

# LilyPond

ver2.22.1-1

# On Windows10

Japanese-version

## Preface(Lilypond)

It must have been during a rehearsal of the EJE (Eindhoven Youth Orchestra), somewhere in 1995 that Jan, one of the cranked violists told Han-Wen, one of the distorted French horn players, about the grand new project he was working on. It was an automated system for printing music (to be precise, it was  $M^P$ , a preprocessor for MusiX $T_E$ X. As it happened, Han-Wen accidentally wanted to print out some parts from a score, so he started looking at the software, and he quickly got hooked. It was decided that  $M^P$  was a dead end. After lots of philosophizing and heated email exchanges, Han-Wen started LilyPond in 1996. This time, Jan got sucked into Han-Wen's new project.

In some ways, developing a computer program is like learning to play an instrument. In the beginning, discovering how it works is fun, and the things you cannot do are challenging. After the initial excitement, you have to practice and practice. Scales and studies can be dull, and if you are not motivated by others, teachers, conductors or audience. It is very tempting to give up. You continue, and gradually playing becomes a part of your life. Some days it comes naturally, and it is wonderful, and on some days it just does not work, but you keep playing, day after day.

Like making music, working on LilyPond can be dull work, and on some days it feels like plodding through a morass of bugs. Nevertheless, it has become a part of our life, and we keep doing it. Probably the most important motivation is that our program actually does something useful for people. When we browse around the net we find many people who use LilyPond, and produce impressive pieces of sheet music. Seeing that feels unreal, but in a very pleasant way.

Our users not only give us good vibes by using our program, many of them also help us by giving suggestions and sending bug reports, so we would like to thank all users that sent us bug reports, gave suggestions or contributed in any other way to LilyPond.

Playing and printing music is more than a nice analogy. Programming together is a lot of fun, and helping people is deeply satisfying, but ultimately, working on LilyPond is a way to express our deep love for music. **May it help you create lots of beautiful music!**

*Han-Wen and Jan.*

*Utrecht/Eindhoven, The Netherlands, July 2002.*

—Excerpt from *LilyPond[ver1.6-lilypond.pdf]*—

## LilyPond プログラム設計の基幹設計理念は

*Han-Wen Nienhuys*・*Jan Nieuwenhuizen* のお二人が

“Proceedings of the XIV Colloquium on Musical Informatics(XIV CIM 2003)” で行った講演で明らかにされています。  
(Firenze, Italy, May 8 10. 2003)

### **LILYPOND, A SYSTEM FOR AUTOMATED MUSIC ENGRAVING**

See “ <http://lilypond.org/publications.html> ” ( download!! pdf)  
[But; Commands in this are old's version(1.6), and Now days is version(2.20)]

*In the latest W32TeX and TeX Live 2021, makeinfo.exe(32bit) etc... are excluded  
(because it is not used except me in windows OS). Absolutely here Required.*

---

LilyPond ver2.22.1 + W32TeX[TeX Live 2017/W32TeX] + GhostScript(ver 9.53.3)

*This book was produced with the pTeX format, version 3.141592653 p1.0b.*

**W32TeX**;format

[texfmt.cmf(2017/10/22)+fmtutil.cmf(2017/01/22)+updump-dist.cfg(2017/12/25)...]

2021 Yukio Yoshida

The command-line utility "LilyPond" is a tool for Music-sheet users.

### LilyPond language 譜面原稿記述において

LilyPond のデフォルトは `nederlands.ly`(In Dutch=オランダ語表記) を読み込みで、与えられたその原稿処理を行います。音楽における **Note Names** , **sharp** , **flat** 等々の strings には国(文化)間で差異がありますので的便に置き換えて作業です。Lilypond が提供しているそれらは、

`english.ly` , `deutsch.ly` , `norsk.ly` , `svenska.ly` , `italiano.ly` , `catalan.ly` ,  
`espanol.ly` , `portugues.ly` , `suomi.ly` , `vlaams.ly`

これらは、譜面原稿ファイル開始時の宣言項目(下記の順番記述)です。例えば、

```
\version "2.22.1"  
\include "italiano.ly"  
と記述(宣言)します。
```

譜面原稿 coding では `lilypond's-running` は OS を選ばないです、`ly` 原稿ファイルは (LilyPond 環境構築) OS なら `running` します。

LilyPond の解説は(そのバージョンによる)。例えば、

<http://lilypond.org/manuals.html> を参照ですが、  
[Other material] 項目の **All** (すべて) を選択し “**Doc tarball for 2.22.1** (`lilypond-2.22.1-1.documentation.tar.bz2`)” これを download し展開するとご自分のパソコンのなかで LilyPond のすべての解説が閲覧できます(インターネットに無接続)。 `notation.pdf` , `uasge.pdf` , 等々も含んでいます(和語訳は皆無)。

LilyPond のプログラムの興味をの追及は、例えば、

`C:\...\lilypond\...\ly\engraver-init.ly` 等々の内容記述に目を通すことを進めます。

” \ (エスケープ)”は日本語キーボードではキーボードの“¥”あるいは“\”で打ち込むことを意味しています。

LilyPond の `version` は “2.22.”これが **major** 番号で、それ以後の “1-1” 等々の 数値 は **miner** 番号で、miner 番号が変わっても major 番号が同じなら `running` します。

原稿をコーディングしている時は、**日本語特有の全角空白、全角文字は使用禁止 (lyrics-歌詞-を除く) です。** LilyPond はその時点で処理を中止します。 `lyricText` は `utf-8 code` で記述(組版には各国のフォントが必要)です。原稿記述には **shift-jis** (code) は **絶対に使わない**

デフォルト組版は 四分音符・4/4 拍子・ト音記号・ハ長調で組版出力されます。`midi` は指定楽器がなければ、`"acoustic grand"` (ピアノ) で作成されます。

**Windows 版 LilyPond は、歌詞(和語=`msmincho,msgothic`)にそのまま対応しています。**

## LilyPond(楽譜組版言語)の誕生についてです.

彼らはLilyPond(楽譜組版言語)の開発でGNUの仲間の参加をも有り、cpu用プログラム言語であるc++を約4年の歳月をかけて拡張(+約4,000行?)を成し遂げて、**LilyPond ver1.6**としてリリース開始(2003年)になっています.

