

## Esmuflily - SMuFL / Ekmelos for LilyPond

Esmuflily is an extension for [LilyPond](#) that supports [SMuFL](#) compliant fonts, in particular, to facilitate the use of glyphs from [Ekmelos](#) : clefs, time signatures, noteheads, articulations, etc.

Esmuflily provides [switches](#) to turn the SMuFL support on or off for individual types of graphical objects (clefs, noteheads, etc.) and it defines additional commands and styles for SMuFL glyphs which are not available in LilyPond (notehead styles, function theory symbols, etc.) So scores can benefit from both SMuFL's comprehensive character set and LilyPond's awesome Emmentaler font.

See [Ekmelily](#) for accidentals and key signatures.

Esmuflily requires LilyPond version 2.24.0 or higher.

26 March 2024

## Contents

---

Author and License . . . . .	3
Download and Installation . . . . .	4
Usage . . . . .	5
Fonts . . . . .	6
Commands . . . . .	7
SMuFL switches . . . . .	8
Clefs and clef modifiers . . . . .	10
Time signatures . . . . .	13
Cadenza signatures . . . . .	14
Staff dividers and separators . . . . .	15
Noteheads . . . . .	16
Shape noteheads . . . . .	21
Note name noteheads . . . . .	26
Note clusters . . . . .	27
Augmentation dots . . . . .	29
Flags and grace note slashes . . . . .	30
Rests . . . . .	32
Dynamics . . . . .	33
Scripts - Expressive marks . . . . .	35
Trill spans and pitches . . . . .	42
Laissez vibrer . . . . .	43
Breathing signs and caesuras . . . . .	44
Colon bar lines . . . . .	45
Segno bar lines . . . . .	46
Percent repeats . . . . .	47
Tremolo marks . . . . .	48
Symbols on stem . . . . .	49
Arpeggios . . . . .	50
Ottavation . . . . .	51
Tuplet numbers . . . . .	56
Fingering instructions . . . . .	58
String number indications . . . . .	60
Piano pedals . . . . .	61
Harp pedals . . . . .	63
Fret diagrams . . . . .	64
Accordion registers . . . . .	65
Accordion ricochet . . . . .	68
Falls and doigts . . . . .	69
Figured bass . . . . .	70
Lyrics . . . . .	72
Analytics symbols . . . . .	73
Function theory symbols . . . . .	74
Arrows and arrow heads . . . . .	80
Percussion symbols . . . . .	82
Electronic music symbols . . . . .	84
Other symbols . . . . .	85
General markup commands . . . . .	90
Extended text . . . . .	93
Definition string . . . . .	94
Orientation . . . . .	95

## Author and License

---

Esmuflily was written by Thomas Richter, [thomas-richter@aon.at](mailto:thomas-richter@aon.at)

Copyright © 2020-2024 Thomas Richter

Esmuflily is licensed under the [MIT License](#) .

This license is copied below, and is also available in the file `LICENSE.txt` , and at [mit-license.org](https://mit-license.org) .

The MIT License (MIT)

Copyright © 2020-2024 Thomas Richter

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## Download and Installation

---

### Download

The folder `ly` contains the include files.

- Copy the file `esmufl.ily` into an appropriate folder, e.g. `LILYPOND/usr/share/lilypond/current/ly` with `LILYPOND` meaning the installation folder of LilyPond.
- Optionally install a [SMuFL compliant font](#) , e.g. [Ekmelos](#) .

## Usage

---

Add the following lines near the top of your LilyPond input file.

```
ekmelicFont = FONTNAME  
\include "esmufl.ily"
```

### Esmuflily + Ekmelily

To combine Esmuflily with [Ekmelily](#) , 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 and Stein / Couper accidentals (`stc`) for quarter-tones (24-EDO). The first line can be omitted when using [Ekmelos](#) .

```
ekmelicFont = FONTNAME  
\include "esmufl.ily"  
\include "ekmel-24.ily"  
\language "nederlands"  
\ekmelicStyle stc
```

## Fonts

---

Esmuflily requires a [SMuFL](#) compliant font.

It uses [Ekmelos](#) by default. Another font can be selected, either with the variable

```
ekmelicFont = FONTNAME
```

**b e f o r e** the include file, or with the command line option

```
-dekmelic-font=FONTNAME
```

**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` .

## Commands

---

Most of the commands, in particular, all markup commands always produce SMuFL output, independent of any [switches](#) . Other commands behave differently when the corresponding switch is turned off:

[ [Ly](#) ]     Produces normal LilyPond output.

[ [Err](#) ]     Causes an error or produces useless output.

Some commands with a corresponding LilyPond command are simpler implemented, e.g. they ignore properties, while others provide additional features.

Some styles and commands make use of [Ekmelos](#) specific supplements, starting at code point U+F600, or assume the Ekmelos font metrics. Ancient symbols and styles are not supported. Most of the ancient glyphs are not implemented in [Ekmelos](#) .

Some commands and properties accept one of the following special values:

- [EXTEXT](#) : A code point, a list of code points, or markup.
- [DEFINITION](#) : A string of keys.
- [ORIENTATION](#) : Sum of axis and direction.

SMuFL glyphs are always accessed by code point (EXTEXT). See the file `ly/ekmelos-map.ily` at [Ekmelos](#) with definitions to access glyphs by name.

All commands have the prefix `ekm` or `ekm-` .

## SMuFL switches

---

```
\ekmSmuflOn #'TYPE
\ekmSmuflOn #'(TYPE ...)
\ekmSmuflOff #'TYPE
\ekmSmuflOff #'(TYPE ...)
```

Turn the SMuFL support on and off, respectively, for one or more types of graphical objects. TYPE is one of the following symbols. Any other value is ignored.

These commands set / undo context and grob properties (usually the stencil) in the current bottom context, except for `colon` and `segno` which are set independently of a context and cannot be turned off.

<code>all</code>	All following types
<code>clef</code>	Clefs and clef modifiers
<code>time</code>	Time signatures
<code>notehead</code>	Noteheads
<code>dot</code>	Augmentation dots
<code>flag</code>	Flags and grace note slashes
<code>rest</code>	Rests
<code>dynamic</code>	Absolute dynamic marks
<code>script</code>	Scripts
<code>lv</code>	Laissez vibrer
<code>trill</code>	Trill span and trill pitch
<code>colon</code>	Colon bar lines
<code>segno</code>	Segno bar lines
<code>percent</code>	Percent repeats
<code>tremolo</code>	Tremolos
<code>arpeggio</code>	Arpeggios
<code>tuplet</code>	Tuplet numbers
<code>fingering</code>	Fingering instructions
<code>stringnumber</code>	String number indications
<code>pedal</code>	Piano pedals
<code>fbass</code>	Figured bass
<code>lyric</code>	Lyric text



## Example:

Demonstrates possible places for SMuFL **switches** : a `\with` block, a `\layout` block, and in the music stream. Note that `\ekmTremolo` has no effect after the `tremolo` switch is turned off.

```
\score {
  \new Staff \with {
    \ekmSmuflOn #'trill
  }
  \relative c'' {
    \ekmSmuflOn #'notehead
    \override NoteHead.style = #'triangle
    c4 a
    \ekmSmuflOff #'notehead
    \revert NoteHead.style

    \autoBeamOff
    a8
    \ekmFlag #'straight
    a <a d> a16 <a d>

    \ekmPitchedTrill #'slash #'bracket
    d2 \ekmStartTrillSpan #-4 e d4 c \stopTrillSpan

    \ekmSmuflOn #'tremolo
    \ekmTremolo unmeasured { c4:16 a: }
    \ekmSmuflOff #'tremolo
    \ekmTremolo unmeasured { c4:16 a: }
  }
  \layout {
    \context {
      \Score
      \ekmSmuflOn #'flag
    }
  }
}
```



## Clefs and clef modifiers

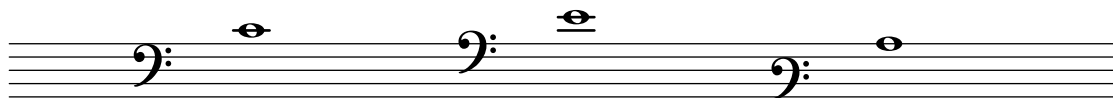
\ekmSmuflOn #'clef

Draw SMuFL clefs and clef modifiers (transposition and style).

G	G2	treble	violin	U+E050	gClef
french				:	
GG				U+E055	gClef8vbOld
tenorG				U+E056	gClef8vbCClef



F	bass	U+E062	fClef
subbass		:	
varbaritone	baritonevarF	:	



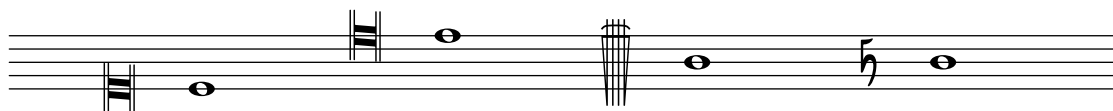
C	alto	U+E05C	cClef
soprano		:	
mezzosoprano		:	
tenor		:	
baritone		:	



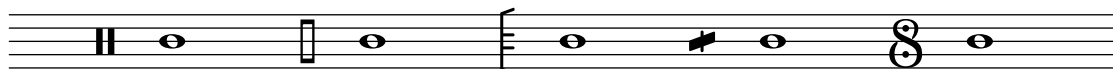
varC	altovarC	U+F633	cClefFrench20C
tenorvarC		:	
baritonevarC		:	



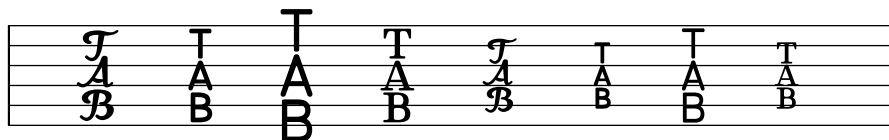
neomensural-c1 ... c5	U+E060	cClefSquare
bridge	U+E078	bridgeClef
accordion	U+E079	accdnDiatonicClef



percussion	U+E069	unpitchedPercussionClef1
varpercussion	U+E06A	unpitchedPercussionClef2
semipitched	U+E06B	semipitchedPercussionClef1
varsemipitched	U+E06C	semipitchedPercussionClef2
indiandrum	U+ED70	indianDrumClef



tab	U+F61E	6stringTabClefClassic
moderntab	U+E06D	6stringTabClef
talltab	U+F40A	6stringTabClefTall
seriftab	U+F40B	6stringTabClefSerif
4stringtab	U+F61F	4stringTabClefClassic
4stringmoderntab	U+E06E	4stringTabClef
4stringtalltab	U+F40C	4stringTabClefTall
4stringseriftab	U+F40D	4stringTabClefSerif



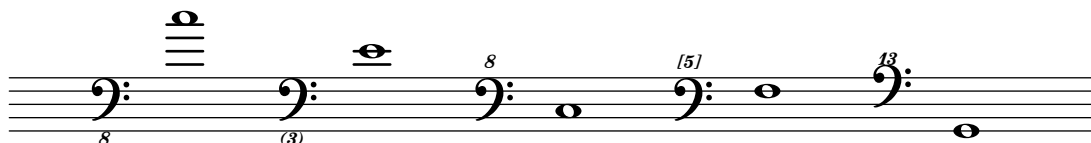
Clef modifiers (transposition and style) are always drawn separately, i.e. not with precomposed glyphs.

8	8	U+E07D	clef8
15	15	U+E07E	clef15
0	0	U+ED80	fingering0Italic
	:		
9	9	U+ED89	fingering9Italic
(	(	U+ED8A	fingeringLeftParenthesisItalic
)	)	U+ED8B	fingeringRightParenthesisItalic
[	[	U+ED8C	fingeringLeftBracketItalic
]	]	U+ED8D	fingeringRightBracketItalic

G<sub>8</sub>  
G<sub>15</sub>  
G<sub>(8)</sub>  
G<sup>2</sup>  
GG<sup>[4]</sup>



F<sub>8</sub>  
F<sub>(3)</sub>  
F<sup>8</sup>  
F<sup>[5]</sup>  
subbass<sup>13</sup>



C<sub>8</sub>  
C<sub>2</sub>  
tenorvarC<sup>7</sup>



Change clefs use special glyphs, except for `bridge` , `accordion` , and `indiandrum` which are drawn with a 2 steps smaller font size.

G	U+E07A	gClefChange
GG	U+F630	gClef8vbOldChange
tenorG	U+F631	gClef8vbCClefChange
F	U+E07C	fClefChange
C	U+E07B	cClefChange
varC	U+F634	cClefFrench20CChange
neomensural-c3	U+F632	cClefSquareChange



bridge	U+E078	bridgeClef
accordion	U+E079	accdnDiatonicClef
percussion	U+F635	unpitchedPercussionClef1Change
varpercussion	U+F636	unpitchedPercussionClef2Change
semipitched	U+F6BE	semipitchedPercussionClef1Change
varsemipitched	U+F6BF	semipitchedPercussionClef2Change
indiandrum	U+ED70	indianDrumClef



## Time signatures

\ekmSmuflOn #'time

Draw SMuFL time signatures.

\ekmCompoundMeter TIME-SIGNATURE

Set the numeric time signature.

\ekm-compound-meter TIME-SIGNATURE

Draw the numeric time signature as markup.

Compound meters use the large plus sign between fractions and the small plus sign between the numbers in a numerator. Some rational numbers can be part of a numerator. If specified in a pair, e.g. (1 . 1/2), this is treated as a single number without a plus sign in between.

