores as above, so let's look at this.

I've done a short extract from a Christmas carol in Noteworthy - the file is here and it looks like this:

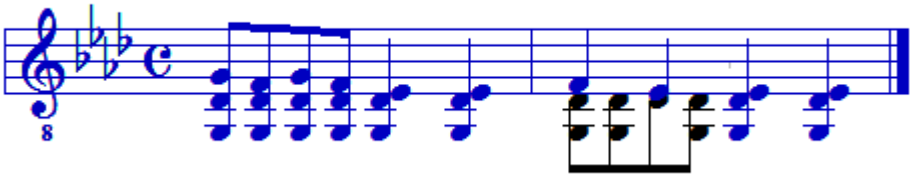


If we export this and simply run the NWCTXT2Ly program, then convert the output to LilyPond, the results look horrible with overlapping stems, and warnings in the LilyPond

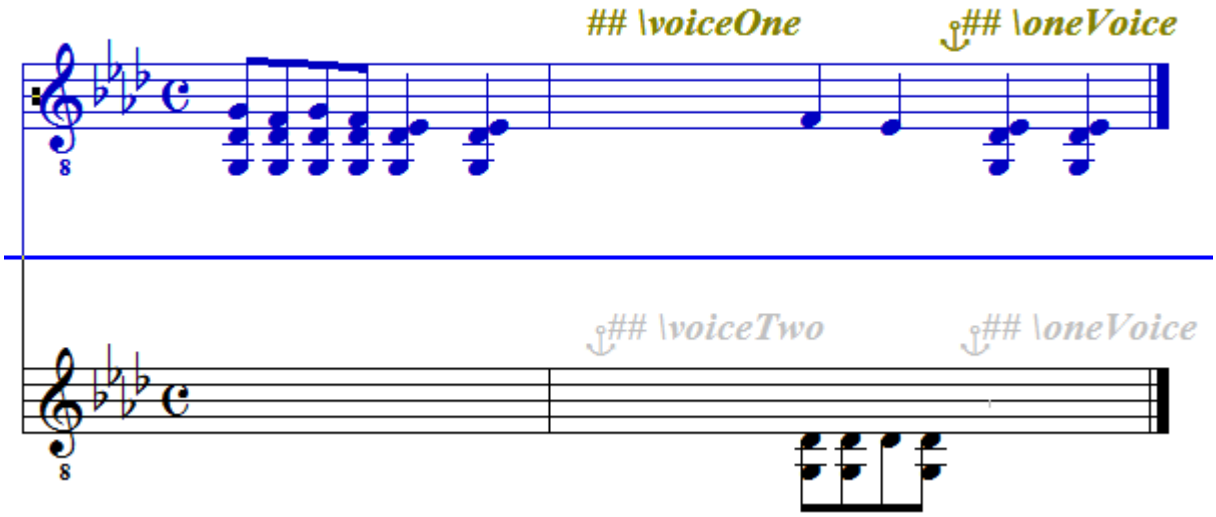
logfile saying "warning: ignoring too many clashing note columns". However, clicking the option to "assign voice names" means that my program assigns the top staves of the layers the voice name "voiceOne" and the lower staves "voiceTwo". LilyPond then knows that voice one has its stems up, and voice two the stems down, and therefore produces a PDF file looking like this:



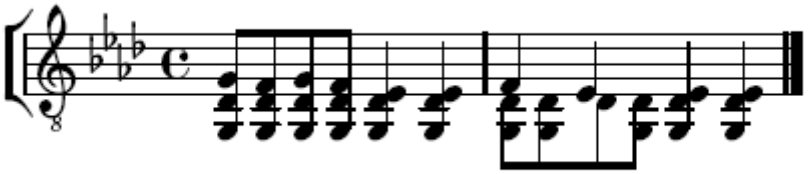
This approach will generally work well with hymns. However, we often don't want to force the upper voice to use all upward stems, as in the snippet below:



Here we have most of the notes of voice one stem up (owing to their position on the stave) then a second voice being added in bar 2 - these are stem down and voice 1 stem up (which they would be, no matter where they are on the stave). This is more difficult to arrange into LilyPond, and the way I've done it is to explicitly tell LilyPond which notes belong to which voice. If I remove the layering, you can see how it's done.