### ¶ LilyPond 構成要因は

A) 全楽譜部分 [音符・記号等々](and \markup{ 譜面楽譜 }) . . . . . ⇒ schem 処理  
 B) 楽譜; テキスト (and \markup{ テキスト内容 }) . . . . . ⇒ pango(1.6) 処理  
 C) 楽譜化の obje . . . . . ⇒ Ghostscript(9.21) 処理  
 全体が python(2.4) でドライヴ, Windows OS メモリ上に unix 系領域確保で running します, 出来上がった楽譜 pdf,png,svg,midi を Windows10 の APP へと引き渡しています.

LilyPond 環境内の schem,pango,GhostScript,python これらは unix 系のプログラムです. そして unix ですとそこに同列位置で T<sub>E</sub>X 組版プログラムがあり, 初期の LilyPond は T<sub>E</sub>X 仕様の dvi ファイルで楽譜組版情報を作成(吐き出し)していましたが, 今日の LilyPond は直に楽譜組版情報を ps ファイルとして作成しています.

Windows10 にも T<sub>E</sub>X,GhostScript,(schem,python) は存在し, Lilypond の ps ファイルを Windows10 側で楽譜化等々の処理はバッチファイルで可能です.

一例をあげれば(楽譜原稿ファイル sheet-1.ly),

```
lilypond --ps sheet-1.ly
lilypond -debackend=svg sheet-1.ly
gswin64c -sDEVICE=eps2write -sOutputFile=sheet-1.eps -dNOPAUSE sheet-1.ps -c quit
gswin64c -sDEVICE=png16m -r110 -g940x200 -sOutputFile=sheet-1.png -dNOPAUSE sheet-1.ps -c quit
```

上記の様なバッチファイルです (cmd.exe or Powershell.exe で running).[一例の参照です] `-debackend=svg` 作成の sheet-1.svg(楽譜ファイル)の Web 表示はサーバー側に対応がなければ, 自己サーバーの .htaccess ファイルへ下記の追加記述をすれば表示可能になります.

```
# Use PHP5 Single php.ini as default
AddHandler application/x-httpd-php5s .php
```

### ¶ A) 全楽譜部分 [音符・記号等々](and \markup{ 譜面楽譜 }) . . . . . ⇒ schem 処理 ですが,

譜面・音符記号等々の根底処理は \context Voice 種属が行い, これが集合が \context Staff へと生まれ, その Staff 種属が \score へと纏められて楽譜データ code に成っている仕組みです. ですので, \score 種族の中で数個の音符に細工を行ってもエラーになります, 音符等々は根底の \Voice 種族原稿記述中で編集しなければなりません.

詳細は notation.pdf(ver 2.22.1-1) の “3 General input and output”(488 頁)以下の参照を試みてください.

\score の上層に \book{ という幾つもの score を束ねた階層がありますが, 通常は常にバックグラウンドで running しています. これは score と score の間にテキストを組みたいときに明示的に原稿ファイルに宣言記述です. いわゆる, 音楽論文作成等々に利用されます. 通常は一つの score でどんなに厚くても一冊の本の様に楽譜を組みます. マタイ受難曲楽譜などは明示的に \book{ で組むようになります.

(Index is nonexistent)

この編集ファイルの場所は

<http://www.sheet-poem.com/lilypond/reading/sheet-2021-texi.tar.gz>

ファイルを展開すると, 2021\\*.\*, 2021\svg-html\\*.\*, 2021\html\ 2021\refer が作成されています. 原稿ファイルは, test-1.texi running バッチファイルは, run-texi-all.cmd です.

実行は>`run--texi-all test-1`[Enter]

running 終了すると, sheet.html(png 画像) と test-1.pdf が表示されます.

同時に 2021\html\ , 2021\svg-html\ の中にも copy は行われます.

## LilyPond ver2.22.1 on windows10

## context Voice

**音符** 音階 (音の高さ) 基本コード; `c d e f g a b` 音階 (音の高さ明示記号)  
 1 オクターブ下は `c`, 2 オクターブ下は `c,,` .....  
 1 オクターブ上は `c'` 2 オクターブ上は `c''` .....



``` c d e f g a b ''` is default position of LilyPond

Units value(default in World); (a,=110Hz,) a=220Hz, a'=440Hz, a''=880Hz

`\relative c' {c d e f g a b}`等と括れば `c' d' e' f' g' a' b'`と同じ値.

`\relative c, {c d e f g a b}`等と括れば `c, d, e, f, g, a, b,`と同じ値.

`\relative c'' {c d e \relative c' {c d e}}` nesting OK.

**But!** `\relativeve`での記述は, 原稿編集集中に自分が今どの階域にいるか不明になります.

NHK(ラジオ)の時報(告知) 音=a'(440hz) a' a''(880HZ)= ピッ・ピッ・ポンー (長さは四分音・四分音符・二分音符?)

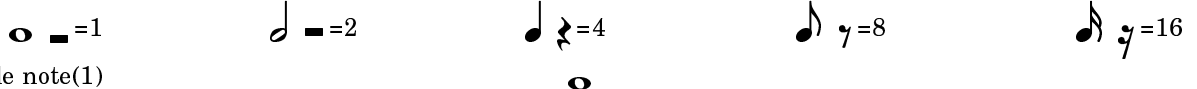
**音符** 音価 (音の長さ) 基本コード;

音階明記の `c d e f g a b`へ数値 1 2 4 8 16 32 64 128 ... の添え字で明記する.

音符記号として `c4`, `a'1`, `g,2` 等々の記述.

1 (Whole)⇒ 2 (Half)⇒ 4 (Quarter)⇒ 8 (Eighth)⇒ 16 (Sixteenth)

1=全音で, 全音を均等に割った数値, 以下図式 ( **休符も音符です** ).



Whole note(1)

Half note(2)

Quarter note(4)

Eighth note(8)

Sixteenth note(16)

これは, 演奏の楽譜化表現の初期時代のころ全音を基幹として成立が確立しそれに基づくものです. 今日ではその意味を逆算され, 楽譜の基幹意味は楽譜の音長さの一番短音から上記図階層上へと成立が一般的傾向です.

Lilypond では音価の付いて無い音符は, その前その前のへと音符音価を検索しその音価を評価します. 例えば, `a4 c8 d e f g c4 s1` = `a4 c8 d8 e8 f8 g8 c4 s1` と解釈処理します.

Lilypond は音符処理として独自の “s” という音価を持った音符を持っています。例えると、幽霊音符で組版 (楽譜) には現れることのない処理音符です。音価は有りますので区切りの間の規則を無視して s 音符 を使うことは出来ません。使用例として、

*play;free-notes*

記述ソースコードは (\Staff setting... 等々は省略).