4/4	<b>C</b>	U+E08A	timeSigCommon
2/2	<b>C</b>	U+E08B	timeSigCutCommon
0	<b>0</b>	U+E080	timeSig0
:	:		
9	<b>9</b>	U+E089	timeSig9
+	<b>+</b>	U+E08C	timeSigPlus
	+	U+E08D	timeSigPlusSmall
1/4	<b>¼</b>	U+E097	timeSigFractionQuarter
1/2	<b>½</b>	U+E098	timeSigFractionHalf
3/4	<b>¾</b>	U+E099	timeSigFractionThreeQuarters
1/3	<b>⅓</b>	U+E09A	timeSigFractionOneThird
2/3	<b>⅔</b>	U+E09B	timeSigFractionTwoThirds

#' (5 8)  
 #' ((2 8) (3 8))  
 #' (2 3 8)  
 #' (1 1/4 2)  
 #' ((1 . 1/4) 2))



#' ((2 4) (1 4) (1 8))  
 #' ((2 4) (2 1 8))  
 #' ((2 4) (1 1/2 4))



#' ((2 4) (3 8))  
 #' ((2 4) ((1 . 1/2) 4))  
 #' (2 (1 . 1/2) 4)



Cadenza signatures

---

\ekmCadenzaOn STYLE

Start a cadenza like \cadenzaOn and set a signature. The style can be one of the following symbols.

time-x

U+E09C

timeSigX

A musical staff with a five-line structure. The first measure contains a time signature 'X' followed by a half note on the first line. The second measure contains a quarter note on the second line. The third measure contains an eighth note on the third line. The fourth measure contains a quarter note on the fourth line. The fifth measure contains an eighth note on the fifth line. The sixth measure contains a quarter note on the first line. The seventh measure contains an eighth note on the second line. The eighth measure contains a quarter note on the third line. The ninth measure contains a half note on the fourth line. The tenth measure contains a half note on the fifth line.

time-penderecki

U+E09D




timeSigOpenPenderecki

A musical staff with a five-line structure. The first measure contains a time signature 'X' followed by a half note on the first line. The second measure contains a quarter note on the second line. The third measure contains an eighth note on the third line. The fourth measure contains a quarter note on the fourth line. The fifth measure contains an eighth note on the fifth line. The sixth measure contains a quarter note on the first line. The seventh measure contains an eighth note on the second line. The eighth measure contains a quarter note on the third line. The ninth measure contains a half note on the fourth line. The tenth measure contains a half note on the fifth line.

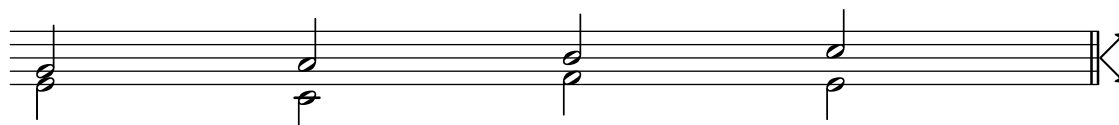
## Staff dividers and separators

`\ekmStaffDivider DIRECTION`

Draw the next barline with an indicator to split or recombine the staff and set a `\break`. The direction specifies the type of indicator (arrow).




#DOWN		U+E00B	staffDivideArrowDown
#UP		U+E00C	staffDivideArrowUp
#CENTER		U+E00D	staffDivideArrowUpDown

`\bar "||" \ekmStaffDivider #CENTER`



`system-separator-markup = \ekmSlashSeparator SIZE`

Draw a system separator mark of the specified size (set within a `\paper block`). SIZE is an integer in the range 0 thru 2.

#0		U+E007	systemDivider
#1		U+E008	systemDividerLong
#2		U+E009	systemDividerExtraLong

## Noteheads

\ekmSmuflOn #'notehead

Draw SMuFL noteheads. The style can be one of the following symbols. The `harmonic` and `cross` glyphs are also used with commands like `\harmonic` and `\xNote`.

default

U+F637 noteheadLongaUp  
 U+F638 noteheadLongaDown  
 U+F639 noteheadDoubleWholeAlt  
 U+E0A2 noteheadWhole  
 U+E0A3 noteheadHalf  
 U+E0A4 noteheadBlack



altdefault

:  
 U+E0A0 noteheadDoubleWhole  
 :



baroque

U+E0A1 noteheadDoubleWholeSquare  
 :



harmonic

U+E0D9 noteheadDiamondHalf



harmonic-black

U+E0DC noteheadDiamondBlackWide  
 U+E0DB noteheadDiamondBlack



harmonic-white

U+E0DE noteheadDiamondWhiteWide  
 U+E0DD noteheadDiamondWhite



harmonic-mixed

U+E0D7 noteheadDiamondDoubleWhole  
 U+E0D8 noteheadDiamondWhole  
 U+E0D9 noteheadDiamondHalf  
 U+E0DB noteheadDiamondBlack





harmonic-wide

U+E0D7 noteheadDiamondDoubleWhole  
 U+E0D8 noteheadDiamondWhole  
 U+E0DA noteheadDiamondHalfWide  
 U+E0DC noteheadDiamondBlackWide



diamond

U+E0DF noteheadDiamondDoubleWholeOld  
 U+E0E0 noteheadDiamondWholeOld  
 U+E0E1 noteheadDiamondHalfOld  
 U+E0E2 noteheadDiamondBlackOld



cross

U+E0A6 noteheadXDoubleWhole  
 U+E0A7 noteheadXWhole  
 U+E0A8 noteheadXHalf  
 U+E0A9 noteheadXBlack



xcircle

U+E0B0 noteheadCircleXDoubleWhole  
 U+E0B1 noteheadCircleXWhole  
 U+E0B2 noteheadCircleXHalf  
 U+E0B3 noteheadCircleX



withx

U+E0B4 noteheadDoubleWholeWithX  
 U+E0B5 noteheadWholeWithX  
 U+E0B6 noteheadHalfWithX  
 U+E0B7 noteheadVoidWithX



withx-black

U+E0B4 noteheadDoubleWholeWithX  
 U+E0B5 noteheadWholeWithX  
 U+E0B6 noteheadHalfWithX  
 U+F680 noteheadBlackWithX



plus

U+E0AC noteheadPlusDoubleWhole  
 U+E0AD noteheadPlusWhole  
 U+E0AE noteheadPlusHalf  
 U+E0AF noteheadPlusBlack



triangle

- U+E0BA noteheadTriangleUpDoubleWhole
- U+E0BB noteheadTriangleUpWhole
- U+E0BC noteheadTriangleUpHalf
- U+E0BE noteheadTriangleUpBlack
- U+E0C3 noteheadTriangleDownDoubleWhole
- U+E0C4 noteheadTriangleDownWhole
- U+E0C5 noteheadTriangleDownHalf
- U+E0C7 noteheadTriangleDownBlack



triangle-up



triangle-down



arrow

- U+E0ED noteheadLargeArrowUpDoubleWhole
- U+E0EE noteheadLargeArrowUpWhole
- U+E0EF noteheadLargeArrowUpHalf
- U+E0F0 noteheadLargeArrowUpBlack
- U+E0F1 noteheadLargeArrowDownDoubleWhole
- U+E0F2 noteheadLargeArrowDownWhole
- U+E0F3 noteheadLargeArrowDownHalf
- U+E0F4 noteheadLargeArrowDownBlack



arrow-up



arrow-down



slash

U+E10A noteheadSlashWhiteDoubleWhole  
 U+E102 noteheadSlashWhiteWhole  
 U+E103 noteheadSlashWhiteHalf  
 U+E101 noteheadSlashHorizontalEnds



slash-muted

U+E109 noteheadSlashWhiteMuted  
 U+E108 noteheadSlashHorizontalEndsMuted



slashed

U+E0D5 noteheadSlashedDoubleWhole1  
 U+E0D3 noteheadSlashedWhole1  
 U+E0D1 noteheadSlashedHalf1  
 U+E0CF noteheadSlashedBlack1



backslashed

U+E0D6 noteheadSlashedDoubleWhole2  
 U+E0D4 noteheadSlashedWhole2  
 U+E0D2 noteheadSlashedHalf2  
 U+E0D0 noteheadSlashedBlack2



circled

U+E0E7 noteheadCircledDoubleWhole  
 U+E0E6 noteheadCircledWhole  
 U+E0E5 noteheadCircledHalf  
 U+E0E4 noteheadCircledBlack



circled-large

U+E0EB noteheadCircledDoubleWholeLarge  
 U+E0EA noteheadCircledWholeLarge  
 U+E0E9 noteheadCircledHalfLarge  
 U+E0E8 noteheadCircledBlackLarge



parenthesised

U+F5DF noteheadDoubleWholeParens  
 U+F5DE noteheadWholeParens  
 U+F5DD noteheadHalfParens  
 U+F5DC noteheadBlackParens



U+E114	noteheadRoundWhite
U+E113	noteheadRoundBlack



U+E111	noteheadRoundWhiteLarge
U+E110	noteheadRoundBlackLarge



U+E115	noteheadRoundWhiteWithDot
U+E113	noteheadRoundBlack



U+E112	noteheadRoundWhiteWithDotLarge
U+E110	noteheadRoundBlackLarge



U+E119	noteheadRoundWhiteSlashed
U+E118	noteheadRoundBlackSlashed



U+E117	noteheadRoundWhiteSlashedLarge
U+E116	noteheadRoundBlackSlashedLarge



U+E0B8	noteheadSquareWhite
U+E0B9	noteheadSquareBlack



U+E11B	noteheadSquareBlackWhite
U+E11A	noteheadSquareBlackLarge



## Shape noteheads

All forms in LilyPond are supported, but some noteheads of Feta don't have exact matches in SMuFL, e.g. the thin shapes of `\southernHarmonyHeads` and the reversed shapes for stem up of `\funkHeads`.

### Sacred Harp

`\sacredHarpHeads`

fa	U+ECD3	noteShapeTriangleLeftDoubleWhole
	U+E1B6	noteShapeTriangleLeftWhite
	U+E1B7	noteShapeTriangleLeftBlack
	U+ECD2	noteShapeTriangleRightDoubleWhole
	U+E1B4	noteShapeTriangleRightWhite
	U+E1B5	noteShapeTriangleRightBlack
sol	U+ECD0	noteShapeRoundDoubleWhole
	U+E1B0	noteShapeRoundWhite
	U+E1B1	noteShapeRoundBlack
la	U+ECD1	noteShapeSquareDoubleWhole
	U+E1B2	noteShapeSquareWhite
	U+E1B3	noteShapeSquareBlack
mi	U+ECD4	noteShapeDiamondDoubleWhole
	U+E1B8	noteShapeDiamondWhite
	U+E1B9	noteShapeDiamondBlack



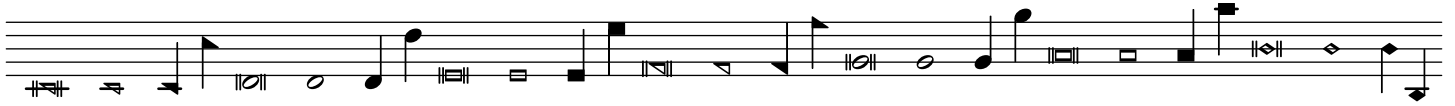
`\sacredHarpHeadsMinor`



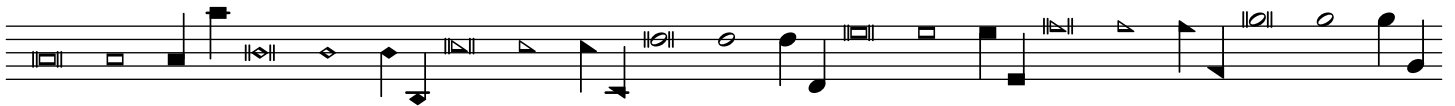
## Southern Harmony

\southernHarmonyHeads

fa	U+ECD3	noteShapeTriangleLeftDoubleWhole
	U+E1B6	noteShapeTriangleLeftWhite
	U+E1B7	noteShapeTriangleLeftBlack
	U+ECD2	noteShapeTriangleRightDoubleWhole
	U+E1B4	noteShapeTriangleRightWhite
	U+E1B5	noteShapeTriangleRightBlack
sol	U+ECD0	noteShapeRoundDoubleWhole
	U+E1B0	noteShapeRoundWhite
	U+E1B1	noteShapeRoundBlack
la	U+ECD1	noteShapeSquareDoubleWhole
	U+E1B2	noteShapeSquareWhite
	U+E1B3	noteShapeSquareBlack
mi	U+ECD4	noteShapeDiamondDoubleWhole
	U+E1B8	noteShapeDiamondWhite
	U+E1B9	noteShapeDiamondBlack



\southernHarmonyHeadsMinor



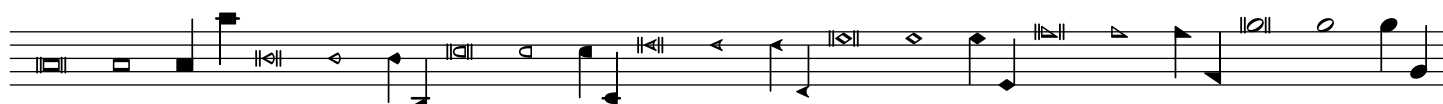
## Funk (Harmonia Sacra)

\funkHeads

do	U+ECDB	noteShapeMoonLeftDoubleWhole
	U+E1C6	noteShapeMoonLeftWhite
	U+E1C7	noteShapeMoonLeftBlack
re	U+ECDC	noteShapeArrowheadLeftDoubleWhole
	U+E1C8	noteShapeArrowheadLeftWhite
	U+E1C9	noteShapeArrowheadLeftBlack
mi	U+ECD4	noteShapeDiamondDoubleWhole
	U+E1B8	noteShapeDiamondWhite
	U+E1B9	noteShapeDiamondBlack
fa	U+ECD3	noteShapeTriangleLeftDoubleWhole
	U+E1B6	noteShapeTriangleLeftWhite
	U+E1B7	noteShapeTriangleLeftBlack
	U+ECD2	noteShapeTriangleRightDoubleWhole
	U+E1B4	noteShapeTriangleRightWhite
	U+E1B5	noteShapeTriangleRightBlack
sol	U+ECD0	noteShapeRoundDoubleWhole
	U+E1B0	noteShapeRoundWhite
	U+E1B1	noteShapeRoundBlack
la	U+ECD1	noteShapeSquareDoubleWhole
	U+E1B2	noteShapeSquareWhite
	U+E1B3	noteShapeSquareBlack
ti	U+ECDD	noteShapeTriangleRoundLeftDoubleWhole
	U+E1CA	noteShapeTriangleRoundLeftWhite
	U+E1CB	noteShapeTriangleRoundLeftBlack



