

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.

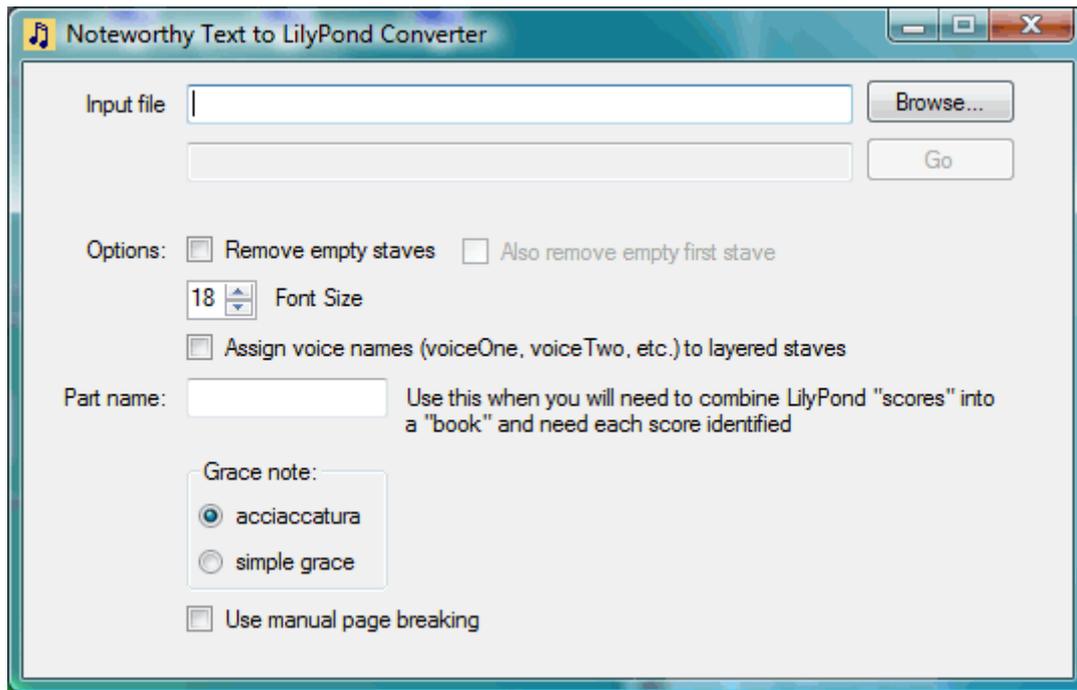
Simple Use

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.

The image displays a musical score for a vocal duet and piano accompaniment. The top section features two vocal staves: 'Vocal' and 'Bass'. The 'Vocal' staff has five numbered measures (1-5) with lyrics: 'me! Thou the tree, and I the flow - er, Thou the i - dol, I the throng;'. The 'Bass' staff has corresponding lyrics: 'I the tree, thou the flow - er, I the i - dol, thou the throng;'. Below the vocal staves is a piano accompaniment section labeled 'Piano', consisting of a grand staff with treble and bass clefs. The tempo marking 'Andante, non troppo lento (♩ = 76)' is visible above the piano staff. The score is rendered in blue ink on a white background.

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 .nwextract extension. The .ly files are LilyPond files and are used to create the score. The .nwextract 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:

Simple Song

Lyrics: Phil Holmes

Music: Phil Holmes

Andante, non troppo lento (♩ = 76)

me! Thou the tree, and I the flow-er;
I the tree, thou the flow-er;
Thou the i-dol, I the thron;
I the i-dol, thou the thron;

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 staff 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:

A musical score for a Christmas carol extract, showing two staves. The top staff is in treble clef, 4/4 time, with a key signature of one sharp (F#). The bottom staff is in bass clef, 4/4 time, with a key signature of one sharp (F#). The lyrics are: "An - gels from the realms of glo - ry,". The first staff has two measures, with a first ending bracket over the first measure and a second ending bracket over the second measure. The notes are: G4 (quarter), A4 (quarter), B4 (quarter), C5 (quarter), B4 (quarter), A4 (quarter), G4 (quarter), F4 (quarter). The first voice (upper) has up-stems, and the second voice (lower) has down-stems.

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:

Angels, from the realms of glory

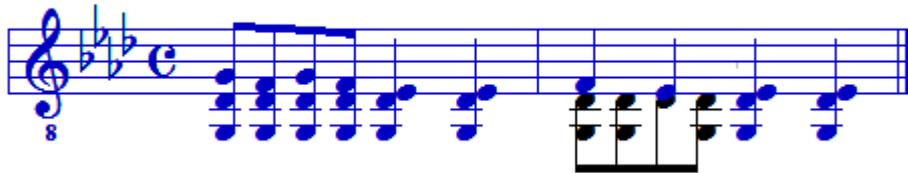
Lyrics: Montgomery

Music: French carol



The image shows a musical score for the hymn "Angels from the realms of glory". It consists of two staves, a treble clef on top and a bass clef on the bottom, both in a key signature of one sharp (F#) and a common time signature (C). The lyrics "Angels from the realms of glory," are written between the two staves. The top staff contains a melody with upward stems, and the bottom staff contains a bass line with downward stems.