```
s4^\mf^\< _\markup {\italic play;free-notes} s2 s2. | s1. | s1. |
g2^\ff\! c'1 * 2/3^\> s1 * 1/3\!\^{\markup {\italic\bold "f.o." }} |
s 音符は音価を持っています. 上記の例えでは \time 3/2 を超えて記述は出来ません,
s 音符は上記例えの如く主たる利用は組版 (楽譜) 専用パートにて幽霊音符の役目です.
s 音符綴りにて幽霊 staff も可能である.
c'1 * 2/3^\> s1 * 1/3\!\^{\markup {\italic\bold "f.o." }} の部分も歴然と組版 (楽譜) 専用記述です.
midi-part なら c1 = {c8 c4\p c4\pp c4.\ppp c2\pppp } の感じでしょう.
```

この意味するところは、同一譜面コードを “組版楽譜用出力・midi用” と LilyPond が備えている `\layout{.....}` と `\midi{.....}` を利用する、組版用をその中で先鋭化させれば譜面コード記述作業は楽になります。その midi は楽譜先鋭化の参照聴感と思った方が good! です。

(曲や味覚等々は時代とともに流動しています、曲に至っては 18 世紀に奏でていた音楽は聴けません。江戸時代からの醤油の味覚は暖簾は今日へと受け継がれていても不可能です。)

(—LilyPond は他に、`\skip <引数>(=do nothing! // unlike “s”)` もあります。—)

音価についてですが、更に全音の上に Lilypond には longa と breve という音符が有ります。

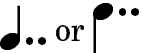

Modern expression;  (longa) and  (breve) Middle Ages expression;  (longa) and  (breve)

この今日的での意味は longa[全音 (1) × 4], breve[全音 (1) × 2] なのですが (=brevis), ヨーロッパ中世期では終止音を書くときに使われ (当時) 通常演奏では決まった長さを持つことはない音価の研究報告があります。

—参照; 日本語訳本 [記譜法の歴史 (モンテヴェルディからベートーヴェンへ)] 春秋社 2015/07/25—

また音価には従来の伝統的; 全音均等分割から、今日的な思考をもたらした “付点音符” という近世期が編み出した音符が有ります。この指向 (思考) が短音基準から全音へという楽譜としての原稿記述や既存楽譜の読みの変化を演奏家・作曲家に浸透始めています。

As other note(duration) are ‘Augmentation dots’ that make augmentation the note.

a4.. ; is a4 note that have duration a4+eight+sixteenth . like this  or 

Lilypond でも a4. (付点四分音符), a4.. (複付点四分音符) 等々とペン書きと同じです。

音の長さ (duration) 表現として音価には、音楽界での“タイ (tie)”記号が有ります。



g4 + g4でg2の音価(響き)を持つのですが、注意が必要になります。

(LilyPondでは g4 ~ | g4 の記述になります。)

通常; 曲は始めの一音に強のアクセントが置かれます,

タイは上図のような感覚(区切りと区切り間で)で使われますのでアクセントの変化が生じます。

これは、“**Syncopation**”(on rhythm)の発生を促します。

音の duration 表現として triplets が有ります, [— 3 —] の部分です数学的にはあり得ない!!

Tuplet(s) とは

音のリズミカルなグループ分け (Rhythmic groupings of notes) で

音の韻律分け (Metric groupings of notes) では有りません。

決められた一区切りの内でどうしても追加音が欲しいが、もっと語るなら二音も三音もその区切りの中で使いたい(奏でたい)。

多くの作曲家が Rhythmic groupings of notes として triplets 明記で実現しています(数学的には割り切れません)。

**LilyPond** で記述は、`\tuplet <引数> <基音(省略OK)> {codeing.....}` で実現できます。

**LilyPond** は全てが自動計算されていることを忘れずに!!

例えば上記の譜面ですが

(四分音符を三個の八分音符, 二分音符を三個の四分音符に分割=音価は四分音符, 二分音符),

```
tenor = {
  \autoBeamOff
  r8 \tuplet 3/2 4 {fis'8 f' fis' f' fis' f' } fis2
  e8[ d c ] |
  c4 e8 g a \tuplet 3/2 4 {b c' d' e' fis' g' } b'4. |
  g2. \tuplet 3/2 4 {<g b>8 <a c'> <b d'> }
  \tuplet 3/2 2 {fis4 f fis ~ } | \noBreak
  fis4 <d a>2 <e g>4 ~ <b e>2\(| \time 4/1 e\longa\)| \break
}
\score {\context Voice = "tenor" {\clef tenor \key g\major \time 3/2
  \set Staff.instrumentName = "tenor " \tenor }
\layout {}
}
```



### Grace notes(装飾音符)

これは、楽譜表現において音価を持たない音符群です。

The image displays a musical score for four voices (soprano, alto, tenor, bass) and piano accompaniment. The score is in 4/4 time and B-flat major. The vocal lines are written in soprano, alto, tenor, and bass clefs. The piano accompaniment is written in a grand staff (treble and bass clefs). The score is divided into three systems. The first system shows the vocal lines and the piano accompaniment. The second and third systems show the piano accompaniment with repeat signs (slashes) indicating a repeating rhythmic pattern. The piano accompaniment features a repeating rhythmic pattern of eighth notes and sixteenth notes, with grace notes (ornaments) attached to some of the notes.

Music engraving by LilyPond 2.22.1—[www.lilypond.org](http://www.lilypond.org)

始めの一小節目 (longa) のなかの音符は全て音価を持っているが、  
三小節目からの repeat3 の繰り返しの部分の 32・16 分音符が装飾音符で  
音価を持っていません。

🔍 ソースコード、および試聴は [commentary-html\(sheet.html\)](#) の方をお願いします。

後続く、おいおいの編集作業です ( \_ \_ ) .

## LilyPond ver2.22.1 on windows10

# Formatting text

LilyPond 原稿記述でのテキスト (組版) 出力は\markupが担っていますが、

これは、絶えずバックグラウンドでrunningしている pango(1.6) が即座に反応して処理している構図です。

\markup や \Lyrics( 歌詞部分 ) の内では各国の活字 (電子) フォントが使えます, 和語活字も OTF 仕様で組版出力出来ます。

特に lyrics-part は完全に “ UTF-8 encoding ” なので、

LilyPond 原稿記述は utf-8 コード記述で作業が最良になります。

また、組版に使う活字体仕様は OTF(OpenType//TTF-TrueType を含む) に限ります。

LilyPond は ps ファイル仕様出力を念頭に置くと、初心者でも簡単な ps 記述で以下のようなことが可能です。

5 小節目の—lilyPond—がそうなのですが、この仕組みを明らかにすると 音楽標語・記号 へ 個人独自の obje 合戦が始まり、楽譜としての統一共有感覚はカオスになるかも？

The image shows two musical staves. The first staff contains four measures with different fermata symbols: a simple fermata (\segno), a fermata with a diamond (\coda), a fermata with a diamond and a vertical line (\varcoda), and a fermata with a diamond and a box (\eps; my-box). The second staff contains four measures with different fermata symbols: a short fermata (\shortfermata), a fermata with a diamond (\fermata), a fermata with a diamond and a vertical line (\longfermata), and a fermata with a diamond and a box (\verylongfermata). The LilyPond logo is visible in the top right corner of the first staff.

Music engraving by LilyPond 2.22.1—[www.lilypond.org](http://www.lilypond.org)

問題の 5 小節目の組版記述ですが、  
 此处では my-epsbox.ps を作成記述⇒eps ファイルへ変換⇒sheet-my-epsbox.ly へ貼り付けの明示です。

---

— ¶ my-epsbox.ps から sheet-my-epsbox.eps 作成バッチファイル。