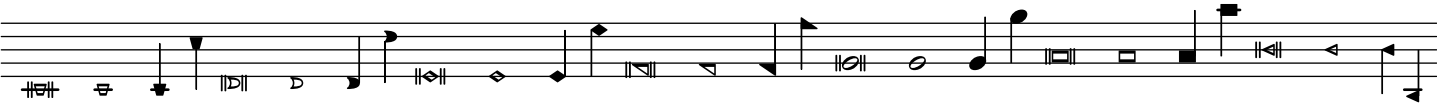
\funkHeadsMinor



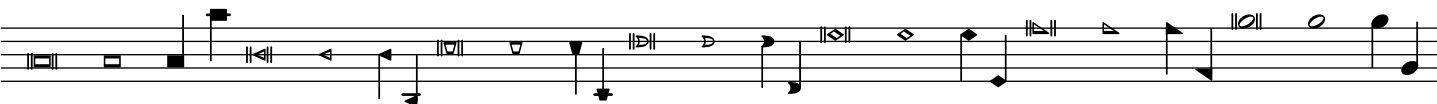
Walker

\walkerHeads

do	U+ECD8	noteShapeKeystoneDoubleWhole
	U+E1C0	noteShapeKeystoneWhite
	U+E1C1	noteShapeKeystoneBlack
re	U+ECD9	noteShapeQuarterMoonDoubleWhole
	U+E1C2	noteShapeQuarterMoonWhite
	U+E1C3	noteShapeQuarterMoonBlack
mi	U+ECD4	noteShapeDiamondDoubleWhole
	U+E1B8	noteShapeDiamondWhite
	U+E1B9	noteShapeDiamondBlack
fa	U+ECD3	noteShapeTriangleLeftDoubleWhole
	U+E1B6	noteShapeTriangleLeftWhite
	U+E1B7	noteShapeTriangleLeftBlack
	U+ECD2	noteShapeTriangleRightDoubleWhole
	U+E1B4	noteShapeTriangleRightWhite
	U+E1B5	noteShapeTriangleRightBlack
sol	U+ECD0	noteShapeRoundDoubleWhole
	U+E1B0	noteShapeRoundWhite
	U+E1B1	noteShapeRoundBlack
la	U+ECD1	noteShapeSquareDoubleWhole
	U+E1B2	noteShapeSquareWhite
	U+E1B3	noteShapeSquareBlack
ti	U+ECDA	noteShapelsoscelesTriangleDoubleWhole
	U+E1C4	noteShapelsoscelesTriangleWhite
	U+E1C5	noteShapelsoscelesTriangleBlack



\walkerHeadsMinor







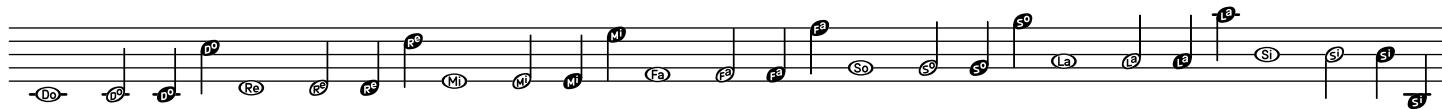
## Note name noteheads

\ekmNameHeads...

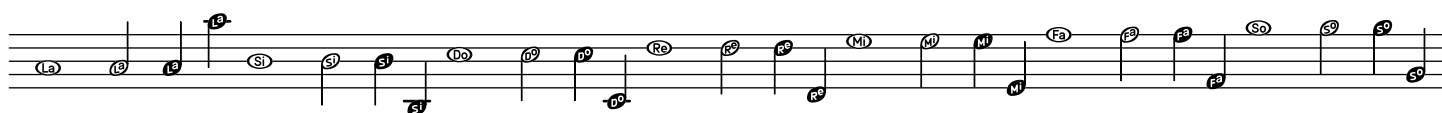
Draw noteheads with solfège (easy play) note names. [ Err ]

\ekmNameHeads

do	U+E150	noteDoWhole
	U+E158	noteDoHalf
	U+E160	noteDoBlack
re	U+E151	noteReWhole
	U+E159	noteReHalf
	U+E161	noteReBlack
mi	U+E152	noteMiWhole
	U+E15A	noteMiHalf
	U+E162	noteMiBlack
fa	U+E153	noteFaWhole
	U+E15B	noteFaHalf
	U+E163	noteFaBlack
so	U+E154	noteSoWhole
	U+E15C	noteSoHalf
	U+E164	noteSoBlack
la	U+E155	noteLaWhole
	U+E15D	noteLaHalf
	U+E165	noteLaBlack
si	U+E157	noteSiWhole
	U+E15F	noteSiHalf
	U+E167	noteSiBlack

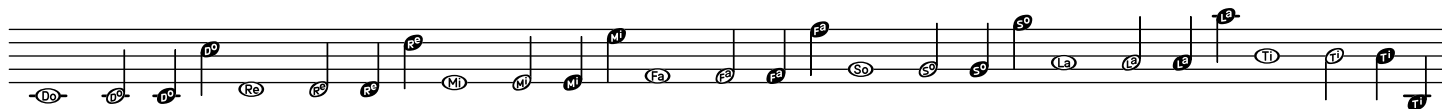


\ekmNameHeadsMinor

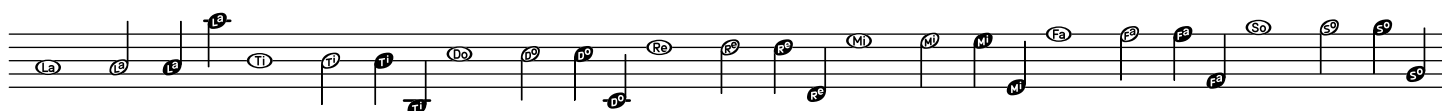


\ekmNameHeadsTi

do ... la	:	
ti	U+E156	noteTiWhole
	U+E15E	noteTiHalf
	U+E166	noteTiBlack



\ekmNameHeadsTiMinor



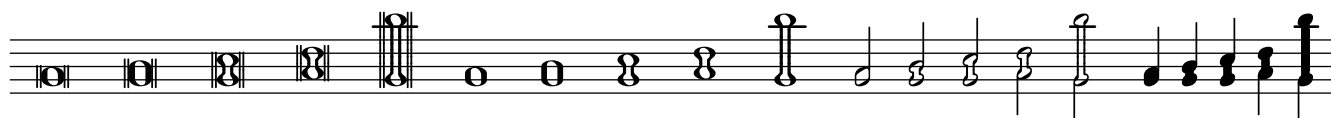
## Note clusters

\ekmMakeClusters MUSIC

Draw clusters instead of chords in MUSIC, consisting of a bottom and a top note head, and ignoring inner notes of the chords ('Cowell clusters'). The notehead style can be one of the following symbols.

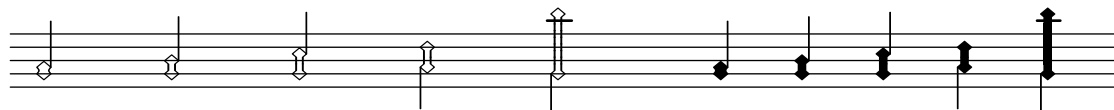
default

U+E124	noteheadClusterDoubleWhole2nd
U+E128	noteheadClusterDoubleWhole3rd
U+E12C	noteheadClusterDoubleWholeTop
U+E12D	noteheadClusterDoubleWholeMiddle
U+E12E	noteheadClusterDoubleWholeBottom
U+E125	noteheadClusterWhole2nd
U+E129	noteheadClusterWhole3rd
U+E12F	noteheadClusterWholeTop
U+E130	noteheadClusterWholeMiddle
U+E131	noteheadClusterWholeBottom
U+E126	noteheadClusterHalf2nd
U+E12A	noteheadClusterHalf3rd
U+E132	noteheadClusterHalfTop
U+E133	noteheadClusterHalfMiddle
U+E134	noteheadClusterHalfBottom
U+E127	noteheadClusterQuarter2nd
U+E12B	noteheadClusterQuarter3rd
U+E135	noteheadClusterQuarterTop
U+E136	noteheadClusterQuarterMiddle
U+E137	noteheadClusterQuarterBottom



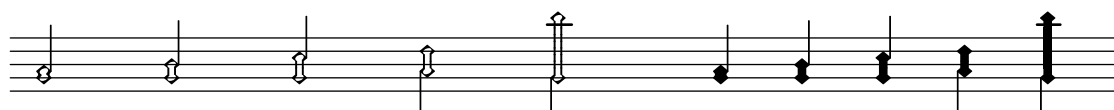
harmonic

U+E138	noteheadDiamondClusterWhite2nd
U+E13A	noteheadDiamondClusterWhite3rd
U+E13C	noteheadDiamondClusterWhiteTop
U+E13D	noteheadDiamondClusterWhiteMiddle
U+E13E	noteheadDiamondClusterWhiteBottom
U+E139	noteheadDiamondClusterBlack2nd
U+E13B	noteheadDiamondClusterBlack3rd
U+E13F	noteheadDiamondClusterBlackTop
U+E140	noteheadDiamondClusterBlackMiddle
U+E141	noteheadDiamondClusterBlackBottom



diamond

U+F64B	noteheadDiamondClusterHalf2nd
U+F64C	noteheadDiamondClusterHalf3rd
U+F64D	noteheadDiamondClusterHalfTop
U+F64E	noteheadDiamondClusterHalfMiddle
U+F64F	noteheadDiamondClusterHalfBottom
U+E139	noteheadDiamondClusterBlack2nd
U+E13B	noteheadDiamondClusterBlack3rd
U+E13F	noteheadDiamondClusterBlackTop
U+E140	noteheadDiamondClusterBlackMiddle
U+E141	noteheadDiamondClusterBlackBottom



square

U+E145

U+E146

U+E147

U+E142

U+E143

U+E144

noteheadRectangularClusterWhiteTop

noteheadRectangularClusterWhiteMiddle

noteheadRectangularClusterWhiteBottom

noteheadRectangularClusterBlackTop

noteheadRectangularClusterBlackMiddle

noteheadRectangularClusterBlackBottom

Note: For intervals larger than a third (except for `square`) the drawn cluster is a stack of one bottom segment, M middle segments, and one top segment. Mid and Top are the staff positions of the middle and top segments relative to the bottom segment.

Interval	M	Mid	Top
4th	0	-	3
5th	1	2	4
6th	2	2 3	5
7th	3	2 3 4	6
octave	4	2 3 4 5	7
...			

The segment glyphs in [Ekmelos](#) are designed for these values.

However, in the implementation notes of SMuFL Note clusters, the left-hand octave cluster is said to have 3 middle segments, while the right-hand 6th cluster has 2 middle segments. The “appropriate number of middle segments” varies apparently depending on the font.

## Augmentation dots

---

\ekmSmuflOn #'dot

Draw SMuFL augmentation dots.



# Flags and grace note slashes

```
\ekmSmuflOn #'flag
```

Draw SMuFL flags and grace note slashes.

```
\ekmFlag STYLE
```

Set the specified flag style. It actually overrides the properties `Flag.style` and `Stem.details.lengths`. The style can be one of the following symbols. Note that the glyphs for `short` stem down flags are [Ekmelos](#) specific. [ Err ]

default	U+E240	flag8thUp
	U+E241	flag8thDown
	U+E242	flag16thUp
	U+E243	flag16thDown
	U+E244	flag32ndUp
	U+E245	flag32ndDown
	U+E246	flag64thUp
	U+E247	flag64thDown
	U+E248	flag128thUp
	U+E249	flag128thDown
	U+E24A	flag256thUp
	U+E24B	flag256thDown
	U+E24C	flag512thUp
	U+E24D	flag512thDown
	U+E24E	flag1024thUp
	U+E24F	flag1024thDown



straight	U+F40F	flag8thUpStraight
	U+F411	flag8thDownStraight
	U+F412	flag16thUpStraight
	U+F414	flag16thDownStraight
	U+F415	flag32ndUpStraight
	U+F417	flag32ndDownStraight
	U+F418	flag64thUpStraight
	U+F41A	flag64thDownStraight
	U+F41B	flag128thUpStraight
	U+F41D	flag128thDownStraight
	U+F41E	flag256thUpStraight
	U+F420	flag256thDownStraight
	U+F421	flag512thUpStraight
	U+F423	flag512thDownStraight
	U+F424	flag1024thUpStraight
	U+F426	flag1024thDownStraight



short

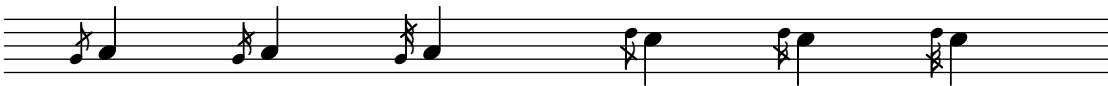
U+F410	flag8thUpShort
U+F6C0	flag8thDownShort
U+F413	flag16thUpShort
U+F6C1	flag16thDownShort
U+F416	flag32ndUpShort
U+F6C2	flag32ndDownShort
U+F419	flag64thUpShort
U+F6C3	flag64thDownShort
U+F41C	flag128thUpShort
U+F6C4	flag128thDownShort
U+F41F	flag256thUpShort
U+F6C5	flag256thDownShort
U+F422	flag512thUpShort
U+F6C6	flag512thDownShort
U+F425	flag1024thUpShort
U+F6C7	flag1024thDownShort



## Grace note slash

\slashedGrace

U+E564	graceNoteSlashStemUp
U+E565	graceNoteSlashStemDown



Draw SMuFL rests. The style can be one of the following symbols.

