Ekmelily - Notation of Microtonal Music

Ekmelily is an extension for LilyPond that supports variable accidentals and key signatures for the notation of microtonal music in several equal-temperament tunings -- 12, 19, 24, 31, 36, 48, 53, 72-EDO -- and in 5-limit JI. For this purpose, it introduces predefined and user-defined notation styles. Each style describes a set of symbols for the alterations up to the five-quarters-tone, at most. Furthermore, Ekmelily defines own note names based on the names for semi- and quarter-tones given in LilyPond.
This documentation uses the Ekmelos font for all SMuFL glyph.

Ekmelily requires LilyPond version 2.19.22 or higher.

See Esmuflily for other music symbols: clefs, noteheads, flags, rests, articulations, dynamics, etc.

9 August 2024

Contents

Author and License

Ekmelily was written by Thomas Richter, thomas-richter@aon.at

Download and Installation

Download

The folder `ly` contains the include files.

- Copy the file(s) for the desired tuning(s) as well as the main include file `ekmel-main.ily` into an appropriate folder, e.g. `LILYPOND/usr/share/lilypond/current/ly` with `LILYPOND` meaning the installation folder of LilyPond.
- Optionally install a font , e.g. Ekmelos .

Note: To make use of a newly installed font in LilyPond prior to 2.24, its font cache, i.e. the folder `~/.lilypond-fonts.cache-2` (on Windows `%HOMEPATH%\.lilypond-fonts.cache-2`) must be emptied or completely deleted. So at the next execution of LilyPond this cache will be rebuilt from scratch inlcuding the new font.

Usage

Add the following lines near the top of your LilyPond input file. All commands are optional – except for `\include` of the desired tuning – but should be specified in this order.

```
ekmFont = FONTNAME
\include "ekmel..."
\language "LANGUAGE"
\ekmelicStyle STYLENAME
\ekmelicUserStyle USERSTYLENAME #'(
  (ALTERATION ELEMENT ...)
  ...
)
```

Ekmelily + Esmuflily

To combine Ekmelily with Esmuflily , add e.g. the following lines near the top of your LilyPond input file. This achieves LilyPond's standard behaviour, i.e. Dutch note names (default) and Stein / Couper accidentals (`stc`) for quarter-tones (24-EDO). The first line can be omitted for Ekmelos .

```
ekmFont = FONTNAME
\include "ekmel-24.ily"
\include "esmufl.ily"
\ekmelicStyle stc
```

Fonts

Ekmelily requires a font for accidentals defined by code point, character literal, or string. This applies to all accidentals in the predefined notation styles . Each of them is a SMuFL recommended character or Ekmelos specific optional glyph.

Ekmelily uses Ekmelos by default. Another font can be selected, either with the variable

```
ekmFont = FONTNAME
```

(or `ekmelicFont` as in previous versions) preceding the include file,
or with the command line option

```
-dekmfont=FONTNAME
```

(or `-dekmelic-font` as in previous versions). Note that this option produces a warning 'no such internal option', which can be ignored. Warnings can be suppressed with the command line option `--loglevel=ERROR` or `--loglevel=NONE` .

The glyphs from LilyPond's Emmentaler font can be used with markup in a user-defined notation style . See the Example `lilysingle` .

Drawing paths

Ekmelily supports drawing paths instead of font glyphs, which allows e.g. to produce stand-alone SVG output. This requires the Scheme procedure `ekm-path-stencil` as it is provided for Ekmelos by the include file `ly/ekmelos-paths.ily`.

A trailing `#` in FONTNAME switches to drawing paths, which effects all accidentals defined by code point or character literal, in particular, all accidentals in the predefined notation styles .

Note that spaces and other glyphs without a contour, as well as side-bearing and font features like stylistic alternates or ligatures are not available with paths.

To draw Ekmelos glyphs as paths, add the following lines near the top of your LilyPond input file. Note that a single `"#"` is equivalent to `"Ekmelos#"` .

```
ekmFont = "#"
\include "ekmelos-paths.ily"
\include "ekmel..."
...
```

Tunings

Ekmelily supports different tunings available as separate include files. Each provides its own set of languages and predefined notation styles . The first language and notation style specified in the following table is the default in the respective tuning. Some languages have alias names (in parentheses).