Where there are no notes in voice 2, we use muted notes with visibility set to never. When the section with 2 voices starts, we add hidden text beginning "##" to describe the voices - I have written my program to recognise that hidden text beginning ## is a special instruction to send the text direct to the LilyPond file. So we mark the start of voice one with \voiceOne (note the capitalisation) and the end with \oneVoice. The second voice is marked \voiceTwo as its start. It's important to put these flags in the correct place to get the optimum stem placement. Marking the voices like this, and then turning layering back on, gives the following LilyPond output (once we've remembered to uncheck the "assign voice names" option):



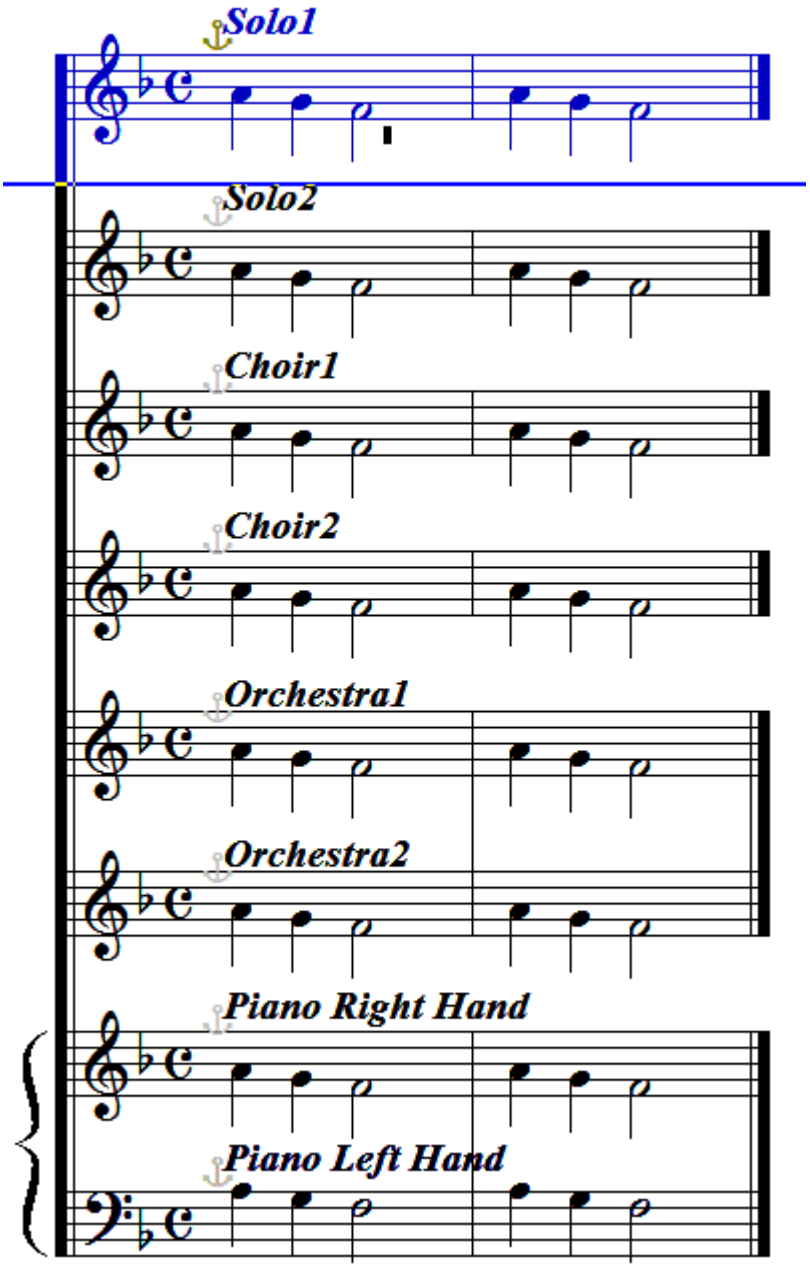
Which is again what was wanted.

As far as I'm aware, this ability to layer in Noteworthy and flag voice names to LilyPond makes it possible to export Noteworthy files, convert them to LilyPond and create a LilyPond score in all cases.