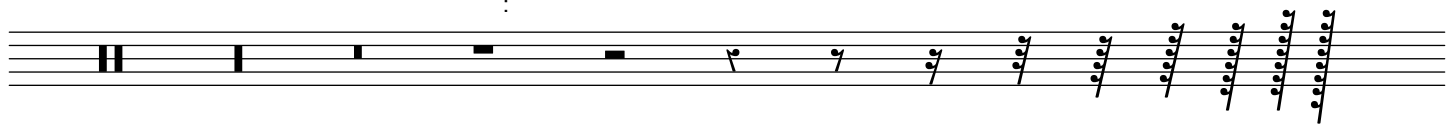
U+E4E0	restMaxima
U+E4E1	restLonga
U+E4E2	restDoubleWhole
U+E4E3	restWhole
U+E4E4	restHalf
U+E4E5	restQuarter
U+E4E6	rest8th
U+E4E7	rest16th
U+E4E8	rest32nd
U+E4E9	rest64th
U+E4EA	rest128th
U+E4EB	rest256th
U+E4EC	rest512th
U+E4ED	rest1024th



```

:
U+E4F2    restQuarterOld
:

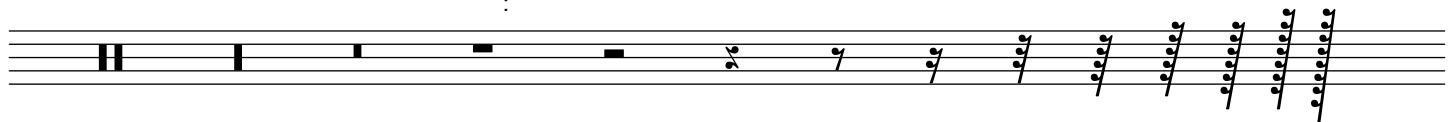
```



```

:
U+E4F6    restQuarterZ
:

```



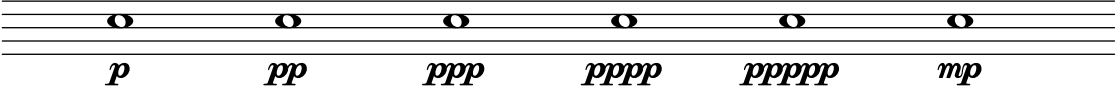


Dynamics

```
\ekmSmuflOn #'dynamic
```

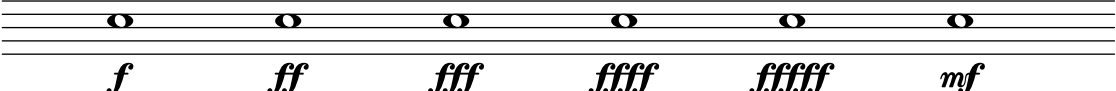
Draw SMuFL absolute dynamic marks.

<code>\p</code>	U+E520	dynamicPiano
<code>\pp</code>	U+E52B	dynamicPP
<code>\ppp</code>	U+E52A	dynamicPPP
<code>\pppp</code>	U+E529	dynamicPPPP
<code>\ppppp</code>	U+E528	dynamicPPPPP
<code>\mp</code>	U+E52C	dynamicMP




***p***      ***pp***      ***ppp***      ***pppp***      ***ppppp***      ***mp***

<code>\f</code>	U+E522	dynamicForte
<code>\ff</code>	U+E52F	dynamicFF
<code>\fff</code>	U+E530	dynamicFFF
<code>\ffff</code>	U+E531	dynamicFFFF
<code>\ffffff</code>	U+E532	dynamicFFFFFF
<code>\mf</code>	U+E52D	dynamicMF



***f***      ***ff***      ***fff***      ***ffff***      ***fffff***      ***mf***

<code>\fp</code>	U+E534	dynamicFortePiano
<code>\sf</code>	U+E536	dynamicSforzando1
<code>\sff</code>	U+F645	dynamicSforzandoFF
<code>\sfp</code>	U+E537	dynamicSforzandoPiano
<code>\sfz</code>	U+E539	dynamicSforzato
<code>\rfz</code>	U+E53D	dynamicRinforzando2
<code>\sp</code>	U+F646	dynamicSP
<code>\spp</code>	U+F647	dynamicSPP
<code>\n</code>	U+E526	dynamicNiente



***fp***      ***sf***      ***sff***      ***sfp***      ***sfz***      ***rfz***      ***sp***      ***spp***      ***n***

```
\ekm-dynamic DEFINITION
```

Draw a dynamic symbol as markup. **DEFINITION** may consist of the letters f, m, n, p, r, s, and z. The symbol is either a precomposed glyph or a sequence of dynamic glyphs for each letter.

```
\ekmParensDyn STYLE DYNAMIC-MARK
```

Draw the absolute dynamic mark parenthesized. The style can be one of the following symbols.

default	U+0028	parenleft
	U+0029	parenright
bracket	U+005B	bracketleft
	U+005D	bracketright
brace	U+007B	braceleft
	U+007D	braceright
angle	U+003C	less
	U+003E	greater

```
\ekmParensHairpin STYLE
```

Draw the subsequent hairpin parenthesized. The style can be one of the following symbols.

default	U+E542	dynamicHairpinParenthesisLeft
	U+E543	dynamicHairpinParenthesisRight
bracket	U+E544	dynamicHairpinBracketLeft
	U+E545	dynamicHairpinBracketRight
brace	U+007B	braceleft
	U+007D	braceright
angle	U+EA93	functionAngleLeft
	U+EA94	functionAngleRight

## Scripts - Expressive marks

```
\ekmSmuflOn #'script
```

Draw SMuFL scripts for expressive marks like articulations, ornamentations, performance indications, fermatas, repeat signs, etc.

```
\ekmScript #'NAME #'(EXTEXT-UP . EXTEXT-DOWN)
```

```
\ekmScript #'NAME EXTEXT
```

Create a script from **EXTEXT**, either a pair for up and down or a single value for both directions. If the latter is a list it must be enclosed in a list. NAME is the symbol of an existing script like `accent marcato trill turn upbow open lheel segno` etc. It determines the vertical positioning of the script. [\[ Ly \]](#)

```
\ekmScriptSmall #'NAME #'(EXTEXT-UP . EXTEXT-DOWN)
```

```
\ekmScriptSmall #'NAME EXTEXT
```

Create a script with a 3 steps smaller font size. [\[ Ly \]](#)

### Articulations

```
\accent U+E4A0 articAccentAbove
          U+E4A1 articAccentBelow
```

```
\ekmScript #'accent #'((#xE4A0 1) . (#xE4A1 1))
```

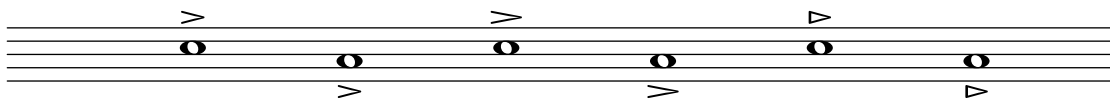
```
          U+F42A articAccentAboveLarge
```

```
          U+F42B articAccentBelowLarge
```

```
\ekmScript #'accent #'((#xE4A0 2) . (#xE4A1 2))
```

```
          U+F532 articAccentAboveRossini
```

```
          U+F533 articAccentBelowRossini
```

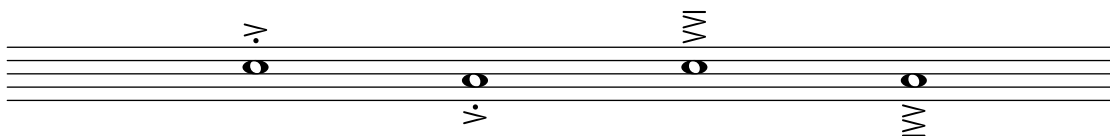


```
\ekmScript #'accent #'(#xE4B0 . #xE4B1)
          U+E4B0 articAccentStaccatoAbove
          U+E4B1 articAccentStaccatoBelow
```

```
\ekmScript #'accent #'((#xE4A4 #xE4A0 #xE4A0) .
                        (#xE4A5 #xE4A1 #xE4A1))
```

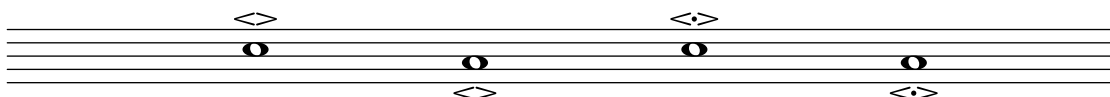
```
          U+F698 articTenutoDoubleAccentAbove
```

```
          U+F699 articTenutoDoubleAccentBelow
```



```
\espressivo U+ED40 articSoftAccentAbove
             U+ED41 articSoftAccentBelow
```

```
\ekmScript #'espressivo #'(#xED42 . #xED43)
          U+ED42 articSoftAccentStaccatoAbove
          U+ED43 articSoftAccentStaccatoBelow
```



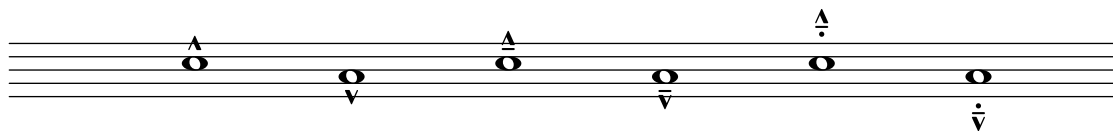
```

\marcato                U+E4AC  articMarcatoAbove
                        U+E4AD  articMarcatoBelow

\ekmScript #'marcato #'(#xE4BC . #xE4BD)
                        U+E4BC  articMarcatoTenutoAbove
                        U+E4BD  articMarcatoTenutoBelow

\ekmScript #'portato #'((#xE4AC #xE4A4 #xE4A2) .
                        (#xE4AD #xE4A5 #xE4A3))
                        U+F692  articMarcatoTenutoStaccatoAbove
                        U+F693  articMarcatoTenutoStaccatoBelow

```

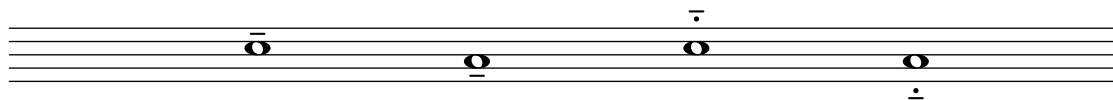


```

\tenuto                 U+E4A4  articTenutoAbove
                        U+E4A5  articTenutoBelow

\portato                U+E4B2  articTenutoStaccatoAbove
                        U+E4B3  articTenutoStaccatoBelow

```



```

\staccato               U+E4A2  articStaccatoAbove
                        U+E4A3  articStaccatoBelow

\staccatissimo          U+E4A6  articStaccatissimoAbove
                        U+E4A7  articStaccatissimoBelow

\ekmScript #'staccatissimo #'(#xE4A8 . #xE4A9)
                        U+E4A8  articStaccatissimoWedgeAbove
                        U+E4A9  articStaccatissimoWedgeBelow

\ekmScript #'staccatissimo #'(#xE4AA . #xE4AB)
                        U+E4AA  articStaccatissimoStrokeAbove
                        U+E4AB  articStaccatissimoStrokeBelow

```

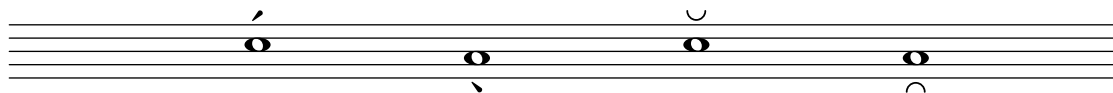


```

\ekmScript #'accent #'(#xE4B6 . #xE4B7)
                        U+E4B6  articStressAbove
                        U+E4B7  articStressBelow

\ekmScript #'accent #'(#xE4B8 . #xE4B9)
                        U+E4B8  articUnstressAbove
                        U+E4B9  articUnstressBelow

```

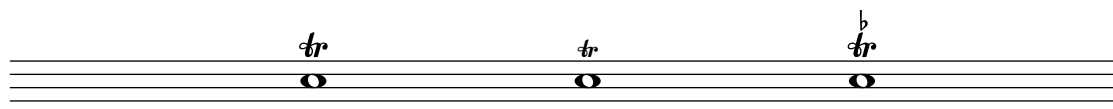


## Ornamentations

`\trill` U+E566 ornamentTrill

`\ekmScriptSmall #'trill ##xE566`  
U+E566 ornamentTrill

`\ekmScript #'trill #'((#xE260 #xE566))`  
U+F5BD ornamentTrillFlatAbove

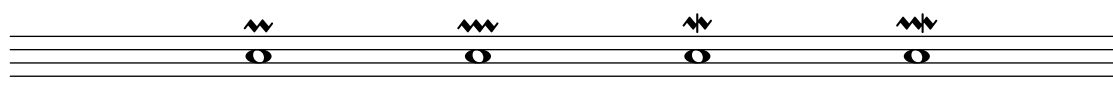


`\prall` U+E56C ornamentShortTrill

`\prallprall` U+E56E ornamentTremblement

`\mordent` U+E56D ornamentMordent

`\prallmordent` U+E5BD ornamentPrecompTrillWithMordent



`\upprall` U+E59A ornamentBottomLeftConcaveStroke

U+E59D ornamentZigZagLineNoRightEnd

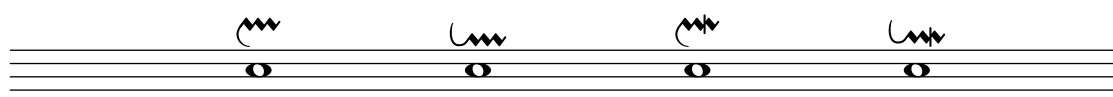
U+E59D ornamentZigZagLineNoRightEnd

U+E59E ornamentZigZagLineWithRightEnd

`\downprall` U+E5C6 ornamentPrecompMordentUpperPrefix

`\upmordent` U+E5B8 ornamentPrecompSlideTrillBach

`\downmordent` U+E5C7 ornamentPrecompInvertedMordentUpperPrefix



`\prallup` U+E59D ornamentZigZagLineNoRightEnd

U+E59D ornamentZigZagLineNoRightEnd

U+E59D ornamentZigZagLineNoRightEnd

U+E5A4 ornamentRightVerticalStroke

`\pralldown` U+E5C8 ornamentPrecompTrillLowerSuffix

`\lineprall` U+E5B2 ornamentPrecompAppoggTrill



`\turn` U+E567 ornamentTurn

`\reverseturn` U+E568 ornamentTurnInverted

`\slashturn` U+E569 ornamentTurnSlash

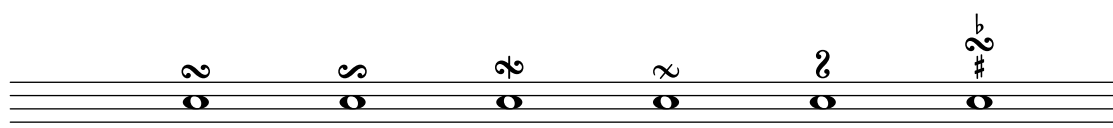
`\haydnturn` U+E56F ornamentHaydn

`\ekmScript #'turn ##xE56A`

U+E56A ornamentTurnUp

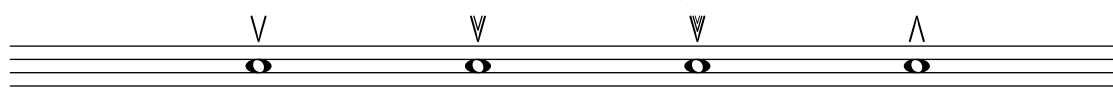
`\ekmScript #'turn #'((#xE260 #xE567 #xE262))`