Tuning	Include file	Languages	Notation styles
12	`ekmel-12.ily`	`nederlands english deutsch català (catalan) español (espanol) italiano français português (portugues) norsk suomi svenska vlaams`	`std sag msag`
19	`ekmel-19.ily`	`nederlands english deutsch català (catalan) español (espanol) italiano français português (portugues) norsk suomi svenska vlaams`	`std sag msag`
24	`ekmel-24.ily`	`nederlands english deutsch català (catalan) español (espanol) italiano français português (portugues) norsk suomi svenska vlaams`	`stc stz go stvt arrow sag msag arabic persian four haba`
24	`ekmel-arabic.ily`	`italiano`	`arabic`
31	`ekmel-31.ily`	`nederlands deutsch español (espanol) français italiano português (portugues)`	`std sag msag stz sth`
36	`ekmel-36.ily`	`nederlands english deutsch norsk suomi svenska`	`go arrow sag msag wys bos haba`
48	`ekmel-48.ily`	`nederlands english deutsch`	`sag msag gostz`
53	`ekmel-53.ily`	`makam thm ktm english number`	`aeu aeuek thm sag dia`
72	`ekmel.ily`	`nederlands english deutsch norsk suomi svenska`	`arrow rhm sims hesse sag msag wys gostz gostc bos fern haba`
5JI	`ekmel-5ji.ily`	`nederlands`	`sag msag he`

Note: `ekmel-arabic.ily` is a variant of `ekmel-24.ily` for Arabic scores, like LilyPond's `arabic.ly` but with the correct accidentals U+ED30 - U+ED38. It supports Arabic maqamat and defines only the Italian language and the Arabic notation style, so the commands `\ekmelicStyle` and `\language` are not required.

Languages and Note names

Each tuning provides one or more languages for note names, which can be selected with the command

```
\language "LANGUAGE"
```

If LANGUAGE is not supported by the respective tuning, the default language is selected (usually `nederlands`). The note names are based on the names for semi- and quarter-tones given in LilyPond. See the Tables with all note names.

Enharmonically equivalent note names

Some notation styles support two distinct, enharmonically equivalent accidentals, e,g, `arrow` , `hesse` , `rhm` , and `sims` for the one-quarter-tone and the three-quarters-tone. Therefore, Ekmelily defines two note names each, e.g. `cqs` and `csaqf` (`english`) or `cih` and `ciseh` (`deutsch`). However, LilyPond does not support different accidentals for the same alteration. As a provisional solution, the combined note names like `csaqf` and `ciseh` have slightly differing alterations (+1/1024) and therefore cause inaccurate MIDI output.

NoteNames context

Ekmelily supports the `NoteNames` context and its properties. Note names can be drawn in any supported language by setting the `printNotesLanguage` property. Else, the language selected for music entry is used.

The format can be specified with the `printAccidentalNames` property. The first three values below are equivalent to LilyPond's normal behaviour.

`#t`	Scale name and accidental (default)
`#f`	Scale name
`'lily`	Note name
`'all`	All alias note names stacked vertically
`'alteration`	Alteration name
`'fraction`	Scale name and fraction of alteration
`'accidental`	Accidental

Predefined Notation Styles

Each tuning provides one or more predefined notation styles, which can be selected either with the command

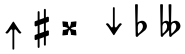
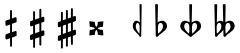
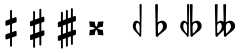
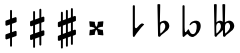
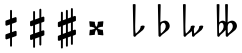
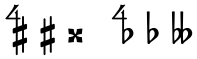
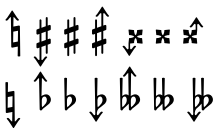
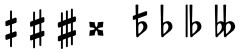
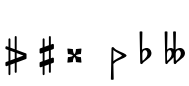
```
\ekmelicStyle STYLENAME
```

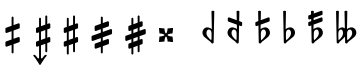
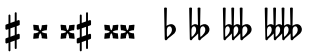
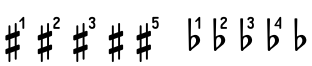
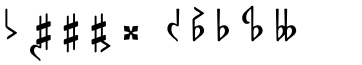
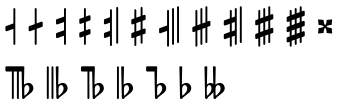
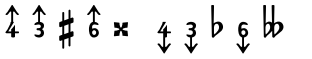
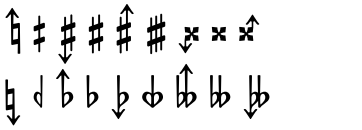
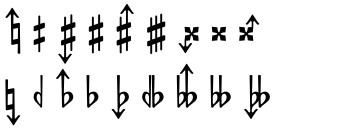
or with the command line option