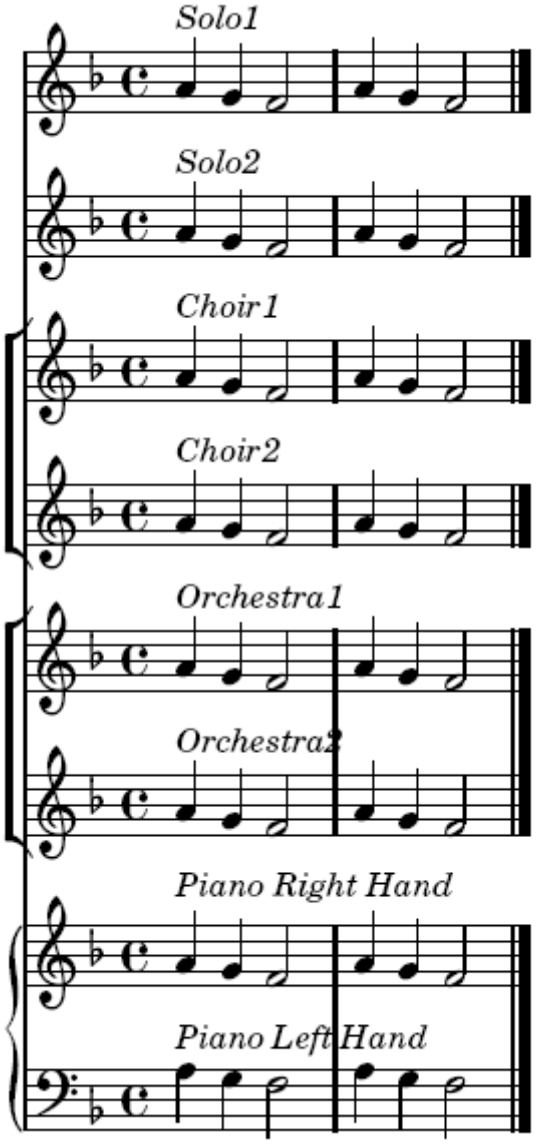
Next - lets go back and look again at how we can assign staff types.

Staffs

My program supports the generation of 4 staff types: piano, orchestral, choir and solo. The Noteworthy file illustrated below shows how this is supported.



Exporting this, translating it to LilyPond format and then typesetting it gives:

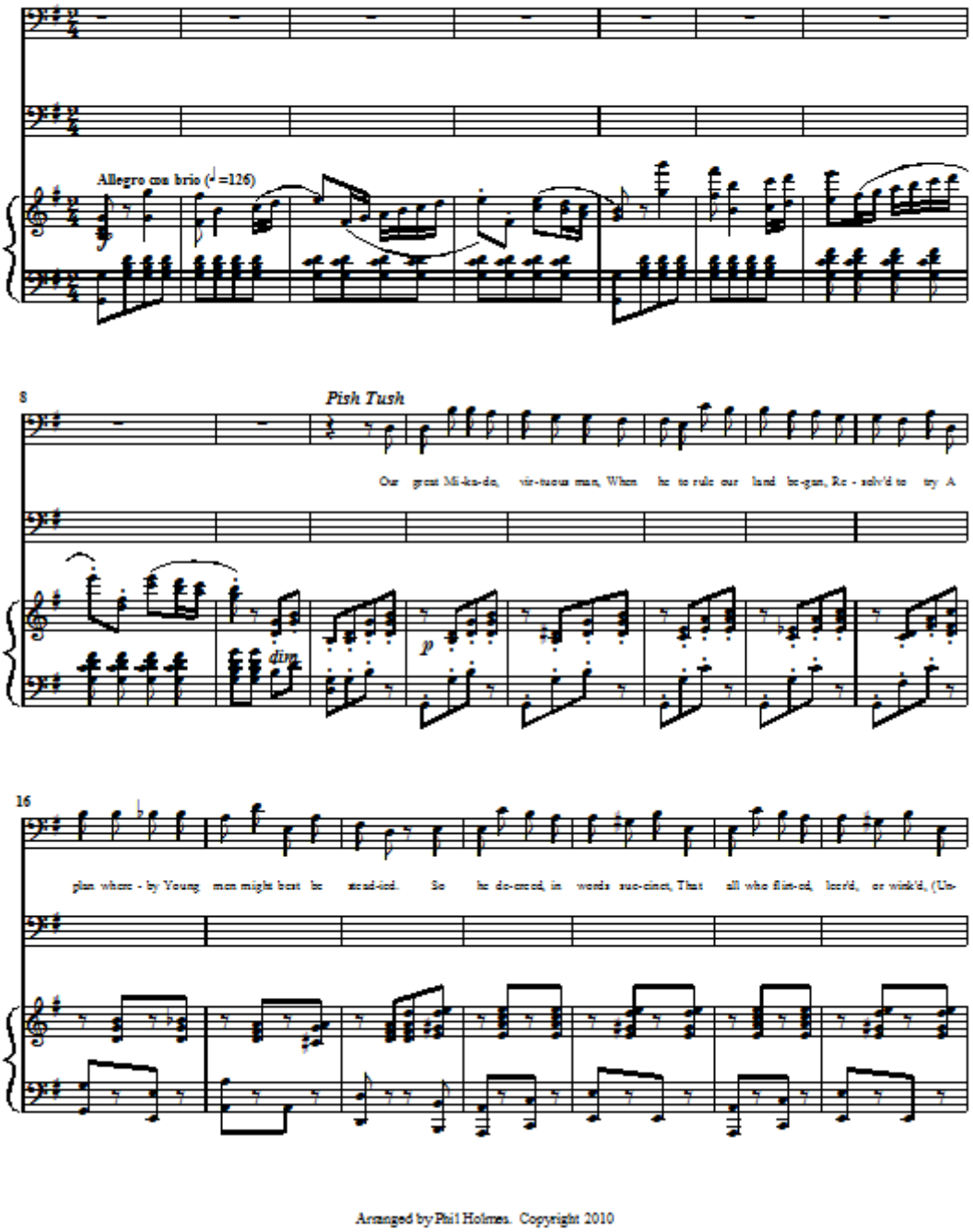


As we can see, at the top we have 2 staves that have no bracketing or linking, then 2 bracketed, unlinked choir staves, 2 bracketed, linked orchestra staves and the bottom piano piece. The way of achieving this was briefly mentioned earlier. Piano staves and orchestra staves are produced by setting the Staff Properties Visual Style to Upper Grand Staff/Lower Grand Staff for piano parts, and Orchestral for Orchestra style. All the vocal parts are set to Standard, and we discriminate the solo parts by setting their name to start with the word "Solo". In my example above, the top staff is named SoloStaff and the next one down is SoloStaff1. Simple.

The next question is how we ensure that LilyPond does not display empty staff lines. Below is the opening page of part of a Mikado score that I've arranged (clicking it downloads the whole score).