```
gswin64c -sDEVICE=eps2write -sOutputFile=sheet-my-epsbox.eps -dNOPAUSE my-epsbox.ps -c quit
```

これは、(64bit)windows10+GhostScript での記述です。

(32bit) なら gswin32c の文言に書き換えると、

-sDEVICE=eps2write は -sDEVICE=epswrite に GhostScript-version によっては書き換えが必要です。いずれも、これらは PC 環境 path 値に書き込みがある場合です。

(LilyPond 環境内 gs(関連.exe) は Windows OS は無関心です。)

— ¶ `my-epsbox.ps` の内容です.

```

%!PS-Adobe-3.0 EPSF-3.0
/BHelvetica-Bold findfont 48 scalefont setfont
20 40 moveto
1 0 0 setrgbcolor
(LilyPond) false charpath
2 setlinewidth stroke
showpage
%Trailer

```

これを, 前項のバッチファイルで running すると `sheet-my-epsbox.eps` が作成されます.

— ¶ `sheet-my-epsbox.ly` の内容です.

```

%% Yukio Yoshida; 8/7/2017 13:47
%% ly snippet: Coding: utf-8
\version "2.20.0"
\paper{
  paper-height = 40.00\mm
  paper-width = 160.00\mm
  page-top-space = 0.00\mm
  top-margin = 0.00\mm
  bottom-margin = 0.00\mm
}
%% \header { tagline = \markup { \null } }
#(set-global-staff-size 16)
\score {
  \context Staff = "strings" { \clef treble \time 4/4
    \override Score.BarNumber.break-visibility = ##(#f #f #f)
    c''1\segno_\markup{"\segno"} | c''1\coda_\markup{"\coda"}
    c''1\varcoda_\markup{"\varcoda"} | s1 |
    c''1^\markup{\epsfile #X #10 #"sheet-my-epsbox.eps" }
      _\markup{eps; my-box} | \break
    c''1\shortfermata_\markup{"\shortfermata"} |
    c''1\fermata_\markup{"\fermata"} |
    c''1\longfermata_\markup{"\longfermata"} |
    c''1\verylongfermata_\markup{"\verylongfermata"} | s1 | }
    \layout { indent = 0.00\mm }
  }
}
% end ly snippet:

```

---

以上の如くなのですが, ps ファイルの作成は ‘文字’ 程度なら上記のような感じでどなたでも記述できます.

PostScript マニュアル (和語) 参照

[http://www.umekki.jp/college/syllabus/06\\_report/citation/postscript.pdf](http://www.umekki.jp/college/syllabus/06_report/citation/postscript.pdf)

小節番号は LilyPond default で staff 2 行目から行頭に組版ですが `#t` を `#f` にして全て無視設定です.

LilyPond は `\markup` の中でも `music` 記述は可能になっています。  
 例えば、47 小節目が `\markup` の中で記述です。

The image shows a musical score snippet. The main staff is a bass line in 3/2 time, starting at measure 44 and ending at measure 47. Measure 47 is highlighted with a red staff. Above this red staff, the text 'treble; G clef on 2nd line' is written. Below the red staff, the text '( Tutti S A T mp )' is written. The measure number '47' is centered below the red staff.

このソース記述です。

---