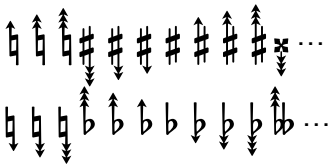
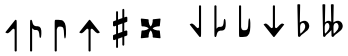
```
-dekmelic-style=STYLENAME
```

Note: This option produces a warning 'no such internal option', which can be ignored. Warnings can be suppressed with the command line option `--loglevel=ERROR` or `--loglevel=NONE` .
If STYLENAME is not supported by the respective tuning, the default notation style is selected.

The following table shows all predefined notation styles. N indicates the default style in tuning N.
All accidentals in these styles require a SMuFL compliant font . All optional glyphs are private supplements of Ekmelos . See the Tables with alterations, the Ekmelos Documentation , or SMuFL for details on the accidentals (code points, glyph names etc.)

Stylename		Tunings	Accidentals
`std`	Standard	12 19	
	Standard	31	
`stc`	Stein / Couper	24	
`stz`	Stein / Zimmermann	24 31	
`stvt`	Stein / Van Blankenburg / Tartini	24	
`sth`	Stein / Half flat Uses the optional glyphs U+F612 - U+F613 for the semi-flats.	31	
`four`	Digit 4	24	
`go`	Gould	24 36	
`arabic`	Arabic See `ekmel-arabic.ily` for Arabic maqamat.	24	
`persian`	Persian See Persian music notation for proper Persian microtonal alterations, note names, keys, etc.	24	

Stylename		Tunings	Accidentals
`aeu`	Arel-Ezgi-Uzdilek Uses the optional glyph U+F619 for the reversed slashed flat. See LilyPond's `turkish-makam.ly` for keys.	53	
`aeuek`	Arel-Ezgi-Uzdilek Equals `aeu` but does not use the reversed slashed flat, i.e. eksik-bakiye = koma.	53	
`dia`	Diatonic Uses the optional glyphs U+F61C - U+F61D for the quadruple sharp / flat.	53	
`thm`	Turkish folk music	53	
`arrow`	Arrow	24 36 72	
`bos`	Bosanquet commatic	36 72	
`haba`	Hába	24	
	Hába Uses the optional glyphs U+F660 - U+F670.	36 72	
`wys`	Wyschnegradsky	36 72	
`fern`	Ferneyhough	72	
`gostc`	Gould / Stein / Couper	72	
`gostz`	Gould / Stein / Zimmermann	48 72	
`hesse`	Hesse Uses the optional glyphs U+F606 - U+F60B for the degrees within the semitone.	72	
`rhm`	Richter Herf / Maedel Uses the optional glyphs U+F600 - U+F605 for the degrees within the semitone.	72	
`sims`	Sims	72	

Stylename		Tunings	Accidentals
`he`	Extended Helmholtz-Ellis	5JI	
`sag`	Sagittal	12 19 24 31 36 48 53 72 5JI	
`msag`	Mixed Sagittal Uses the large double sharp U+E47D. Hence it is different from `std` even for 12-EDO.	12 19 24 31 36 48 72 5JI	
`void`	No accidentals	all	

The universal notation style `void` defines no accidentals at all, except for the special symbols . As a consequence, the default accidental is drawn for each alteration. See the Example `numeric` which makes use of `void` .

User-defined Notation Styles

A new notation style derived from the currently selected style can be created with the command

```
\ekmelicUserStyle USERSTYLENAME #'(
  (ALTERATION ELEMENT ...)
  ...
)
```

USERSTYLENAME is a freely chosen style name. If it is an empty string `""` , the name of the currently selected style extended with the suffix `-user` is taken.

ALTERATION is a rational number or the name of a special symbol . For each specified alteration, a new symbol is defined which is the composition of the elements. For all other alterations where the previous symbol of ALTERATION appears likewise (usually combined with other symbols), this symbol is also replaced with the new one. Therefore, the order of alterations in the definition list can be significant, in particular, when a replaced symbol is again defined but for another alteration. See the Example `diaQuarter` .