U+F5C1 ornamentTurnFlatAboveSharpBelow

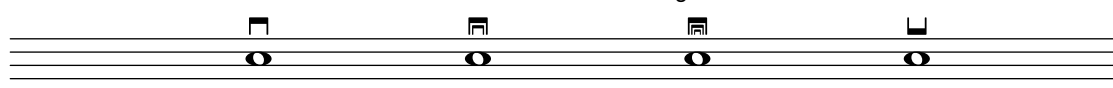


## Performance indications

<code>\upbow</code>	U+E612	stringsUpBow
<code>\ekmScript #'upbow ##xE61C</code>	U+E61C	stringsOverpressureUpBow
<code>\ekmScript #'upbow ##xE61E</code>	U+E61E	stringsOverpressurePossibleUpBow
<code>\ekmScript #'upbow ##xE613</code>	U+E613	stringsUpBowTurned



<code>\downbow</code>	U+E610	stringsDownBow
<code>\ekmScript #'downbow ##xE61B</code>	U+E61B	stringsOverpressureDownBow
<code>\ekmScript #'downbow ##xE61D</code>	U+E61D	stringsOverpressurePossibleDownBow
<code>\ekmScript #'downbow ##xE611</code>	U+E611	stringsDownBowTurned



<code>\ekmScript #'downbow ##xE626</code>	U+E626	stringsChangeBowDirection
<code>\ekmScript #'downbow #'((#xE626 1))</code>	U+F431	stringsChangeBowDirectionLiga
<code>\ekmScript #'downbow #'((#xE626 2))</code>	U+F43E	stringsChangeBowDirectionImposed



<code>\ekmScript #'upbow ##xE61F</code>	U+E61F	stringsOverpressureNoDirection
<code>\ekmScript #'downbow #'(#xE620 . #xE621)</code>	U+E620	stringsJeteAbove
	U+E621	stringsJeteBelow



<code>\flageolet</code>	U+E614	stringsHarmonic
<code>\ekmScriptSmall #'flageolet ##xE614</code>	U+E614	stringsHarmonic
<code>\open</code>	U+F63C	stringsOpen
<code>\halfopen</code>	U+F63D	stringsHalfOpen
<code>#(make-articulation 'halfopenvertical)</code>	U+F63E	stringsHalfOpenVertical

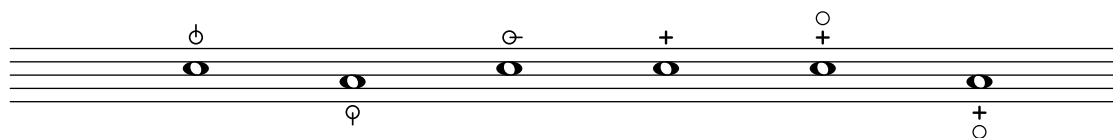


\snappizzicato                    U+E631    pluckedSnapPizzicatoAbove  
                                      U+E630    pluckedSnapPizzicatoBelow

\ekmScript #'snappizzicato ##xE632  
                                      U+E632    pluckedBuzzPizzicato

\stopped                            U+E633    pluckedLeftHandPizzicato

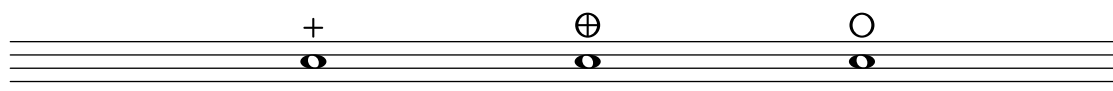
\ekmScript #'stopped #'((#xE614 #xE633) .  
                                      (#xE633 #xE614) )  
                                      U+F6AD    pluckedLeftHandPizzicatoHarmonicAbove  
                                      U+F6AE    pluckedLeftHandPizzicatoHarmonicBelow



\ekmScript #'stopped ##xE5E5  
                                      U+E5E5    brassMuteClosed

\ekmScript #'halfopen ##xE5E6  
                                      U+E5E6    brassMuteHalfClosed

\ekmScript #'open ##xE5E7  
                                      U+E5E7    brassMuteOpen



\lheel                                U+E661    keyboardPedalHeel1

\rheel                                U+E662    keyboardPedalHeel2

\ltoe                                 U+E664    keyboardPedalToe1

\rtoe                                 U+E665    keyboardPedalToe2

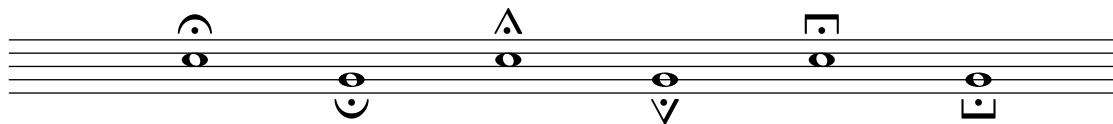
\ekmScript #'lheel ##xE663  
                                      U+E663    keyboardPedalHeel3

\ekmScript #'rtoe ##xE674  
                                      U+E674    keyboardPedalHeelToToe

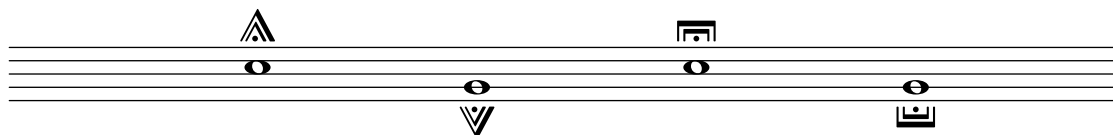


## Fermatas

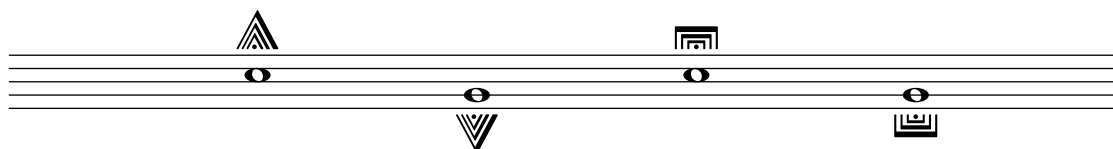
<code>\fermata</code>	U+E4C0	fermataAbove
	U+E4C1	fermataBelow
<code>\shortfermata</code>	U+E4C4	fermataShortAbove
	U+E4C5	fermataShortBelow
<code>\longfermata</code>	U+E4C6	fermataLongAbove
	U+E4C7	fermataLongBelow



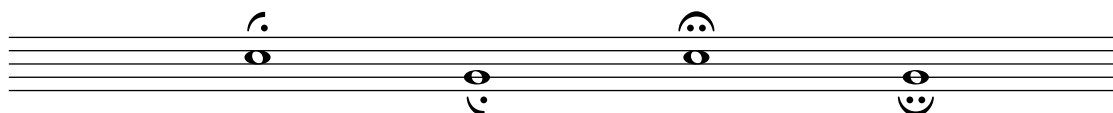
<code>\veryshortfermata</code>	U+E4C2	fermataVeryShortAbove
	U+E4C3	fermataVeryShortBelow
<code>\verylongfermata</code>	U+E4C8	fermataVeryLongAbove
	U+E4C9	fermataVeryLongBelow



<code>\ekmScript #'veryshortfermata #'</code> (#xF69E . #xF69F)	U+F69E	fermataExtraShortAbove
	U+F69F	fermataExtraShortBelow
<code>\ekmScript #'verylongfermata #'</code> (#xF6A0 . #xF6A1)	U+F6A0	fermataExtraLongAbove
	U+F6A1	fermataExtraLongBelow



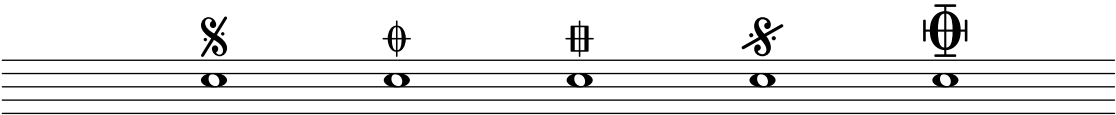
<code>\henzeshortfermata</code>	U+E4CC	fermataShortHenzeAbove
	U+E4CD	fermataShortHenzeBelow
<code>\henzelongfermata</code>	U+E4CA	fermataLongHenzeAbove
	U+E4CB	fermataLongHenzeBelow





Repeat signs

<code>\segno</code>	U+E047	segno
<code>\coda</code>	U+E048	coda
<code>\varcoda</code>	U+E049	codaSquare
<code>\ekmScript #'segno #'((#xE047 1))</code>	U+F404	segnoJapanese
<code>\ekmScript #'coda #'((#xE048 1))</code>	U+F405	codaJapanese



## Trill spans and pitches

`\ekmSmuflOn #'trill`

Draw SMuFL trill spans (trills with extender lines) and trill pitches.

`\ekmStartTrillSpan TEMPO`

Start a trill span. TEMPO is an integer in the range -4 thru 4 from slowest to fastest, i.e. from longest to shortest width of the extender line segments. [ [Ly](#) ]

`\startTrillSpan` is equivalent to `\ekmStartTrillSpan #0`

#0	U+E566	ornamentTrill
	U+EAA4	wiggleTrill

#-4	U+EAA8	wiggleTrillSlowest
#-3	U+EAA7	wiggleTrillSlowerStill
#-2	U+EAA6	wiggleTrillSlower
#-1	U+EAA5	wiggleTrillSlow

#1	U+EAA3	wiggleTrillFast
#2	U+EAA2	wiggleTrillFaster
#3	U+EAA1	wiggleTrillFasterStill
#4	U+EAA0	wiggleTrillFastest

`\ekmPitchedTrill NOTEHEAD-STYLE PARENS-STYLE`  
`MAIN-NOTE AUXILIARY-NOTE`

Draw a trill pitch. For NOTEHEAD-STYLE see [Noteheads](#) . PARENS-STYLE can be one of the following symbols. Variable accidentals for auxiliary notes are supported by [Ekmelily](#) . [ [Err](#) ]

default	U+E26A	accidentalParensLeft
	U+E26B	accidentalParensRight
bracket	U+E26C	accidentalBracketLeft
	U+E26D	accidentalBracketRight
brace	U+F6D4	accidentalBraceLeft
	U+F6D5	accidentalBraceRight
angle	U+F6D6	accidentalAngleLeft
	U+F6D7	accidentalAngleRight

# Laissez vibrer

---

```
\ekmSmuflOn #'lv
```

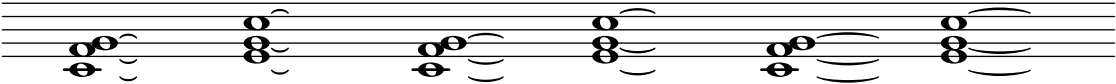
Draw SMuFL laissez vibrer ties.

```
\ekmLaissezVibrer SIZE
```

Draw a laissez vibrer tie after a note. SIZE is an integer in the range 0 thru 2. [ Ly ]

```
\laissezVibrer is equivalent to \ekmLaissezVibrer #0
```

#0	U+E4BA	articLaissezVibrerAbove
	U+E4BB	articLaissezVibrerBelow
#1	U+F6FC	articLaissezVibrerAboveLong
	U+F6FD	articLaissezVibrerBelowLong
#2	U+F6FE	articLaissezVibrerAboveExtraLong
	U+F6FF	articLaissezVibrerBelowExtraLong



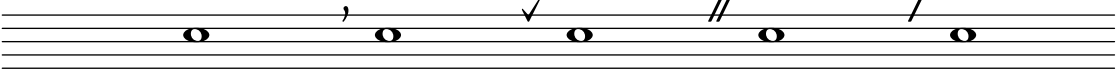
Breathing signs and caesuras

---

\ekmBreathing EXTEXT

Draw a breathing sign or caesura from EXTEXT .

##xE4CE	U+E4CE	breathMarkComma
##xE4CF	U+E4CF	breathMarkTick
##xE4D1	U+E4D1	caesura
#' (##xE4D1 1)	U+F42C	caesuraSlashSingleStroke

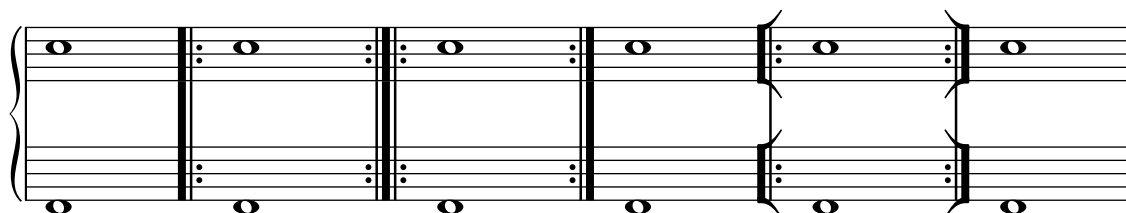


## Colon bar lines

```
\ekmSmuflOn #'colon
```

Draw SMuFL colon (repeat) bar lines. Note that `colon` is set independently of a context and cannot be turned off. Therefore, the SMuFL colon symbol is also drawn at the segno bar lines on the next page.

```
\bar ".|:"           U+E043   repeatDots
\bar ":|.|:"
\bar ":|. "
\bar "[|:"
\bar "[:|]"
```



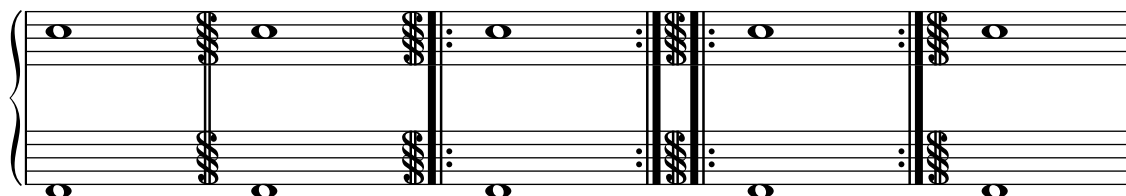
## Segno bar lines

---

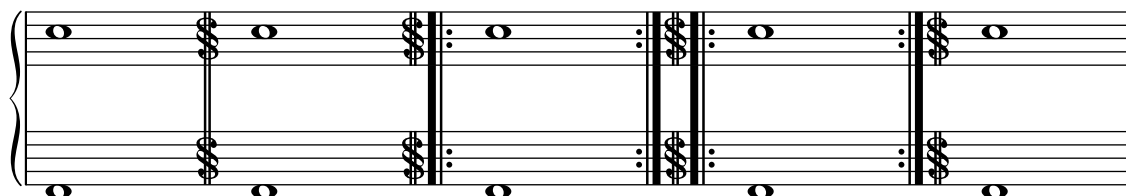
\ekmSmuflOn #'segno

Draw SMuFL segno bar lines. Note that `segno` is set independently of a context and cannot be turned off.