```

%% Yukio Yoshida: 8/10/2017 03:42
%% ly snippet: Coding: utf-8
\version "2.20.0"
\paper{ paper-height = 30.00\mm
        paper-width = 160.00\mm
        page-top-space = 0.00\mm
        top-margin = 0.00\mm
        bottom-margin = 0.00\mm }
\header { tagline = \markup { \null } }
#(set-global-staff-size 16)
bass = { \autoBeamOff \bar ""
        c16 d c8 b1 r4 | d16 c b8 c1 r4 | b16 c d1 r4. |
        c1.^{\markup{ \hspace #-10 \score { {
            \set Staff.fontSize = #-4
            \override Staff.StaffSymbol #'staff-space = #(magstep -4)
            \override Staff.StaffSymbol #'thickness = #(magstep -4)
            \override NoteHead #'color = #red
            \override Stem #'color = #red
            \clef treble \key c \major \time 3/2
            r2^{\markup{ char92\hspace #-4 "treble; G clef on 2nd line"}
            < g'' c'' g' >1.\markup{ \hspace #-8 ( \italic "Tutti" S A T
            \hspace #1 \bold \italic "mp" ) } }
            \layout { indent = 0.00\mm line-width = 23.00\mm } } } | \break
        }
\score {
  \new Voice = "basso" { \clef bass \key c \major \time 3/2 \bass }
  \layout { \context { \Score {
    \override Score.BarNumber.break-visibility = ##(#f #t #t)
    \set Score.currentBarNumber=44 } } |
  }
%% end ly snippet:

```

---

この譜面の `midi-out` の方は、同一譜面で各パート`{r1. | r1. | r1. | r2 (各音符)1 |}`と`\midi{`に記述すれば、譜面通りの曲になります。同じ譜面形態でも LilyPond には、様々な記述方法があります、この例はその一つで著者の方法です、`\bass` の意味は `bass=`で記述したコードを此処へ呼び込め...の記述行動です。またこの記述方法はバックグラウンド `running` している `\book`を明示化して、`\book{ \bookpart{... } \bookpart{... }..... }` 記述は優れて利便性が発揮してきます。

楽譜記述のサンプルです.(原稿は\book{.....}を明示化して記述しました)

*Rainy in  
Midnight*

### *Rainy in Midnight*

#### **Summary;**

On a quiet midnight has being struck by rain.  
 Be hatred, Be beloved, Be relieved, Be hardship,  
 Be sadness, Be worries. All of them,  
 Breath of life have being struck rain in a quiet midnight

夜の静かさが雨に打たれている。  
 憎しみと愛しさと安らぎと苦しみと  
 悲しみと悩みと, それら生命の息吹が  
 夜の静かさの中で雨に打たれている。

Childhood days has being struck by rain.  
 Bygone memories of in my heart,  
 Certainly, there were the footprints of life,  
 In a quiet midnight has being struck by rain.

幼かった日々が雨に打たれている。  
 過ぎ去った思い出の数々が,  
 確かにあった日々が, 生命の足跡が  
 夜の静かさの中で雨に打たれている。

All melted all sink,  
 All darkness,  
 Wet in the rain at night,  
 To erase the living. ( **sing; repeat 2 + 1** )

全て溶かし, 全て流し  
 全て消して  
 雨が夜の中をぬらし  
 生きた事を消し去っている。

*Yukio Yoshida.*

Surge up Somthing  
**Rainy in Midnight**

Is that about Death!

soprano; C clef on 1st line, alto; C clef on 3rd line.  
 tenor; C clef on 4th line, bass; F clef on 4th line.

Yukio Yoshida  
 Largo

yukio yoshida  
 Sep. 05, JST 2008  
 Op. 33, No.1

soprano  
 On a quiet midnight has being struck by rain. Be hatred, Be beloved, Be relieved,

alto  
 On a quiet midnight has being struck by rain. hatred, beloved, relieved,  
*mf*

tenor  
 On midnight has being struck by rain. Be hatred, Be beloved, Be relieved,  
*mf*

bass  
 A quiet midnight has being struck by rain. hatred, beloved, relieved,

4

soprano  
 Be hardship, Be sadness, Be worries. All of them, Breath of life have being struck rain  
*p*

alto  
 hardship, sadness, worries. All of them, Breath of life have being struck rain  
*p*

tenor  
 Be hardship, Be sadness, Be worries. All of them, Breath of life have being struck rain  
*mf*

bass  
 hardship, sadness, worries. All of them, Breath of life have being struck rain  
*mf*

4

8

in a quiet midnight. Child-hood days

in a quiet midnight.

in a quiet midnight.

in a quiet midnight. Child-hood days

12

by rain. Bygone memories of in my heart, Cer-tain-ly,

has being satruck by rain. Bygone memories of in my heart, Cer-tain-ly,

has being satruck by rain. Bygone memories of in my heart, Cer-tain-ly,

by rain. Bygone memories of in my heart, Cer-tain-ly,

16

there were the foot-prints of life, In a quiet mid-night has being struck by rain.

there were the foot-prints of life, In a quiet mid-night has being struck by rain.

there were the foot-prints of life, In a quiet mid-night has being struck by rain.

there were the foot-prints of life, In a quiet mid-night has being struck by rain.



20

All melted all sink, All darkness,

All melted all sink, All darkness,

22

All melted all sink, All darkness, Wet in the ra... in at night,

All melted all sink, All darkness, Wet in the ra... in at night,

24

To erase the living. To erase the li... vi... ng.

To erase the living. To erase the li... vi... ng.

To erase the living. To erase the li... vi... ng.

To erase the living. To erase the li... vi... ng.

ソースコードの参照は [sheet.html](http://www.sheet-poem.com/lilypond/reading/sheet.html) の当確箇所で行います。  
<http://www.sheet-poem.com/lilypond/reading/sheet.html>

**sheet-rainy-midnight.ly** の中で **cover** のタイトル文字 (ps 記述) を処理しています,  
**前もって (編集作業中)**  
(See;[cover.ps] を前もって windows 側で eps 処理しその %%BoundingBox の値,cover.ps 記述を貼り付けて”sheet-my-epstitle.ps”として記述しています。)  
原稿 ly ファイルの処理の全ては…Preloading format された unix 系での running ですので、当然ここも lilypond 内の gs 処理で,—windows OS 側の gswin, 等々は関与していません—。

後続く, おいおいの編集作業です ( \_ \_ ).



## LilyPond ver2.22.1 on windows10

# Important description

LilyPond 原稿記述で最も重要で絶えず念頭に置いておくことは、Call;Stave に於いて、

```
\context Voice = "ID-name" { music notation ....}   という記述と。
単に簡単に (\context Voice = \new Voice 同一です),
\new Voice { music notation ....}   という記述の二通りあります。
これは,(同様に;\context Staff = \new Staff 同一です)
\context Staff = "ID-name" { music notation ....}
\new Staff { music notation ....} にも当てはまります。
```



但し (Call;Stave) の、Staff と Voice は違います、明確に意識しておく事は重要です  
(例表示; 下記譜面).

From Réquiem-ceremony  
**Agnus Déi**

soprano

Voice; sop

A - gnus Dé - i, qui tól - lis pec - cá - ta mún - di:  
tól - lis pec - cá - ta mún - di:

At location( **note-head** ) , ; *The apportionment must be decided by you .*

Music engraving by LilyPond 2.22.1—[www.lilypond.org](http://www.lilypond.org)

上記譜面のソースコード参照は [sheet.html](http://sheet.html) の当確箇所で行願いたします。

<https://www.sheet-poem.com/lilypond/reading/sheet.html>

@menu の Important description をクリック

要点を述べると;

上記譜面の赤色で綴ってある、合唱が二手に分かれて歌う箇所のコード綴りですが、  
call;Stave には \context Voice or \new Voice で Stave を呼び出 (make) さなければならな

い (must be) ということです (歌詞との連携をも考慮.)

(ちなみに、当確場所を `\context Staff` or `\new Staff` へ書き換えて `running` をすれば一目瞭然です.)

この例記述は私 (著者) の例です、`lilypond` は他のアプローチ記述もあります。が、当確場所でも `\context Staff` or `\new Staff` は使えません。