ELEMENT is one of the following:

- a code point (integer), e.g. `#xE47B` for Wilson plus
- a character literal, e.g. `#\b` for flat
- a string, e.g. `"bb"` for double-flat
- markup, e.g. `,(markup #:semisharp)`

Two or more elements are juxtaposed with a padding of 0.12 staff units, but no extra space is inserted between the characters of a string. Note that in a SMuFL compliant font , all accidental glyphs have a zero side-bearing. This also applies to the Basic Latin (ASCII) characters in the Ekmelos font, so that e.g. `"bb"` is drawn without padding and `#\b #\b` with padding. See the Example `hewm` .

Special Symbols

Every notation style includes three special symbols. They can be user-defined like accidentals by specifying the respective name instead of an alteration.

```
\ekmelicUserStyle USERSTYLENAME #'(
  (leftparen ELEMENT ...)
  (rightparen ELEMENT ...)
  (default ELEMENT ...)
  ...
)
```

- `leftparen` : Left parenthesis for cautionary accidentals. Predefined is U+E26A.
- `rightparen` : Right parenthesis for cautionary accidentals. Predefined is U+E26B.
- `default` : Default accidental for alterations which are not defined in the selected notation style. This applies to the style `void` and to some styles in `ekmel-53.ily` , or it can occur with `\transpose` . Predefined is no symbol.

Additional Commands

`\ekmelicOutputSuffix`

Set the name of the selected notation style as the output filename suffix for the current `\book` section.

`\ekmelic-style-name`

Draw the name of the selected notation style as markup.

`\ekmelic-font-name`

Draw the name of the selected font as markup.

`\ekmelic-char ALTERATION`

Draw the accidental of ALTERATION (a rational number) according to the selected notation style as markup.
Used property:

- `font-size` (1)

`\ekmelic-elem ELEMENT`

Draw ELEMENT with the selected font as markup. ELEMENT is a code point (integer), a character literal, a string, or markup. This command is intended to combine glyphs from the selected font with other markup in a user-defined notation style .

`\ekmelic-fraction ALTERATION`

Draw ALTERATION (a rational number) as markup. If the denominator is 1, only the numerator is drawn.

`\ekmelic-fraction-small ALTERATION`

Draw ALTERATION (a rational number) as markup with a 3 steps smaller font size. If the denominator is 1, only the numerator is drawn, but with the current font size.
Used property:

- `font-size` (0)

`\ekm-fraction ARG1 ARG2`

Draw a fraction of ARG1 and ARG2 as markup. This command is a variant of LilyPond's `\fraction` with consistent vertical alignment. It is used by the above commands.
Used property:

- `font-size` (0)

`\ekmelic-table NATURAL COMPOSITE ORDER`

Draw a table of all accidentals in the selected notation style as markup, including the natural symbol if NATURAL is true, and all composite accidentals if COMPOSITE is true. Enharmonically equivalent accidentals and special symbols are always ignored. The accidentals are arranged in a row with the respective alteration placed beneath, and sorted by ascending or descending alteration if ORDER is 1 or -1, respectively, or by absolute alteration if ORDER is 2 or -2.
Used properties:

- `font-size` (0)
- `width` (4) : Horizontal extent for each accidental.
- `baseline-skip` : Distance between accidental and alteration.

Examples

See also the file `styles/user-styles.ly` for further examples of user-defined styles.

```
ekmFont = "Bravura"
\include "ekmel.ily"
\language "english"
\ekmelicStyle sims
```

Sets the predefined Sims notation for 72-EDO, selects the English note names, and draws the accidentals with the Bravura font.

```
\include "ekmel-24.ily"
\ekmelicStyle stz
\ekmelicUserStyle myNotation #'(
  (-3/4 #xE327)
  (1 #xE262 #xE262))
```

Sets the predefined Stein / Zimmermann notation (`stz`) for quarter-tones (24-EDO) and modifies it into a user-defined notation with the Sagittal flat 11 medium diesis down symbol 𝄫 (U+E327) for three-quarter-tones flat, and two sharp symbols 𝄪 (U+E262) for double-sharp.

```
\include "ekmel-24.ily"
\ekmelicStyle stz
\ekmelicUserStyle myNotation #`(
  (,THREE-Q-FLAT #xE327)
  (,DOUBLE-SHARP #xE262 #xE262))
```

This is the same example as above but it makes use of the corresponding Scheme symbols.

```
\include "ekmel.ily"
\language "english"
\ekmelicUserStyle hewm #'(
  (1 #\x)
  (-1 #\b #\b)
  (1/2 #\#)
  (-1/2 #\b)
  (1/4 #\^)
  (-1/4 #\v)
  (1/6 #\>)
  (-1/6 #\<)
  (1/12 #\+)
  (-1/12 #\-))
```