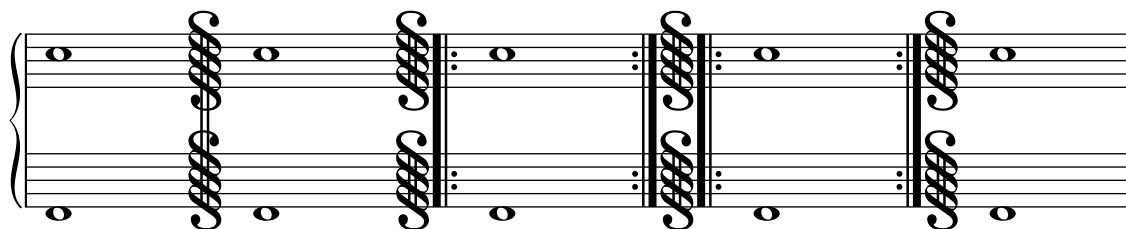
```
\bar "S"                                U+E04A  segnoSerpent1
\bar "S. | :-S"
\bar " : | .S. | :-S"
\bar " : | .S-S"
```



```
\bar "s"                                U+F6C8  segnoSerpentSmall1
\bar "s. | :-s"
\bar " : | .s. | :-s"
\bar " : | .s-s"
```



```
\bar "$"                                U+F6CA  segnoSerpentLarge1
\bar "$. | :- $"
\bar " : | .$. | :- $"
\bar " : | .$.- $"
```



## Percent repeats

---

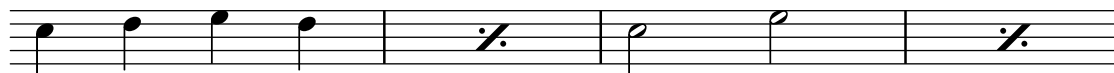
\ekmSmuflOn #'percent

Draw SMuFL percent repeats.

\repeat percent 4 {...} U+E504 repeatBarSlash



\repeat percent 2 {...} U+E500 repeat1Bar



\repeat percent 2 {...} U+E501 repeat2Bars



## Tremolo marks

`\ekmSmuflOn #'tremolo`

Draw SMuFL tremolo marks on stems. The style (shape) can be one of the following symbols. Note: The symbol `ekm` is used internally by `\ekmTremolo` (see below).

beam-like	:8	U+E220	tremolo1
	:16	U+E221	tremolo2
	:32	U+E222	tremolo3
	:64	U+E223	tremolo4
	:128	U+E224	tremolo5



fingered	:8	U+E225	tremoloFingered1
	:16	U+E226	tremoloFingered2
	:32	U+E227	tremoloFingered3
	:64	U+E228	tremoloFingered4
	:128	U+E229	tremoloFingered5



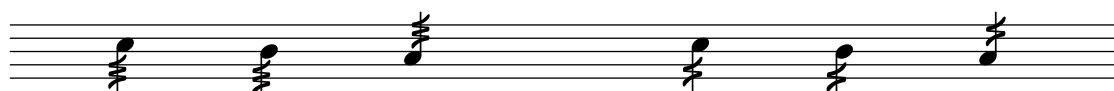
`\ekmTremolo EXTEXT MUSIC`

Draw a tremolo mark from [EXTEXT](#) on the stems of the tremolo notes in **MUSIC**, independent of the subdivision `:N`. The following names (strings) draw predefined symbols. A list of code points or a markup is centered horizontally, while a single code point is assumed being a centered stem decoration. [\[ Ly \]](#)

buzzroll	U+E22A	buzzRoll
penderecki	U+E22B	pendereckiTremolo
stockhausen	U+E232	stockhausenTremolo

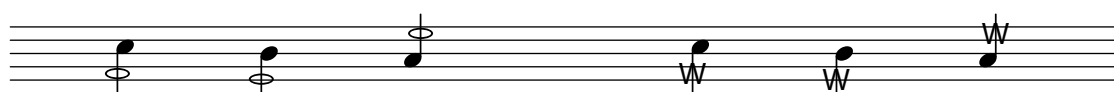


unmeasured	U+E22C	unmeasuredTremolo
unmeasuredS	U+E22D	unmeasuredTremoloSimple



<code>##xE233</code>	U+E233	oneHandedRollStevens
----------------------	--------	----------------------

`\markup { \sans "W" }`






## Symbols on stem (stem decoration)


\ekmStem EXTEXT MUSIC

Draw a symbol from **EXTEXT** vertically centered on the stems in **MUSIC**. The following names (strings) draw predefined symbols. A list of code points or a markup is centered horizontally, while a single code point is assumed being a centered stem decoration.


sprechgesang	U+E645	vocalSprechgesang
halbGesungen	U+E64B	vocalHalbGesungen
sussurando	U+E646	vocalsSussurando



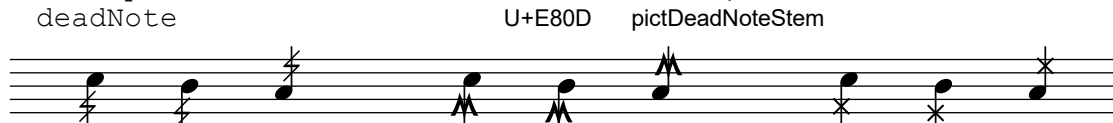
bowBehindBridge	U+E618	stringsBowBehindBridge
bowOnBridge	U+E619	stringsBowOnBridge
bowOnTailpiece	U+E61A	stringsBowOnTailpiece



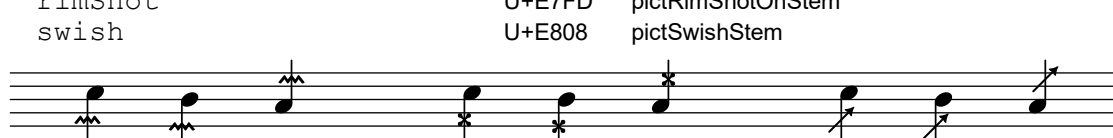
fouette	U+E622	stringsFouette
vibrato	U+E623	stringsVibratoPulse
damp	U+E63B	pluckedDampOnStem




stringNoise	U+E694	harpStringNoiseStem
multiphonics	U+E607	windMultiphonicsBlackStem
deadNote	U+E80D	pictDeadNoteStem



crush	U+E80C	pictCrushStem
rimShot	U+E7FD	pictRimShotOnStem
swish	U+E808	pictSwishStem

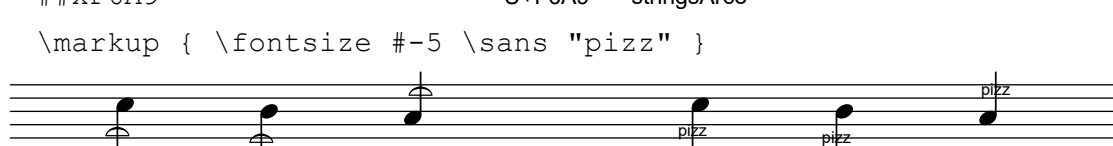


turnRight	U+E809	pictTurnRightStem
turnLeft	U+E80A	pictTurnLeftStem
turnRightLeft	U+E80B	pictTurnRightLeftStem



##xF6A9	U+F6A9	stringsArco
---------	--------	-------------




\markup { \fontsize #-5 \sans "pizz" }



## Arpeggios

```
\ekmSmuflOn #'arpeggio
```

Draw SMuFL arpeggios.

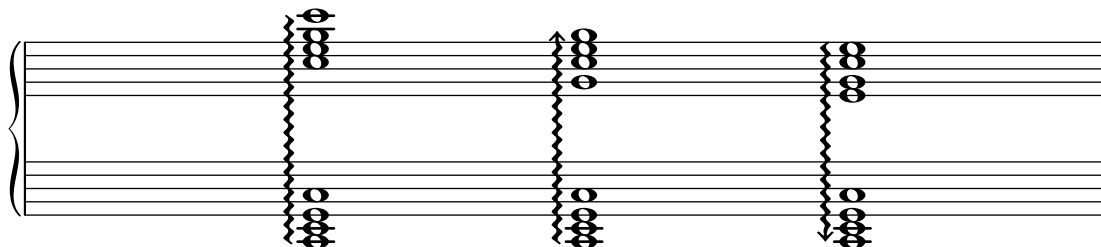
<code>\arpeggio</code>	U+EAA9	wiggleArpeggiatoUp
		
<code>\arpeggioArrowUp</code>	U+EAA9 U+EAAD	wiggleArpeggiatoUp wiggleArpeggiatoUpArrow
		
<code>\arpeggioArrowDown</code>	U+EAAA U+EAAE	wiggleArpeggiatoDown wiggleArpeggiatoDownArrow
		

Cross-staff arpeggios with

```
\set PianoStaff.connectArpeggios = ##t
```

**Note:** Instead of `\arpeggioArrowUp|Down`, they require

```
\override PianoStaff.Arpeggio.arpeggio-direction = #UP|#DOWN
```



## Ottavation

The following predefined lists of ottavation texts support the octave numbers  $\pm 1, 2, 3, 4$ .

### ekm-ottavation-numbers

$\pm 1$	U+E510	ottava
$\pm 2$	U+E514	quindicesima



$\pm 3$	U+E517	ventiduesima
$\pm 4$	U+F6F8	ventinovesima



### ekm-ottavation-ordinals

1	U+E511	ottavaAlta
-1	U+E512	ottavaBassa
2	U+E515	quindicesimaAlta
-2	U+E516	quindicesimaBassa



3	U+E518	ventiduesimaAlta
-3	U+E519	ventiduesimaBassa
4	U+F6F9	ventinovesimaAlta
-4	U+F6FA	ventinovesimaBassa



## ekm-ottavation-simple-ordinals

1	U+E510	ottava
	U+EC97	octaveBaselineV
	U+EC91	octaveBaselineA
-1	U+E51C	ottavaBassaVb
2	U+E514	quindicesima
	U+EC95	octaveBaselineM
	U+EC91	octaveBaselineA
-2	U+E51D	quindicesimaBassaMb



3	U+E517	ventiduesima
	U+EC95	octaveBaselineM
	U+EC91	octaveBaselineA
-3	U+E51E	ventiduesimaBassaMb
4	U+F6F8	ventinovesima
	U+EC95	octaveBaselineM
	U+EC91	octaveBaselineA
-4	U+F6FB	ventinovesimaBassaMb



## ekm-ottavation-ordinals-b

1	U+E511	ottavaAlta
-1	U+E51C	ottavaBassaVb
2	U+E515	quindicesimaAlta
-2	U+E51D	quindicesimaBassaMb



3	U+E518	ventiduesimaAlta
-3	U+E51E	ventiduesimaBassaMb
4	U+F6F9	ventinovesimaAlta
-4	U+F6FB	ventinovesimaBassaMb



## ekm-ottavation-ordinals-bassa

1	U+E511	ottavaAlta
-1	U+E512	ottavaBassa
	U+2009	thinspace
	U+E51F	octaveBassa
2	U+E515	quindicesimaAlta
-2	U+E516	quindicesimaBassa
	U+2009	thinspace
	U+E51F	octaveBassa



3	U+E518	ventiduesimaAlta
-3	U+E519	ventiduesimaBassa
	U+2009	thinspace
	U+E51F	octaveBassa
4	U+F6F9	ventinovesimaAlta
-4	U+F6FA	ventinovesimaBassa
	U+2009	thinspace
	U+E51F	octaveBassa



## ekm-ottavation-ordinals-ba

1	U+E511	ottavaAlta
-1	U+E513	ottavaBassaBa
2	U+E515	quindicesimaAlta
-2	U+E514	quindicesima
	U+EC93	octaveBaselineB
	U+EC91	octaveBaselineA



3	U+E518	ventiduesimaAlta
-3	U+E517	ventiduesima
	U+EC93	octaveBaselineB
	U+EC91	octaveBaselineA
4	U+F6F9	ventinovesimaAlta
-4	U+F6F8	ventinovesima
	U+EC93	octaveBaselineB
	U+EC91	octaveBaselineA



ekm-ottavation-numbers-ba

- 1            U+E510    ottava
- 1          U+E513    ottavaBassaBa
- 2            U+E514    quindicesima
- 2          U+E514    quindicesima
- U+EC93    octaveBaselineB
- U+EC91    octaveBaselineA

8

15

8ba

15ba

- 3            U+E517    ventiduesima
- 3          U+E517    ventiduesima
- U+EC93    octaveBaselineB
- U+EC91    octaveBaselineA
- 4            U+F6F8    vintinovesima
- 4          U+F6F8    vintinovesima
- U+EC93    octaveBaselineB
- U+EC91    octaveBaselineA

22

29

22ba

29ba

Note: According to the implementation notes of SMuFL Octaves, the suffixes *vb* and *mb* as used in ekm-ottavation-simple-ordinals and ekm-ottavation-ordinals-b are corruptions of the more correct forms *va bassa* and *ma bassa* as used in ekm-ottavation-ordinals-bassa .

The recommended abbreviation for *8va bassa* is *8ba* as used in ekm-ottavation-ordinals-ba and ekm-ottavation-numbers-ba .

\ekm-ottavation DEFINITION

Draw an ottavation text as markup. **DEFINITION** may consist of the following keys.