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:



The image shows a musical snippet on a single treble clef staff. The key signature is three flats (B-flat, E-flat, A-flat) and the time signature is common time (C). The first bar contains a single note with an upward stem. The second bar contains a chord with four notes, all with upward stems. The third bar contains a chord with four notes, all with upward stems. The fourth bar contains a chord with four notes, all with upward stems. The fifth bar contains a chord with four notes, all with upward stems. The sixth bar contains a chord with four notes, all with upward stems. The seventh bar contains a chord with four notes, all with upward stems. The eighth bar contains a chord with four notes, all with upward stems. The ninth bar contains a chord with four notes, all with upward stems. The tenth bar contains a chord with four notes, all with upward stems. The eleventh bar contains a chord with four notes, all with upward stems. The twelfth bar contains a chord with four notes, all with upward stems. The thirteenth bar contains a chord with four notes, all with upward stems. The fourteenth bar contains a chord with four notes, all with upward stems. The fifteenth bar contains a chord with four notes, all with upward stems. The sixteenth bar contains a chord with four notes, all with upward stems. The seventeenth bar contains a chord with four notes, all with upward stems. The eighteenth bar contains a chord with four notes, all with upward stems. The nineteenth bar contains a chord with four notes, all with upward stems. The twentieth bar contains a chord with four notes, all with upward stems. The twenty-first bar contains a chord with four notes, all with upward stems. The twenty-second bar contains a chord with four notes, all with upward stems. The twenty-third bar contains a chord with four notes, all with upward stems. The twenty-fourth bar contains a chord with four notes, all with upward stems. The twenty-fifth bar contains a chord with four notes, all with upward stems. The twenty-sixth bar contains a chord with four notes, all with upward stems. The twenty-seventh bar contains a chord with four notes, all with upward stems. The twenty-eighth bar contains a chord with four notes, all with upward stems. The twenty-ninth bar contains a chord with four notes, all with upward stems. The thirtieth bar contains a chord with four notes, all with upward stems. The thirty-first bar contains a chord with four notes, all with upward stems. The thirty-second bar contains a chord with four notes, all with upward stems. The thirty-third bar contains a chord with four notes, all with upward stems. The thirty-fourth bar contains a chord with four notes, all with upward stems. The thirty-fifth bar contains a chord with four notes, all with upward stems. The thirty-sixth bar contains a chord with four notes, all with upward stems. The thirty-seventh bar contains a chord with four notes, all with upward stems. The thirty-eighth bar contains a chord with four notes, all with upward stems. The thirty-ninth bar contains a chord with four notes, all with upward stems. The fortieth bar contains a chord with four notes, all with upward stems. The forty-first bar contains a chord with four notes, all with upward stems. The forty-second bar contains a chord with four notes, all with upward stems. The forty-third bar contains a chord with four notes, all with upward stems. The forty-fourth bar contains a chord with four notes, all with upward stems. The forty-fifth bar contains a chord with four notes, all with upward stems. The forty-sixth bar contains a chord with four notes, all with upward stems. The forty-seventh bar contains a chord with four notes, all with upward stems. The forty-eighth bar contains a chord with four notes, all with upward stems. The forty-ninth bar contains a chord with four notes, all with upward stems. The fiftieth bar contains a chord with four notes, all with upward stems. The fifty-first bar contains a chord with four notes, all with upward stems. The fifty-second bar contains a chord with four notes, all with upward stems. The fifty-third bar contains a chord with four notes, all with upward stems. The fifty-fourth bar contains a chord with four notes, all with upward stems. The fifty-fifth bar contains a chord with four notes, all with upward stems. The fifty-sixth bar contains a chord with four notes, all with upward stems. The fifty-seventh bar contains a chord with four notes, all with upward stems. The fifty-eighth bar contains a chord with four notes, all with upward stems. The fifty-ninth bar contains a chord with four notes, all with upward stems. The sixtieth bar contains a chord with four notes, all with upward stems. The sixty-first bar contains a chord with four notes, all with upward stems. The sixty-second bar contains a chord with four notes, all with upward stems. The sixty-third bar contains a chord with four notes, all with upward stems. The sixty-fourth bar contains a chord with four notes, all with upward stems. The sixty-fifth bar contains a chord with four notes, all with upward stems. The sixty-sixth bar contains a chord with four notes, all with upward stems. The sixty-seventh bar contains a chord with four notes, all with upward stems. The sixty-eighth bar contains a chord with four notes, all with upward stems. The sixty-ninth bar contains a chord with four notes, all with upward stems. The seventieth bar contains a chord with four notes, all with upward stems. The seventy-first bar contains a chord with four notes, all with upward stems. The seventy-second bar contains a chord with four notes, all with upward stems. The seventy-third bar contains a chord with four notes, all with upward stems. The seventy-fourth bar contains a chord with four notes, all with upward stems. The seventy-fifth bar contains a chord with four notes, all with upward stems. The seventy-sixth bar contains a chord with four notes, all with upward stems. The seventy-seventh bar contains a chord with four notes, all with upward stems. The seventy-eighth bar contains a chord with four notes, all with upward stems. The seventy-ninth bar contains a chord with four notes, all with upward stems. The eightieth bar contains a chord with four notes, all with upward stems. The eighty-first bar contains a chord with four notes, all with upward stems. The eighty-second bar contains a chord with four notes, all with upward stems. The eighty-third bar contains a chord with four notes, all with upward stems. The eighty-fourth bar contains a chord with four notes, all with upward stems. The eighty-fifth bar contains a chord with four notes, all with upward stems. The eighty-sixth bar contains a chord with four notes, all with upward stems. The eighty-seventh bar contains a chord with four notes, all with upward stems. The eighty-eighth bar contains a chord with four notes, all with upward stems. The eighty-ninth bar contains a chord with four notes, all with upward stems. The ninetieth bar contains a chord with four notes, all with upward stems. The ninety-first bar contains a chord with four notes, all with upward stems. The ninety-second bar contains a chord with four notes, all with upward stems. The ninety-third bar contains a chord with four notes, all with upward stems. The ninety-fourth bar contains a chord with four notes, all with upward stems. The ninety-fifth bar contains a chord with four notes, all with upward stems. The ninety-sixth bar contains a chord with four notes, all with upward stems. The ninety-seventh bar contains a chord with four notes, all with upward stems. The ninety-eighth bar contains a chord with four notes, all with upward stems. The ninety-ninth bar contains a chord with four notes, all with upward stems. The hundredth bar contains a chord with four notes, all with upward stems.