Sets the HEWM (Helmholtz / Ellis / Wolf / Monzo) notation for 72-EDO and selects the English note names. Note that double-flat is defined with `#\b #\b` which is drawn with a padding contrary to `"bb"` .

```
\include "ekmel-24.ily"
\ekmelicStyle stz
\ekmelicUserStyle stockhausen #'(
  (1/4 #xED58)
  (-1/4 #xED59)
  (3/4 #xED5A)
  (-3/4 #xED59 #xE260))
```

Sets the notation after Karlheinz Stockhausen for 24-EDO with the fractional sharp symbols  (U+ED58, U+ED5A) , and the quarter-tone flat symbol  (U+ED59) .

```
\include "ekmel.ily"
\ekmelicUserStyle smuflHesse #'(
  (1/4 #xE27A)
  (-1/4 #xE27B)
  (1/6 #xE2A4)
  (-1/6 #xE2A1)
  (1/12 #xE479)
  (-1/12 #xE47A))
```

Sets a variant of the Hesse notation for 72-EDO using SMuFL characters: Gould arrows  (U+E27A, U+E27B) , Sims half arrows  (U+E2A4, U+E2A1) , and Bosanquet commatic symbols (U+E479, U+E47A) . It is very similar to the Arrow notation.

```
\include "ekmel-24.ily"
\ekmelicUserStyle diaQuarter #'(
  (1 #xF61C)
  (-1 #xF61D)
  (3/4 #xE265)
  (-3/4 #xE266)
  (1/2 #xE263)
  (-1/2 #xE264)
  (1/4 #xE262)
  (-1/4 #xE260)
  (5/4 #xF61C #xE262)
  (-5/4 #xF61D #xE260))
```

Sets the Standard sharp / flat symbols, single thru quintuple, like the predefined Diatonic notation (`dia`) but for quarter-tones (24-EDO). The quadruple symbols   (U+F61C, U+F61D) are private supplements of the Ekmelos font. Note that here, the order of alterations is significant since the standard accidentals in the default notation (`stc`) are rearranged.

```
\include "ekmel-48.ily"
\ekmelicUserStyle lilysingle #`(
  (0    ,(markup #:natural))
  (1/8  ,(markup #:musicglyph "accidentals.natural.arrowup"))
  (-1/8 ,(markup #:musicglyph "accidentals.natural.arrowdown"))
  (1/4  ,(markup #:semisharp))
  (-1/4 ,(markup #:semiflat))
  (3/8  ,(markup #:musicglyph "accidentals.sharp.arrowdown"))
  (-3/8 ,(markup #:musicglyph "accidentals.flat.arrowup"))
  (1/2  ,(markup #:sharp))
  (-1/2 ,(markup #:flat))
  (5/8  ,(markup #:musicglyph "accidentals.sharp.arrowup"))
  (-5/8 ,(markup #:musicglyph "accidentals.flat.arrowdown"))
  (3/4  ,(markup #:sesquisharp))
  (-3/4 ,(markup #:sesquiflat))
  (7/8  ,(markup #:musicglyph "accidentals.sharp.slashslashslash.stemstem"))
  (-7/8 ,(markup #:musicglyph "accidentals.flatflat.slash"))
  (1    ,(markup #:doublesharp))
  (-1   ,(markup #:doubleflat))
  (leftparen ,(markup #:musicglyph "accidentals.leftparen"))
  (rightparen ,(markup #:musicglyph "accidentals.rightparen")))
```

Sets single glyphs from LilyPond's Emmentaler font for 48-EDO, as well as parentheses for cautionary accidentals. ±9/8 and ±5/4 are omitted here since they are set automatically to combinations of ±1 with ±1/8 and ±1/4.

```
#(define-markup-command (numeric-accidental layout props)
  ()
  (let ((alt (ly:chain-assoc-get 'alteration props 0)))
    (interpret-markup layout
      (cons '((font-size . -3)) props)
      (markup #:vcenter #:ekmelic-fraction alt))))

\ekmelicStyle void
\ekmelicUserStyle numeric #`(
  (default ,(markup #:numeric-accidental)))
```

Sets alteration values instead of accidental symbols. They are drawn as fraction, or integer if the denominator is 1, with the command `\ekmelic-fraction` and with a 3 steps smaller font size. This is applicable to all tunings. The `void` notation ensures that the default accidental is used for each alteration.