8	<b>8</b>	U+E510	ottava
8^va	<b>8<sup>va</sup></b>	U+E511	ottavaAlta
8va	<b>8<sub>va</sub></b>	U+E512	ottavaBassa
8ba	<b>8<sub>ba</sub></b>	U+E513	ottavaBassaBa
8vb	<b>8<sub>vb</sub></b>	U+E51C	ottavaBassaVb
8^vb	<b>8<sup>vb</sup></b>	U+F652	ottavaBassaSupVb
15	<b>15</b>	U+E514	quindicesima
15^ma	<b>15<sup>ma</sup></b>	U+E515	quindicesimaAlta
15ma	<b>15<sub>ma</sub></b>	U+E516	quindicesimaBassa
15mb	<b>15<sub>mb</sub></b>	U+E51D	quindicesimaBassaMb
15^mb	<b>15<sup>mb</sup></b>	U+F653	quindicesimaBassaSupMb
22	<b>22</b>	U+E517	ventiduesima
22^ma	<b>22<sup>ma</sup></b>	U+E518	ventiduesimaAlta
22ma	<b>22<sub>ma</sub></b>	U+E519	ventiduesimaBassa
22mb	<b>22<sub>mb</sub></b>	U+E51E	ventiduesimaBassaMb
22^mb	<b>22<sup>mb</sup></b>	U+F654	ventiduesimaBassaSupMb
29	<b>29</b>	U+F6F8	ventinovesima
29^ma	<b>29<sup>ma</sup></b>	U+F6F9	ventinovesimaAlta
29ma	<b>29<sub>ma</sub></b>	U+F6FA	ventinovesimaBassa
29mb	<b>29<sub>mb</sub></b>	U+F6FB	ventinovesimaBassaMb
29^mb	<b>29<sup>mb</sup></b>	U+F655	ventinovesimaBassaSupMb
(	<b>(</b>	U+E51A	octaveParensLeft
)	<b>)</b>	U+E51B	octaveParensRight
bassa	<b>bassa</b>	U+E51F	octaveBassa
loco	<b>loco</b>	U+EC90	octaveLoco
^a	<b>a</b>	U+EC92	octaveSuperscriptA
^b	<b>b</b>	U+EC94	octaveSuperscriptB
^m	<b>m</b>	U+EC96	octaveSuperscriptM
^v	<b>v</b>	U+EC98	octaveSuperscriptV
a	<b>a</b>	U+EC91	octaveBaselineA
b	<b>b</b>	U+EC93	octaveBaselineB
m	<b>m</b>	U+EC95	octaveBaselineM
v	<b>v</b>	U+EC97	octaveBaselineV

## Tuplet numbers

```
\ekmSmuflOn #'tuplet
```

Draw SMuFL tuplet numbers as numerator only. Set the first formatting function listed below, so this switch is not required if one of these functions is set explicitly.

0	<b>0</b>	U+E880	tuplet0
	:		
9	<b>9</b>	U+E889	tuplet9
:	:	U+E88A	tupletColon

```
ekm-tuplet-number::calc-denominator-text
ekm-tuplet-number::calc-fraction-text
(ekm-tuplet-number::non-default-tuplet-denominator-text NUM)
(ekm-tuplet-number::non-default-tuplet-fraction-text NUM DENOM)
(ekm-tuplet-number::append-note-wrapper
  FUNCTION DURATION)
(ekm-tuplet-number::fraction-with-notes
  NUM-DURATION DENOM-DURATION)
(ekm-tuplet-number::non-default-fraction-with-notes
  NUM NUM-DURATION DENOM DENOM-DURATION)
```

Tuplet formatting functions. The last three draw `metronome` style notes for the specified durations.

```
(ekm-tuplet-number NUM DENOM)
```

Draw NUM:DENOM, or NUM only if DENOM is 0. Use the actual tuplet fraction for NUM or DENOM if `#f` is specified. It is called by the first four functions above, i.e. they are equivalent to:

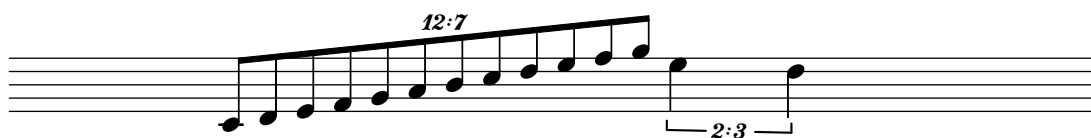
```
(ekm-tuplet-number #f 0)
(ekm-tuplet-number #f #f)
(ekm-tuplet-number NUM 0)
(ekm-tuplet-number NUM DENOM)
```

## Examples:

```
ekm-tuplet-number::calc-denominator-text
```



```
ekm-tuplet-number::calc-fraction-text
```





```
(ekm-tuplet-number::append-note-wrapper
  ekm-tuplet-number::calc-fraction-text
  (ly:make-duration 2 0))
```



```
(ekm-tuplet-number::fraction-with-notes
  (ly:make-duration 2 1)
  (ly:make-duration 3 0))
```



```
(ekm-tuplet-number::non-default-fraction-with-notes
  12 (ly:make-duration 3 0)
  4 (ly:make-duration 2 0))
```



## Fingering instructions

---

`\ekmSmuflOn #'fingering`

Draw SMuFL fingering instructions specified with a digit or with `\finger`, as well as right-hand fingerings specified with `\rightHandFinger`, using `\ekm-finger`.

`\ekm-finger DEFINITION`

Draw a fingering instruction as markup with a 5 steps larger font size intended for use in `\finger`.

**DEFINITION** may consist of the following keys. If the first character is *\** the italic versions of digits, parentheses and brackets are drawn.

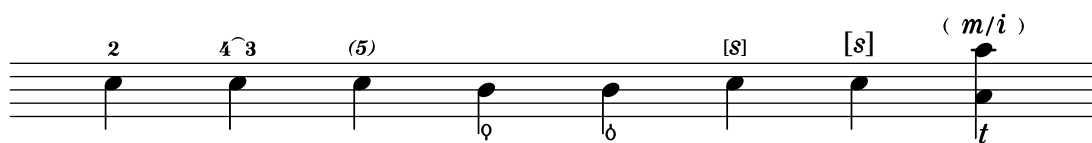
0	<b>0</b>	U+ED10	fingering0
	:		
5	<b>5</b>	U+ED15	fingering5
6	<b>6</b>	U+ED24	fingering6
	:		
9	<b>9</b>	U+ED27	fingering9
*...0	<i>0</i>	U+ED80	fingering0Italic
	:		
*...9	<i>9</i>	U+ED89	fingering9Italic
th	<b>ø</b>	U+E624	stringsThumbPosition
ht	<b>ó</b>	U+E625	stringsThumbPositionTurned
T	<b>T</b>	U+ED16	fingeringTUpper
t	<b>t</b>	U+ED18	fingeringTLower
p	<b>p</b>	U+ED17	fingeringPLower
i	<b>i</b>	U+ED19	fingeringILower
m	<b>m</b>	U+ED1A	fingeringMLower
a	<b>a</b>	U+ED1B	fingeringALower
c	<b>c</b>	U+ED1C	fingeringCLower
x	<b>x</b>	U+ED1D	fingeringXLower
e	<b>e</b>	U+ED1E	fingeringELower
o	<b>o</b>	U+ED1F	fingeringOLower
q	<b>q</b>	U+ED8E	fingeringQLower
s	<b>s</b>	U+ED8F	fingeringSLower
(	<b>(</b>	U+ED28	fingeringLeftParenthesis
)	<b>)</b>	U+ED29	fingeringRightParenthesis
[	<b>[</b>	U+ED2A	fingeringLeftBracket
]	<b>]</b>	U+ED2B	fingeringRightBracket
*... (	<i>(</i>	U+ED8A	fingeringLeftParenthesisItalic
*... )	<i>)</i>	U+ED8B	fingeringRightParenthesisItalic
*... [	<i>[</i>	U+ED8C	fingeringLeftBracketItalic
*... ]	<i>]</i>	U+ED8D	fingeringRightBracketItalic

.	•	U+ED2C	fingeringSeparatorMiddleDot
,	◦	U+ED2D	fingeringSeparatorMiddleDotWhite
/	/	U+ED2E	fingeringSeparatorSlash
~~	ˆ	U+ED20	fingeringSubstitutionAbove
~	˘	U+ED21	fingeringSubstitutionBelow
-	-	U+ED22	fingeringSubstitutionDash
M	[	U+ED23	fingeringMultipleNotes

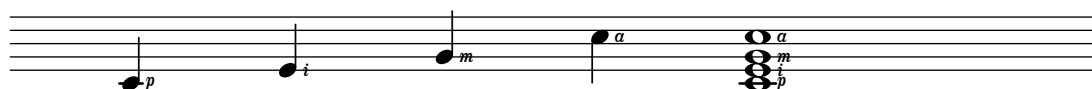
Note: The `\thumb` command always produces normal LilyPond output. Use `\finger "th"` to draw the corresponding SMuFL glyph.

## Examples:

```
-2
-\finger ...
  "4~~3"
  "*" (5) "
  "th"
  "ht"
  "[s]"
  \markup \concat { "[" \ekm-finger #"s" "]" }
  "t"
  "( m/_i )"
```



```
\rightHandFinger #1 ... #4
```



## String number indications

`\ekmSmuflOn #'stringnumber`

Draw SMuFL string number indications specified with `\NUMBER`, using `\ekm-string-number`.

**Note:** `\romanStringNumbers` overrides the SMuFL switch so that reverting with `\arabicStringNumbers` produces normal LilyPond output.

`\ekm-string-number ARG`

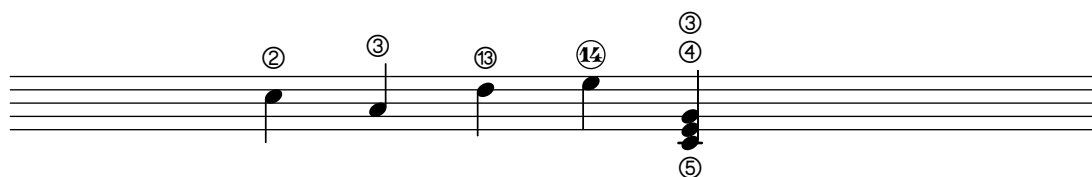
Draw a string number indication as markup. ARG is a number or string. For a number or a string representing a number, the respective SMuFL symbol is drawn if the number is in the range 0 thru 13, else the number itself is drawn with a circle around. Any other string, e.g. a Roman numeral, is drawn in italic style.

0	①	U+E833	guitarString0
	:		
9	⑨	U+E83C	guitarString9
10	⑩	U+E84A	guitarString10
	:		
13	⑬	U+E84D	guitarString13

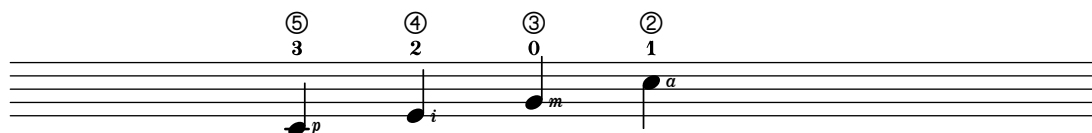
## Examples:

The second staff combines SMuFL left-hand fingerings, string number indications, and right-hand fingerings, which requires `\ekmSmuflOn #'(fingering stringnumber)`.

```
c\2
a\3
d\13
e\14
< c,\5 e\4 g\3 >
```



```
< c -3 \5 \rightHandFinger #1 >
< e -2 \4 \rightHandFinger #2 >
< g -0 \3 \rightHandFinger #3 >
< c -1 \2 \rightHandFinger #4 >
```































## Piano pedals

\ekmSmuflOn #'pedal

Draw SMuFL piano pedals for sustain, sostenuto, and una corda, using \ekm-piano-pedal.

\ekm-piano-pedal DEFINITION

Draw piano pedal symbols as markup. **DEFINITION** may consist of the following keys.

Ped.		U+E650	keyboardPedalPed
Ped		U+F434	keyboardPedalPedNoDot
P		U+E651	keyboardPedalP
e		U+E652	keyboardPedalE
d		U+E653	keyboardPedalD
Sost.		U+E659	keyboardPedalSost
Sost		U+F435	keyboardPedalSostNoDot
Sos.		U+F6D1	keyboardPedalSos2
sos.		U+F6D0	keyboardPedalSos
S		U+E65A	keyboardPedalS
unacorda	<i>una corda</i>	U+F6CC	keyboardPedalUnaCorda
trecorde	<i>tre corde</i>	U+F6CD	keyboardPedalTreCorde
u.c.	<i>u.c.</i>	U+F6CE	keyboardPedalUC
t.c.	<i>t.c.</i>	U+F6CF	keyboardPedalTC
.		U+E654	keyboardPedalDot
-		U+E658	keyboardPedalHyphen
*		U+E655	keyboardPedalUp
o		U+E65D	keyboardPedalUpSpecial
,		U+E65B	keyboardPedalHalf2
'		U+E65C	keyboardPedalHalf3
H		U+E656	keyboardPedalHalf
^		U+E657	keyboardPedalUpNotch
1/2Ped		U+F6B0	keyboardPedalHalf4
1/4		U+F6BA	keyboardPedalPosQuarter
1/2		U+F6BB	keyboardPedalPosHalf
3/4		U+F6BC	keyboardPedalPosThreeQuarters
1		U+F6BD	keyboardPedalPosFull
l		U+E65E	keyboardLeftPedalPictogram
m		U+E65F	keyboardMiddlePedalPictogram
r		U+E660	keyboardRightPedalPictogram
(		U+E676	keyboardPedalParensLeft
)		U+E677	keyboardPedalParensRight

## Examples:

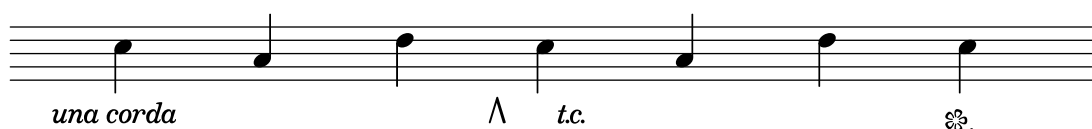
```
\set Staff.pedalSustainStrings =
  #'("Ped" ", |1/4" "*")
```



```
\set Staff.pedalSostenutoStyle = #'text
\set Staff.pedalSostenutoStrings =
  #'("Sost-P" "(" "S__*")
```



```
\set Staff.pedalUnaCordaStyle = #'text
\set Staff.pedalUnaCordaStrings =
  #'("unacorda" "^____t.c." "o_.")
```



# Harp pedals

\ekm-harp-pedal DEFINITION

Draw a harp pedal diagram as markup, similar to \harp-pedal but composed of the following glyphs. Note that the glyphs for pedal changes (o^ o- ov) are [Ekmelos](#) specific. Space characters are allowed between the keys.