```
sopranoVoice = {
  \autoBeamOff
  \time 3/2 a'8-. \melisma b'-. c''-. \melismaEnd b'8( c''2) d''2 |
  \once \override NoteHead.front-size = #3
  \once \override NoteHead.style = #'petrucci
  \once \override Stem.color = #(rgb-color 1 1 1)
  e''2.
  << \context Voice = "STII" \with \autoBeamOff
    { \colorRed { f''4. f'' |
      f''4. \rest c''8 \ ( d'' e'' \ ) f''2. \ ( | g''2 \ ) } } \
  \context Voice = "STI" \with \autoBeamOff
    { \colorRed { r8 f'4. a8 \rest e'8 ~ |
      e'4 cis'4 cis' \once \override NoteHead.front-size = #3
      \once \override NoteHead.style = #'petrucci
      \once \override Stem.color = #(rgb-color 1 1 1)
      cis'2. \ ( | e'2 \ ) } }
  >> r1 | \break
}
```

明確に、`\context Voice` (or `\new Voice`) でコーテングです。

此処を、`\context Staff` (or `\new Staff`) でコーテングは出来ません

著者の書いている原稿譜面コードを `LilyPond` が `running` 処理が行うとき、一番根底で処理を行うのが “Voice Groups” の音符処理 (`schem=***.scm` , `***.ly` 等々参照) で、その収検が。

“Staff Group” へ集合処理となり、各 `Staffs` の収検が “Score” へ集合と纏められて楽譜へと組版されます。

この処理呼び出しに二つの方式が用意されている点です。

例えば、Voice 処理したコンピューター・メモリのスタックスに “ID-name(任意命名)” を付けて置く `running` 方法と、

コンピューター・メモリのスタックスに名前を付けずに `running` へと継続している方法です。

### コンピューター

処理・メモリのスタックスに “ID-name(任意命名)” を付けて置く `running` 方法は、この ID-name で原稿記述 `\score{...}` の中でいつでも呼び出せる事です

また留意・注意しておくことは  
`\context Staff{ or, \new Staff{` は Voice groups をそのまま引き継ぐとあることですが、これは引き継ぐことであって、著者の Voice groups への変更をも作用する(受け継ぐ)ことではないです。

Voice group への各種 `\override` は Voice 呼び出しで行う。  
 Staff group への各種 `\override` は Staff 呼び出しで行う。  
 Score group への各種 `\override` は Score 呼び出しで行う。

例えば、以下の如くにです。

```
\override Score.BarNumber #'font-size = #2 (小節番号フォントサイズ;2)
\override Staff.StaffSymbol #'color = #blue (五線譜シンボルは色は all-blue)
\override Voice.NoteHead #'color = #blue (音符頭の色は blue)
```

LilyPond の default へ戻すのは、

```
\revert Staff.StaffSymbol #'color
\revert Voice.NoteHead #'color
```

#### 前頁譜面の html での全ソースコード参照で

`\book{ \score{.....} }` と記述ですが、見ての通り譜面組版出力が済んだ後で `\score{.....}` 外で `\markup` を使用するためです。 `\score` 記述全体を `\book` で包んでしまう書式です。 `\book` で囲った中で `\markup` を用いて、文・画像・obje・等々処理可能で(譜面はその一部門です!), lilypond で runnig すればその pdf/svg が出来上がります。

但し eps 画像 (adobe 形式 svg) 等々部分は白紙組版です (pdf/png 出力 running は OK). lilypond も svg(描画) を作成しますが adobe 形式では無く, xml[http://www.w3.org/...(勧告)] 等々に因る書式方式ですので相容れません [svg 出力では eps 画像部分は白紙出力です]. また, `\book{ \bookpart{.....} \bookpart{.....} }` 書式もあります。

更に, scheme が常時 open しているなら Ghostscript に因るグラフィック (描画) も可能で。前項の; Ave maria 楽譜の Sheet-my-ave.ly の記述始め項目は Ghostscript 記述 (楽譜表紙描画) から始めています。

後に続く, おいおいの編集作業です m(\_ \_)m.



## LilyPond ver2.22.1 on windows10

## context Staff

Staff 記述で忘れてならないのは、  
その Staff で発信した情報はその Staff 系でしか受信できない。 定理です。

The image shows a musical staff for soprano. It consists of three measures: measure 45, measure 46, and measure 47. Measure 45 is in 4/1 time and contains a half note G4, a quarter note A4, and a quarter note B4. Measure 46 is in 3/2 time and contains a half note C5, a quarter note D5, and a quarter note E5. Measure 47 is in 3/2 time and contains a half note F5 and a quarter note G5. The notes are connected by a slur across measures 45 and 46, and another slur across measures 46 and 47.

このソース記述ですが、  
‘45-47 小節目’ の記述を読み解くと Staff の情報発信と受信の仕組みが理解できると思います。

s 音符と幽霊 staff でもって、発信と受信を同一 Staff にして実施しています。

```
%% 45-47
\time 4/1
<< { s\breve ees'\breve ~ | ees'1 s1 s\breve } \\  
  { <c' g'\breve ~ c'\breve ~ | c'\longa ~ } >> |
\time 3/2
<< { s1. } \\  
  { c'1 s2 } >> \bar "|." \break
```

LilyPond 原稿記述に於いては直下型 (コマンドの) 作用での原稿記述進行よりもむしろ、応用型思考 (コマンドの) 作用での原稿記述を行うと、美しき楽譜組版と著者が導かれます。

LilyPond の原稿記述は人それぞれの正解の書き方が多様にあります、上記の原稿記述は絶対的ではありません、著者 (私) のそのヶ所での発想の処理記述です。優れた方は全体を見渡した思考で書き出すでしょうし。

`\context StaffGroup <<.....>>` と `\context ChoirStaff <<.....>>` の絶対的な相違の言及です。簡易要約すると、幾つかの staff を "StaffGroup" or "ChoirGroup" の何方で括るかである? 括る形状の相違でなく、如実の違いが有るということです。

`\context ChoirStaff...` の括り方は `\hide Staff.BarLine` の指示が効かないという違いです。詳細は "notation.pdf (p-457, p-624)" を参照してください。

譜面参照としては (下記; クリック)

<https://www.sheet-poem.com/poem/html/2010/svg-avenue-wind/svg-avenue-wind.html>  
avenue-wind.pdf と avenue-wind-sheet.pdf の譜面を参照して下さい。



後が続く, おいおいの編集作業です m(\_ \_)m.

## LilyPond ver2.22.1 on windows10

# Lyric(Japan)

Windows 版 LilyPond は標準で和語=`msmincho,msgothic` にそのまま対応していますし、通常は和語歌詞は `gothic` 体で出力されます。

変更は、`\override Lyrics . LyricText #'font-name = "mincho"`

歌詞部分は utf-8 コードでの記述が求められています。

それ故に、楽譜記述は utf-8 で書き綴った方が何かと便利の気軽さがあります。

LilyPond の music Sheet での日本語対応は utf-8 コードです。他の和語、例えば和語書体は `sazanami`(フリー・ライセンス) を使う場合です、

但し、組版に使う活字体仕様は OTF(OpenType//TTF-TrueType を含む) に限ります

<http://sourceforge.jp/projects/efont/>

Lilypond への組み込みは (LilyPond デフォルト・インストールで; Windows10(64bit)),

(例: True Type = `sazanami font`)

C:\Program Files (x86)\LilyPond\usr\share\lilypond\current\fonts\otf\sazanami-mincho.ttf

C:\Program Files (x86)\LilyPond\usr\share\lilypond\current\fonts\otf\sazanami~gothic.ttf

格納するだけで自動認識します。又、

Windows OS の fonts 格納場所 (フォルダー) の中に置いても自動認識はします。

フォント名から====>LilyPond 和語指定コマンド書式は

**Lyrics**(歌詞パート) の中で、

明朝体組なら