Here we have most of the notes of voice one stem up (owing to their position on the staff) 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 staff). 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.

The image displays a musical score for a piece in 4/4 time, featuring eight staves. The score is written in a key signature of one flat (B-flat) and common time (C). The staves are labeled as follows:

- Solo1**: The first staff, marked with a blue anchor icon.
- Solo2**: The second staff, marked with a grey anchor icon.
- Choir1**: The third staff, marked with a grey anchor icon.
- Choir2**: The fourth staff, marked with a grey anchor icon.
- Orchestra1**: The fifth staff, marked with a grey anchor icon.
- Orchestra2**: The sixth staff, marked with a grey anchor icon.
- Piano Right Hand**: The seventh staff, marked with a grey anchor icon.
- Piano Left Hand**: The eighth staff, marked with a grey anchor icon.

The music consists of a simple melodic line in each part, primarily using quarter and half notes. The piano part is written in grand staff notation, with the right hand in the treble clef and the left hand in the bass clef. A blue horizontal line is drawn across the top of the score, and a vertical blue line is drawn at the beginning of the first staff.

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

The image displays a musical score with eight staves. The top two staves are labeled 'Solo1' and 'Solo2'. The next two staves are labeled 'Choir1' and 'Choir2'. The following two staves are labeled 'Orchestra1' and 'Orchestra2'. The bottom two staves are labeled 'Piano Right Hand' and 'Piano Left Hand'. The score is in common time (C) and has a key signature of one flat (Bb). The notation shows a simple melody in the upper parts and a piano accompaniment in the lower parts.

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

The image shows a musical score for a piece titled 'No 3 - Song (Pish Tush) and Chorus'. The score is arranged for voice and piano. It begins with a piano introduction marked 'Allegro ma brio' with a tempo of quarter note = 126. The introduction features a rhythmic piano accompaniment. At measure 8, the vocal line begins with the lyrics 'Our great Mi-ka-da, vic-tor-ious man, When he to rule our land be-gan, Re-solv'd to try A'. The piano accompaniment continues with a steady rhythm. At measure 16, the vocal line continues with the lyrics 'plan where - by Young men might best be stud-ied. So he de-creed, in words suc-cinct, That all who sin-ed, leas'd, or wink'd, (Un-'. The piano accompaniment provides harmonic support throughout the vocal lines.

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:

No 3 - Song (Pish Tush) and Chorus

Lyrics: W S Gilbert

Music: A S Sullivan

Allegro con brio (♩ = 126)

8 *Pish Tush*
Our great Mikado, virtuous man, When he to rule our land began, Re-

15
solv'd to try A plan whereby Young men might best be steadied. So he decreed, in words succinct, That

Arranged by Phil Holmes. Copyright 2010

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 staff" then all the empty ones are removed, as below. I always run with these options set.

No 3 - Song (Pish Tush) and Chorus

Lyrics: W S Gilbert

Music: A S Sullivan

Allegro con brio (♩ = 126)

Pish Tush

Our great Mikado, virtuous man, When he to rule our land began, Re-

dim *p*

15 solv'd to try A plan whereby Young men might best be steadi'd. So he decreed, in words succinct, That

21 all who flirted, leer'd, or wink'd, Un-less connubi - al-ly link'd,) Should forthwith be be-headed, be-

Arranged by Phil Holmes. Copyright 2010

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 containing only "spacer rests" (in LilyPond notation). Spacer rests are translated from notes or rests with the display property 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 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.

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.