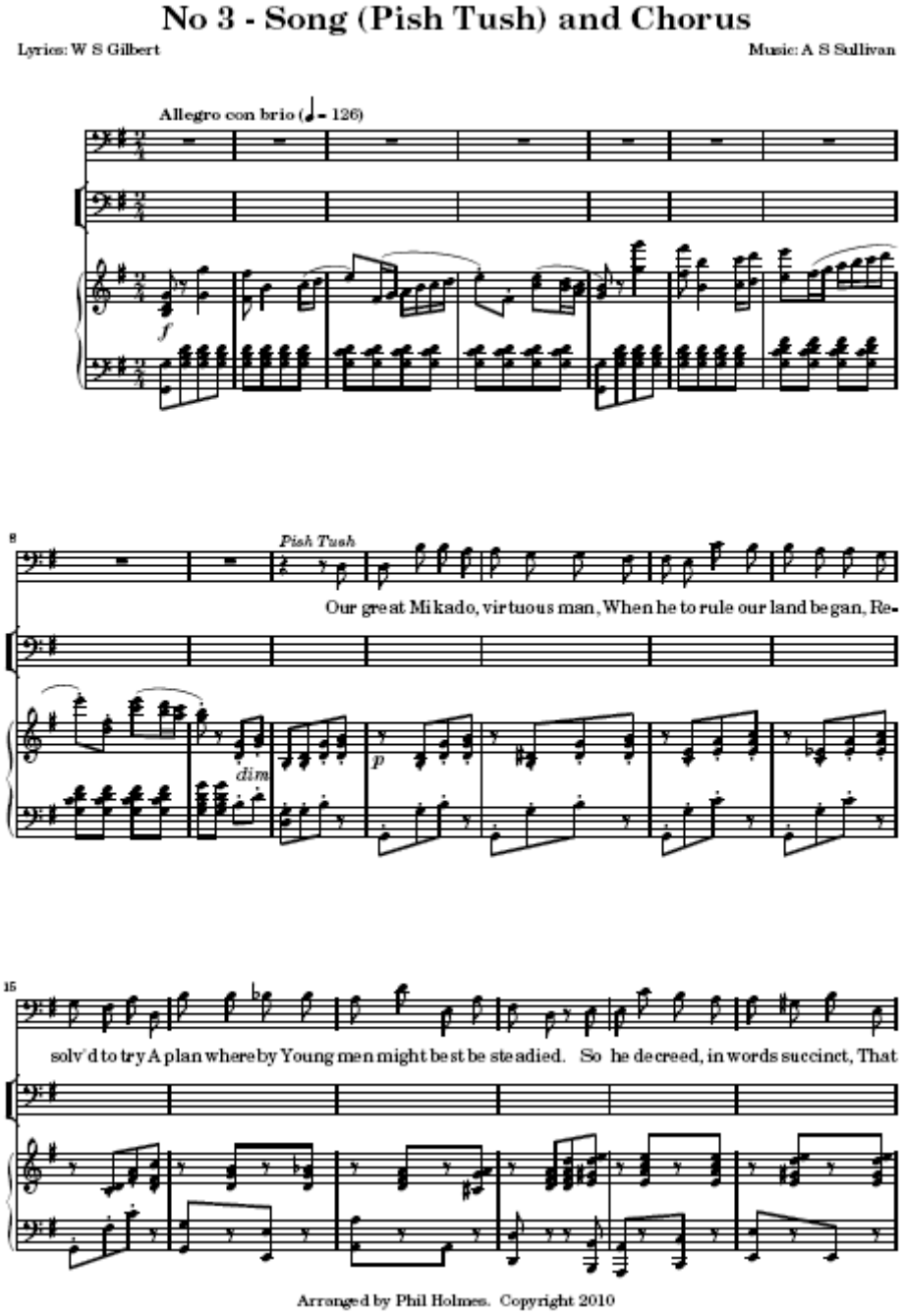
No 3 - Song (Pish Tush) and Chorus

W S Gilbert A S Sullivan

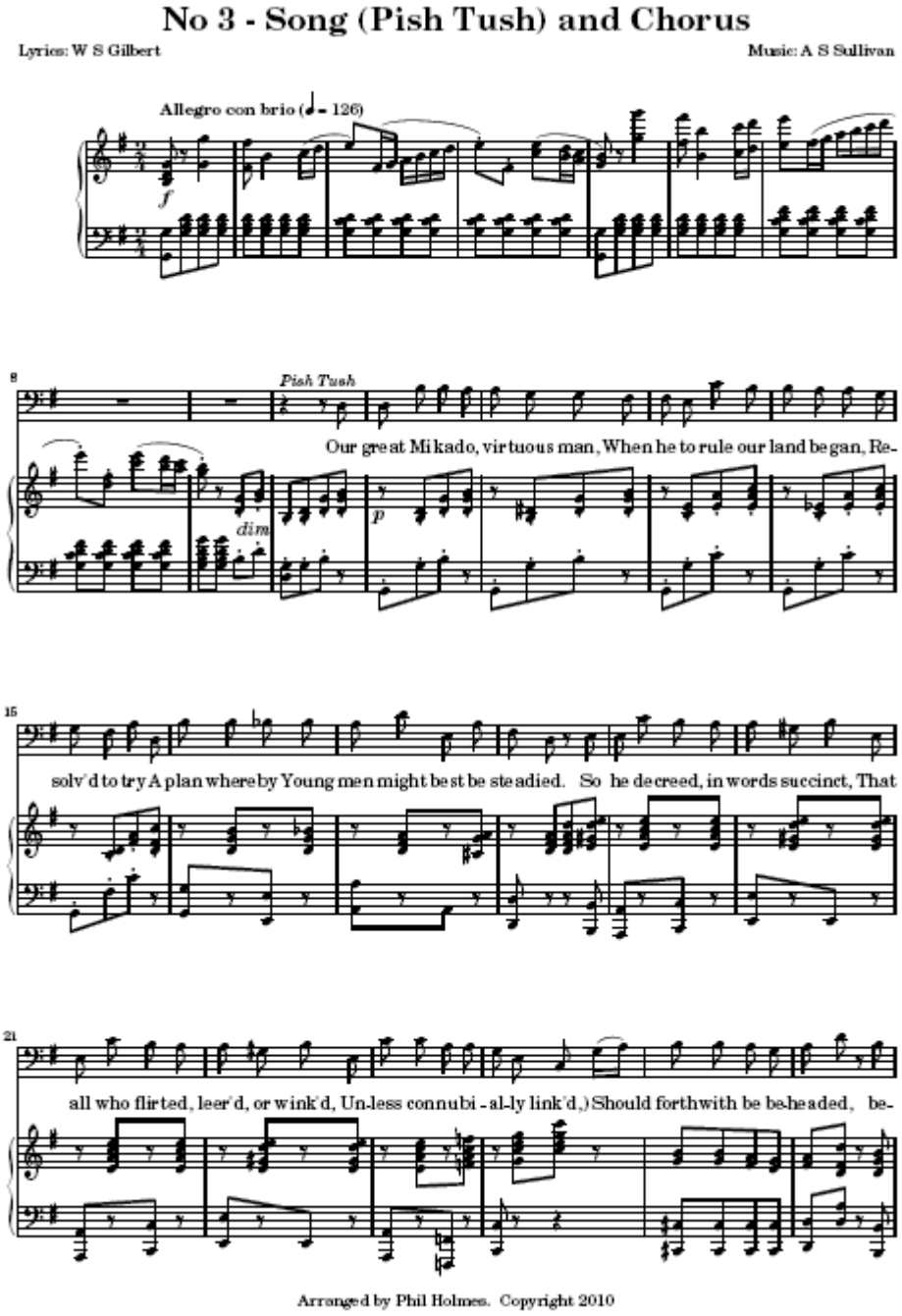


Arranged by Phil Holmes. Copyright 2010

If we do nothing else but export it, transate to LilyPond using the default settings and then typeset it (note - this is the first example of quite a large score and the typesetting takes some time - almost 30 seconds on my machine when nothing seems to happen) we get the page below:



As you see, we still have the empty staves. However, if we select the option "Remove empty staves" all bar the top empty staves are not displayed. If we also select "Also remove first empty stave" then all the empty ones are removed, as below. I always run with these options set.



That's better. Some notes on what counts as an "empty staff" for LilyPond:

1. Bars containing "whole bar rests". For my program, this counts as a bar with a duration of a semi-breve. Let's be clear - in a 2/2 bar, a minim rest would occupy the whole bar, but would not be translated as a "whole bar rest" and therefore would still display. Even for bars where a semi-breve isn't strictly a full bar, Noteworthy still treats it as occupying the bar and my program translates it to a "whole bar rest".
2. Bars contaning only "spacer rests" (in LilyPond notation). Spacer rests are translated from notes or rests with the display propery of "Never".

Other features

Font size

The global font size of the output file can be changed with the font size setting on NWCTXT2Ly

Part Name