```
\override Lyrics . LyricText #'font-name = "sazanami mincho"
```

ゴシック体組なら

```
\override Lyrics . LyricText #'font-name = "sazanami gothic"
```

を指定してから、和語本文 (歌詞) を綴ります。

**\markup** の中なら、

```
\override #'(font-name . "sazanami gothic") "和語文字列"
```

```
\override #'(font-name . "sazanami mincho") "和語文字列"
```

他の ttf, otf 仕様のフォントなら上記に準じて設定すれば対応するはずですが、

私 (著者) は和語は使わないので、試していません。

明治の文明開化で西洋様式化はしましたが...?? 古来から日本文化様式化は存在していたと思っています。邦楽の様式化は日本の場合は万人 (社会) へ開放ではなく 専門家系内の継承でしようけど (歴史家や音楽家でないので...わかりません)。

後続く、おいしいの編集作業です m(\_ \_)m.



## LilyPond ver2.22.1 on windows10

## context Score

始めに述べておきますが、Score {...} 環境の中で context Voice(音符群類) を直に書かない(著者はです)、たとえ単一(楽器)パート譜面を描くにしてもです。以下の A), B) 記述で例えを示します。

A); Score の中で、直に context Voice を記述する。

```

%% Time-stamp: "11/30/2017 6:01"
%% -*- Coding: utf-8 -*-
\version "2.20.0"
#(set-global-staff-size 16)
\paper{
paper-height = 80.00\mm
paper-width = 160.00\mm
page-top-space = 0.00\mm
top-margin = 0.00\mm
bottom-margin = 0.00\mm
}
\header {
title = \markup { \override #'(font-name . "gothic") "江戸の子守唄" }
composer = \markup { \override #'(font-name . "gothic") "子守唄" }
poet = \markup { \override #'(font-name . "gothic") "江戸地方民謡" }
}

\score {
\relative c' {\clef treble \key f\major \time 4/4
\set Staff.midiInstrument = "synthstrings 2"
a'2 a4 g~ | g8 a c4 a g | r8 e g[ a] c4 a | g2. r4 \bar "|" \break
c,4. c8 d4 c~ | c8 d e4 a g | r8 c, d4 e c | d2. r4 \bar "|." }

\addlyrics { \set stanza = "1."
\override Lyrics . LyricText #'font-name = "mincho"
しばのおりどのしず - がやに \break
おきなとおうながすまいけり }

\addlyrics { \set stanza = "2."
\override Lyrics . LyricText #'font-name = "gothic"
おきなはや - まへしば - かりに \break
おうなはか - わへきぬすすぎ }

\addlyrics { \set fontSize = #-3 \set stanza = "3."

```

```

\override Lyrics . LyricText #'font-name = "mincho"
      ぼうやはよいこだねん-ねしな \break
      ぼうやのおもりはどこへいった}

\addlyrics { \set fontSize = #-3 \set stanza = "4."
\override Lyrics . LyricText #'font-name = "gothic"
      あのやまこえてさとへいった \break
      さとのおみ-やになにもろた}

\layout {}
\midi {}
}

```

- 結果の譜面です.

## 江戸の子守唄

江戸地方民謡

子守唄

1. し ば の お り ど の し ず が や に  
 2. お き な は や ま へ し ば か り に  
 3. ぼ う や は よ い こ だ ね ん - ね し な  
 4. あ の や ま こ え て さ と へ い っ た

5  
 お き な と お う な が す ま い け り  
 お う な は か わ へ き ぬ す す り ぎ  
 ぼ う や の お も り は ど こ へ い っ た  
 さ と の お み - や に な に も ろ た

Music engraving by LilyPond 2.22.1—[www.lilypond.org](http://www.lilypond.org)

この例えは、`Score{.....}`の中で直に `context Voice` (音符群) を書き綴っています。他の解説などを紐解くと多くはこのような方法で記述されています。これは、通り一遍のサンプル記述であり(安易簡単な記述)、多くの方はこういう事柄からより複雑系へと LilyPond 提供の解説書の理解へと入っていきます。

```

B); context Voice を Score の中へ呼び込む記述.
%% -*- Coding: utf-8 -*-
\version "2.20.0"
#(set-global-staff-size 16)
\paper{
paper-height = 80.00\mm
paper-width = 160.00\mm
page-top-space = 0.00\mm
top-margin = 0.00\mm
bottom-margin = 0.00\mm
}
\header {
title = \markup { \override #'(font-name . "gothic") "江戸の子守唄" }
composer = \markup { \override #'(font-name . "gothic") "子守唄" }
poet = \markup { \override #'(font-name . "gothic") "江戸地方民謡" }
}

japan = \relative c' { a'2 a4 g ~ | g8 a c4 a g | r8 e g[ a ] c4 a | g2. r4 | \break
c,4. c8 d4 c ~ | c8 d e4 a g | r8 c, d4 e c | d2. r4 \bar "|." }

japanM = \relative c' { a'2 a4 g ~ | g8 a c4 a g | r8 e g[ a ] c4 a | g2. r4 | \break
c,4. c8 d4 c ~ | c8 d e4 a g | r8 c, d4 e c | d2. r4 \bar "|." }

japanVoiceL = \lyricmode { \set stanza = "1."
\override Lyrics . LyricText #'font-name = "mincho"
しばのおりどのしず- がやに \break
おきなとおうながすまいけり }
japanVoiceLII = \lyricmode { \set stanza = "2."
おきなはや- まへしば- かりに \break
おうなはか- わへきぬすすぎ }
japanVoiceLIII = \lyricmode { \set fontSize = #-3 \set stanza = "3."
\override Lyrics . LyricText #'font-name = "mincho"
ぼうやはよいこだねん- ねしな \break
ぼうやおもりはどこへいった }
japanVoiceLIV = \lyricmode { \set fontSize = #-3 \set stanza = "4."
\override Lyrics . LyricText #'font-name = "gothic"
あのやまこえてさとへいった \break
さとのおみ- やになにもろた }

%%% get sheet %%%
\score {
<<
\context Voice = "japanOne" { \clef treble \key f\major \time 4/4 \japan }
\new Lyrics = "mainL" \lyricsto "japanOne" { \japanVoiceL }
\new Lyrics = "mainL" \lyricsto "japanOne" { \japanVoiceLII }
\new Lyrics = "mainL" \lyricsto "japanOne" { \japanVoiceLIII }
\new Lyrics = "mainL" \lyricsto "japanOne" { \japanVoiceLIV }
>>
\layout { }
}
%%% get midi %%%
\score {
\context Staff { \clef treble \key f\major \time 4/4
\set Staff.midiInstrument = "synthstrings 2" \japanM }
\midi { }
}
%%% End All

```

- 結果の譜面です.

## 江戸の子守唄

江戸地方民謡

子守唄

1. し ば の お り ど の し ず が や に  
 2. お き な は や ま へ し ば か り に  
 3. ほ う や は よ い こ だ ね ん ね し な  
 4. あ の や ま こ え て さ と へ い っ た

5  
 お き な と お う な が す ま い け り  
 お う な は か わ へ き ぬ す す り ぎ  
 ほ う や の お も り は ど こ へ い っ た  
 さ と の お み や に な に も ろ た

Music engraving by LilyPond 2.22.1—[www.lilypond.org](http://www.lilypond.org)

譜面はまったく同じです.

context Voice(new Voice) と context Staff(new Staff) の理解は容易でしょう.

```
\context Voice = "japanOne" { \clef treble \key f\major \time 4/4 \japan }
```

処理スタックスの ID-name の名前付け・呼び込むのも、その部分の記述を context Voice で行うが理解できます.

楽譜組版へは context Staff で行っても Voice 記述での ID-name は引継ぎされません.

また、context Voice 記述を組版用と midi 用に同じ記述を行っています。これも、他意はありません.

譜面記述を簡素にするためです.

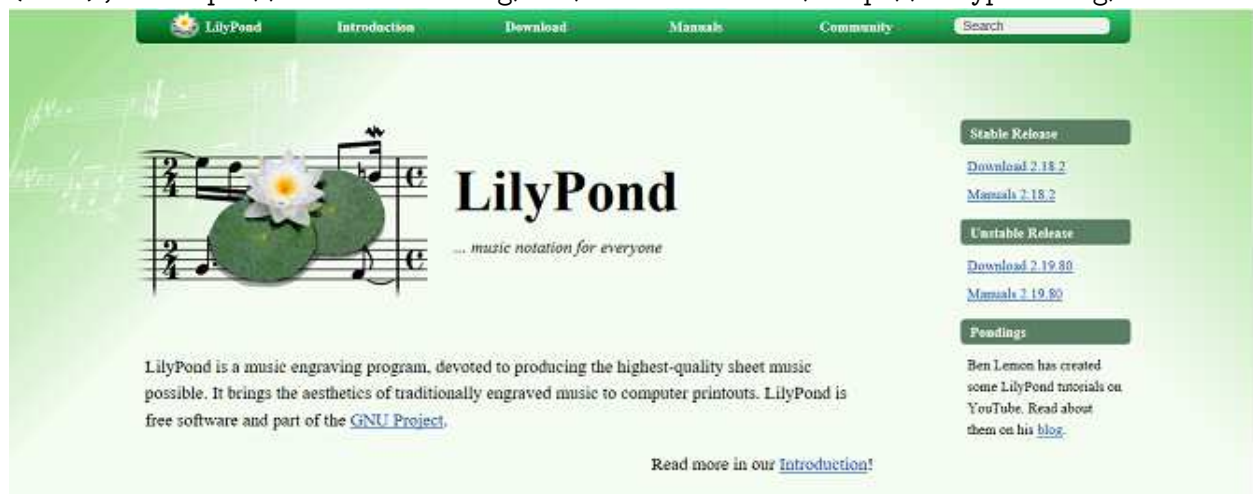
後続く、おいおいの編集作業です m(\_ \_)m.

# LilyPond(URL)

At the moment(12/4/ 2017); WEB ページです.

<http://lilypond.org/index.html>

( or ); <https://web.archive.org/web/20180516024136/http://lilypond.org/>



## .....FootPrints.....

$\text{T}_{\text{E}}\text{X}$  が公開され (1983 年 11 月頃), やがて, それをコアに  $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ , Texinfo, Musix $\text{T}_{\text{E}}\text{X}$ , PSTrics 等々が続々とヨーロッパやアメリカで開発されました. 日本では藤田先生によって  $\text{X}_{\text{M}}\text{T}_{\text{E}}\text{X}$  が化学者用に開発提供され世界へ発信されています.

Musix $\text{T}_{\text{E}}\text{X}$ 用のプリプロセッサ (M<sup>P</sup>)を開発していた, Mr.Jan Nieuwenhuizen が EJE(Eindhoven Youth Orchestra)の Mr.Han-Wen へリハーサル中に話しかけた(技術的問題の相談をした(1995))事がきっかけとなり, Mr.Han-Wen によって LilyPond 開発が誕生しています(1996).

Mr.Jan も開発に加わり, やがて, Mr.Han-Wen and Mr.Jan は幾度となく挫折と希望の谷間で戦いますが, とうとう, 失意の中で LilyPond 開発の断念寸前までに追いやられます. その時, Gnu の仲間たちが彼らを勇気づけし技術的困難の解決に参加し始めます. 開発が立ち上がってから 6 年後の 2002 年 Web URL に LilyPond が Ver1.6.0 としてリリースされました. — c++ の約 4,000 行以上に渡る拡張作業 —

## May it help you create lots of beautiful music!

この言葉は, Mr.Han-Wen and Mr.Jan が挫折と失意の中でも持ち続けた信仰なのかもしれません.

この期(約 1995 年頃からは, ネットワーク ftp:// に  $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$  の様な仕組みでのネットワークを CERN の物理学者達によって HTML として試作され, 最終的にアメリカのキャンパスで HTML を見るための Mosaic と呼ばれた Web ブラウザ が世に出て爆発的に http:// のインターネット推移の時期に Lilypond は静かに関係者によって進められていた!!.

全終了.





# Running Options(LilyPond)

Not yet drafted