If you're planning to make a LilyPond book, you will need all the individual LilyPond files in one place (to make it easy). However, if you name the staves from (say) two songs as "Bass", then these will be both be called Bass.ly. To overcome this, you can give them Part Names, like SongOne and SongTwo - the two files will then be called SongOneBass.ly and SongTwoBass.ly and can both be put into the same directory.

Grace Notes

As far as I'm aware, Noteworthy only supports a single type of grace note, whereas LilyPond supports a range. For a given song, you can select whether grace notes will be translated as grace notes or acciaccaturas. Grace notes work better where the note is already within a slur, since LilyPond does not support nested slurs.

Page and Line Breaking

By default, LilyPond organises line and page breaks by itself. However, you can force a line break by putting ##\break as hidden text on one of the stave. You can force a page break by putting ##\pageBreak as hidden text. Even if you do this, you may still find the LilyPond breaks where you don't want. To force it only to use manual page breaks, select the "Use manual page breaking".

Other LilyPond markup

Any markup that you can put directly into LilyPond you can enter using the "hidden text starting with ##" feature. For example, I frequently use:

- ##\arpeggio - put after a chord marks the chord with an arpeggio marking
- ##-^ - put after a note, marks the note with a "top hat" *marcato* marking
- ##\ottava #1 - starts a section marked with "Octave up". ##\ottava #0 ends it

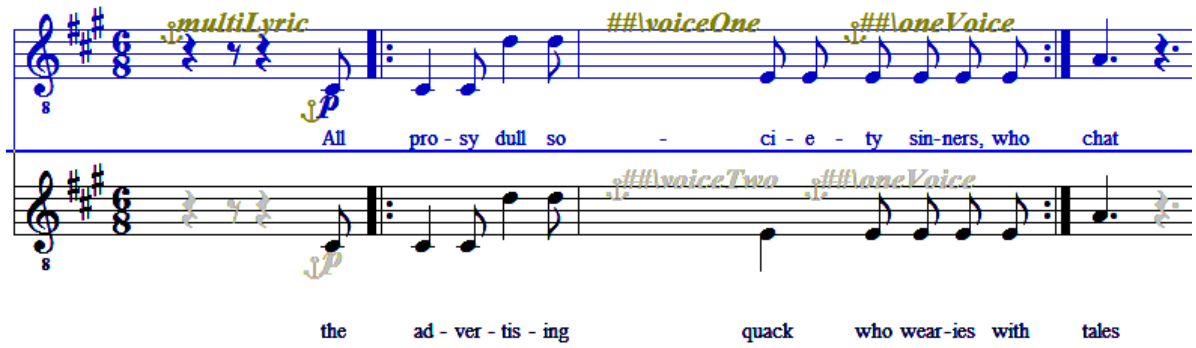
No doubt there are many others. Be careful that most markup that requires an apostrophe will not work, since the apostrophe is "escaped" with a slash during translation.

Tremolo

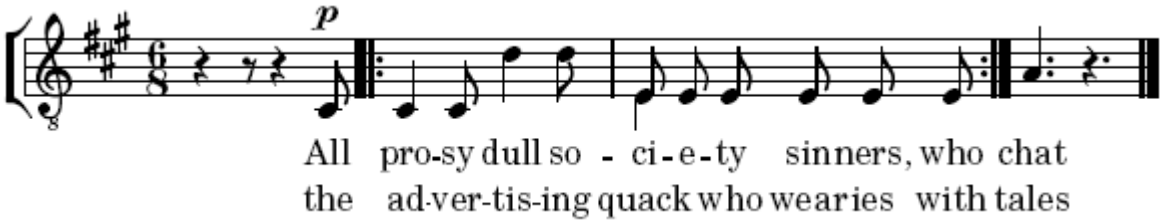
To export tremolo notes, mark the start of the tremolo notes with tremoloOn as hidden text (note the capitalisation) and the end of the tremolo notes with tremoloOff. The notes in the Noteworthy score should be appropriate for the time signature (e.g. 4 crochets per bar for 4/4 time) and these will be notated as 4 notes with semi-quaver tails in the LilyPond score.

Different lyric syllables on each verse

If you have lyrics where, say, the first verse has a single syllable fitting a crochet and the second has 2 syllables fitted to 2 quavers, it gets a bit complicated :-(. I've solved this by using multiple voices with music in each voice that are normally identical, only varying where the difference in syllables occurs. The frst verse lyrics go on voice one, and the second on voice two. Rests in voice 2 are hidden. We then mark voice one with the special hidden text "multiLyric". This suppresses warnings from LilyPond that there are clashing note stems - we know this, since the music is the same in the 2 voices! This method is illustrated in the image/.nwc file below - though remember that if you're typing this in rather than downloading it, you should layer the 2 staves. I've separated them for clarity.



(The Mikado strikes again). In Lilypond, we get the result below:

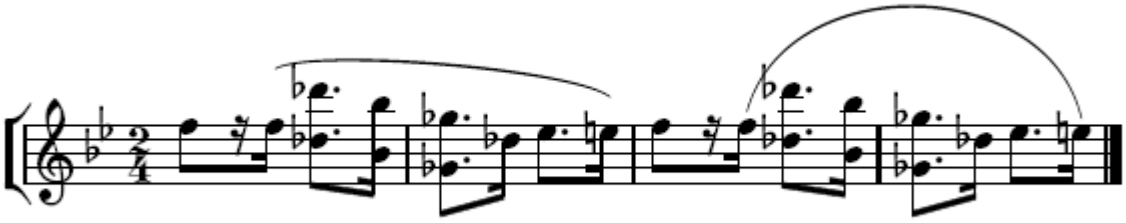


Which is what we wanted.

Slurs

Just occasionally, LilyPond takes an extreme approach to fitting a slur to the notes. An example is below:

I wanted to be able to control this, and LilyPond does have two mechanisms to allow finer control. I selected the simpler - over-riding the start and finish position. To do this, add setSlur(x,y) as hidden text just before the slur. X and Y should be numbers specifying how much the start and finish should be raised. You should also add setSlur(0,0) afterwards, to reset the slur position. Using setSlur(6,4) with the above music, I got:



with the result to the left being the adjusted slur and the one on the right the default. The one on the left works better for me.

NWC2Ly

It only remains to mention that the User Tool, NWC2Ly implements all the "within staff" features mentioned about NCWTXT2Ly.