^	⏏	U+E680	harpPedalRaised
o^	⏏◌	U+F648	harpPedalRaisedChange
-	⏏	U+E681	harpPedalCentered
o-	⏏◌	U+F649	harpPedalCenteredChange
v	⏏	U+E682	harpPedalLowered
ov	⏏◌	U+F64A	harpPedalLoweredChange
	⏏	U+E683	harpPedalDivider






## Examples:

#" ^ v - | v v - ^"  
#" ^ o- - | v v - ^"

## Fret diagrams

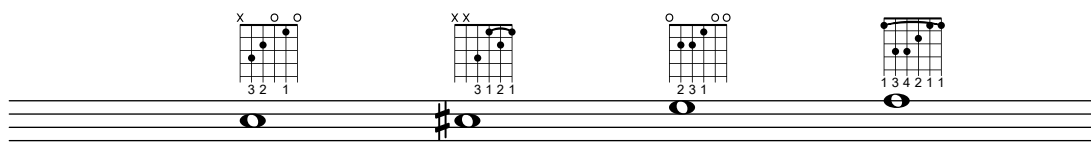
\ekm-fret-diagram-terse DEFINITION

Draw a fret diagram as markup, similar to \fret-diagram-terse but composed of the following glyphs and simplified, i.e. the properties `fret-diagram-details`, `thickness`, `size`, and `align-dir` are ignored. Fingering is always placed below.

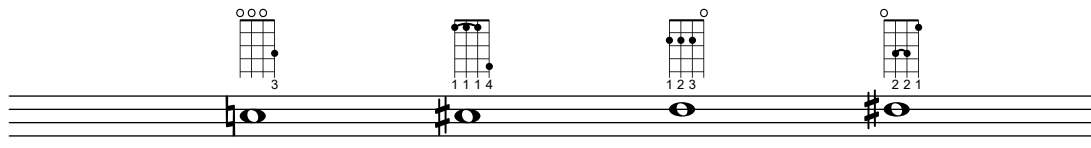
3		U+E851	fretboard3StringNut
4		U+E853	fretboard4StringNut
5		U+E855	fretboard5StringNut
6		U+E857	fretboard6StringNut
.		U+E858	fretboardFilledCircle
x		U+E859	fretboardX
o		U+E85A	fretboardO

## Examples:

```
#"x;3-3;2-2;o;1-1;o;"
#"x;x;3-3;1-1-(;2-2;1-1-);"
#"o;2-2;2-3;1-1;o;o;"
#"1-1-(;3-3;3-4;2-2;1-1;1-1-);"
```



```
#"o;o;o;3-3;"
#"1-1-(;1-1;1-1-);4-4;"
#"2-1;2-2;2-3;o;"
#"o;3-2-(;3-2-);1-1;"
```





## Accordion registers

`\ekm-accordion NAME`

Draw an accordion register symbol as markup, similar to the commands in `(scm accreg)`. The name can include a prefix for the register type, separated by a space:

<code>d</code>	Discant (default)
<code>sb sb4 sb5 sb6</code>	Standard bass, four / five / six reed
<code>fb</code>	Free bass
<code>sq</code>	Square
























Most of the symbols use precomposed glyphs. The others are composed using `accdnCombRH3RanksEmpty` (U+E8C6) et al.

`\ekmAccordion NAME`

Set an accordion register symbol as a standalone music event.

This is equivalent to `<> ^ \markup \ekm-accordion NAME`

"d 1"		U+E8A4	accdnRH3RanksBassoon
"d 10"		U+E8A1	accdnRH3RanksClarinet
"d 11"		U+E8AB	accdnRH3RanksBandoneon
"d 1+0"		U+E8A2	accdnRH3RanksUpperTremolo8
"d 1+1"			
"d 1-0"		U+E8A3	accdnRH3RanksLowerTremolo8
"d 1-1"			
"d 20"		U+E8AE	accdnRH3RanksTwoChoirs
"d 21"		U+E8AF	accdnRH3RanksTremoloLower8ve
"d 2+0"		U+E8A6	accdnRH3RanksViolin
"d 2+1"		U+E8AC	accdnRH3RanksAccordion
"d 2-0"			
"d 2-1"			
"d 30"		U+E8A8	accdnRH3RanksAuthenticMusette
"d 31"		U+E8B1	accdnRH3RanksDoubleTremoloLower8ve
"d 100"		U+E8A0	accdnRH3RanksPiccolo
"d 101"		U+E8A9	accdnRH3RanksOrgan
"d 110"		U+E8A5	accdnRH3RanksOboe
"d 111"		U+E8AA	accdnRH3RanksHarmonium
"d 11+0"			
"d 11+1"			
"d 11-0"			
"d 11-1"			

"d 120"		U+E8B0	accdnRH3RanksTremoloUpper8ve
"d 121"		U+E8AD	accdnRH3RanksMaster
"d 12+0"		U+E8A7	accdnRH3RanksImitationMusette
"d 12+1"			
"d 12-0"			
"d 12-1"			
"d 130"		U+E8B2	accdnRH3RanksDoubleTremoloUpper8ve
"d 131"		U+E8B3	accdnRH3RanksFullFactory
"sb Soprano"		U+E8B4	accdnRH4RanksSoprano
"sb Alto"		U+E8B5	accdnRH4RanksAlto
"sb Tenor"		U+E8B6	accdnRH4RanksTenor
"sb Master"		U+E8B7	accdnRH4RanksMaster
"sb Soft Bass"		U+E8B8	accdnRH4RanksSoftBass
"sb Soft Tenor"		U+E8B9	accdnRH4RanksSoftTenor
"sb Bass/Alto"		U+E8BA	accdnRH4RanksBassAlto
"sb4 Soprano"		U+E8B4	accdnRH4RanksSoprano
"sb4 Alto"		U+E8B5	accdnRH4RanksAlto
"sb4 Tenor"			
"sb4 Master"			
"sb4 Soft Bass"			
"sb4 Bass/Alto"		U+E8BA	accdnRH4RanksBassAlto
"sb4 Soft Bass/Alto"			
"sb4 Soft Tenor"		U+E8B9	accdnRH4RanksSoftTenor

"sb5 Bass/Alto"		U+E8BA	accdnRH4RanksBassAlto
"sb5 Soft Bass/Alto"			
"sb5 Alto"			
"sb5 Tenor"			
"sb5 Master"			
"sb5 Soft Bass"			
"sb5 Soft Tenor"		U+E8B9	accdnRH4RanksSoftTenor
"sb5 Soprano"		U+E8B4	accdnRH4RanksSoprano
"sb5 Sopranos"			
"sb5 Solo Bass"			
"sb6 Soprano"		U+E8B4	accdnRH4RanksSoprano
"sb6 Alto"			
"sb6 Soft Tenor"		U+E8B9	accdnRH4RanksSoftTenor
"sb6 Master"		U+E8B7	accdnRH4RanksMaster
"sb6 Alto/Soprano"			
"sb6 Bass/Alto"		U+E8BA	accdnRH4RanksBassAlto
"sb6 Soft Bass"		U+E8B8	accdnRH4RanksSoftBass
"fb 10"		U+E8BB	accdnLH2Ranks8Round
"fb 1"		U+E8BC	accdnLH2Ranks16Round
"fb 11"		U+E8BD	accdnLH2Ranks8Plus16Round
"fb Master"		U+E8BE	accdnLH2RanksMasterRound
"fb Master 1"		U+E8BF	accdnLH2RanksMasterPlus16Round
"fb Master 11"		U+E8C0	accdnLH2RanksFullMasterRound
"sq 1"		U+E8C1	accdnLH3Ranks8Square
"sq 100"		U+E8C2	accdnLH3Ranks2Square
"sq 2"		U+E8C3	accdnLH3RanksDouble8Square
"sq 101"		U+E8C4	accdnLH3Ranks2Plus8Square
"sq 102"		U+E8C5	accdnLH3RanksTuttiSquare

Accordion ricochet

\ekmRicochet NUMBER

Draw a ricochet symbol as an expressive mark (script). NUMBER is an integer in the range 2 thru 6. [ Ly ]

#2  
#3  
#4  
#5  
#6

U+E8CD  
U+E8CE  
U+E8CF  
U+E8D0  
U+E8D1

accdnRicochet2  
accdnRicochet3  
accdnRicochet4  
accdnRicochet5  
accdnRicochet6

\ekmStemRicochet NUMBER MUSIC

Draw a ricochet symbol vertically centered on the stems in MUSIC. NUMBER is an integer in the range 2 thru 6.

#2  
#3  
#4  
#5  
#6

U+E8D2  
U+E8D3  
U+E8D4  
U+E8D5  
U+E8D6

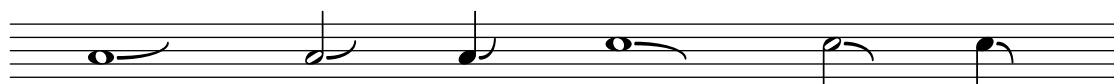
accdnRicochetStem2  
accdnRicochetStem3  
accdnRicochetStem4  
accdnRicochetStem5  
accdnRicochetStem6

## Falls and doits

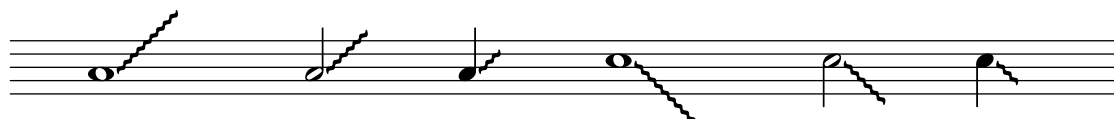
\ekmBendAfter STYLE DIRECTION

Draw a fall or doit (lift) symbol after a note. The style can be one of the following symbols. Note that only the sign of the direction is respected, contrary to \bendAfter.

#'bend #UP	U+E5D6	brassDoitLong
	U+E5D5	brassDoitMedium
	U+E5D4	brassDoitShort
#'bend #DOWN	U+E5D9	brassFallLipLong
	U+E5D8	brassFallLipMedium
	U+E5D7	brassFallLipShort



#'rough #UP	U+E5D3	brassLiftLong
	U+E5D2	brassLiftMedium
	U+E5D1	brassLiftShort
#'rough #DOWN	U+E5DF	brassFallRoughLong
	U+E5DE	brassFallRoughMedium
	U+E5DD	brassFallRoughShort



#'smooth #UP	U+E5EE	brassLiftSmoothLong
	U+E5ED	brassLiftSmoothMedium
	U+E5EC	brassLiftSmoothShort
#'smooth #DOWN	U+E5DC	brassFallSmoothLong
	U+E5DB	brassFallSmoothMedium
	U+E5DA	brassFallSmoothShort



## Figured bass

---

\ekmSmuflOn #'fbass

Draw SMuFL bass figures with \figuremode. Some raised / diminished figures use precomposed glyphs which ignore the property figuredBassPlusDirection.

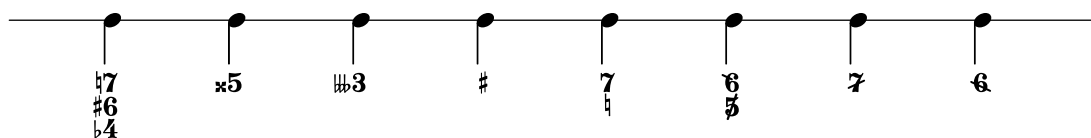
0	<b>0</b>	U+EA50	figbass0
1	<b>1</b>	U+EA51	figbass1
2	<b>2</b>	U+EA52	figbass2
3	<b>3</b>	U+EA54	figbass3
4	<b>4</b>	U+EA55	figbass4
5	<b>5</b>	U+EA57	figbass5
6	<b>6</b>	U+EA5B	figbass6
7	<b>7</b>	U+EA5D	figbass7
8	<b>8</b>	U+EA60	figbass8
9	<b>9</b>	U+EA61	figbass9
!	<b>b</b>	U+EA65	figbassNatural
-	<b>b</b>	U+EA64	figbassFlat
+	<b>#</b>	U+EA66	figbassSharp
--	<b>bb</b>	U+EA63	figbassDoubleFlat
++	<b>##</b>	U+EA67	figbassDoubleSharp
---	<b>bbb</b>	U+ECC1	figbassTripleFlat
+++	<b>###</b>	U+ECC2	figbassTripleSharp
\+	<b>+</b>	U+EA6C	figbassPlus
/	<b>/</b>	U+EA6D	figbassCombiningRaising
\\	<b>\</b>	U+EA6E	figbassCombiningLowering
2\+	<b>2+</b>	U+EA53	figbass2Raised
4\+	<b>4+</b>	U+EA56	figbass4Raised
5\+	<b>5+</b>	U+EA58	figbass5Raised1
5\\	<b>5</b>	U+EA59	figbass5Raised2
5/	<b>5</b>	U+EA5A	figbass5Raised3
6\\	<b>6</b>	U+EA5C	figbass6Raised
6\+	<b>6</b>	U+EA6F	figbass6Raised2
7\+	<b>7</b>	U+EA5E	figbass7Raised1
7\\	<b>7</b>	U+EA5F	figbass7Raised2
7/	<b>7</b>	U+ECC0	figbass7Diminished
9\\	<b>9</b>	U+EA62	figbass9Raised

## Examples:

```

< 7! 6+ 4- >
< 5++ >
< 3--- >
< _+ >
< 7_! >
< 6\+ 5/ >
< 7/ >
< 6\\ >

```



```

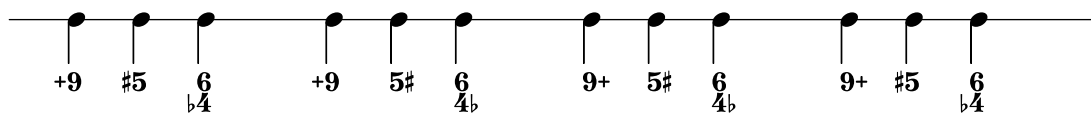
< 9\+ >
< 5+ >
< 6 4- >

```

```
with
```

```
\set figuredBassAlterationDirection = #LEFT | #RIGHT
```

```
\set figuredBassPlusDirection = #LEFT | #RIGHT
```



## Lyrics

---

`\ekmSmuflOn #'lyric`

Draw the words in a lyric input mode (`\lyricmode` etc.) with `\ekm-tied-lyric`.

Note that the characters `_ %` must be quoted in order to be passed on to this command.

`\ekm-tied-lyric STRING`

Draw the string as markup, replacing the characters `~ _ %` with the glyphs specified below. The space between the adjoining words depends on the width of the respective glyph, while the property `word-space` is ignored. The narrow elision for single characters works with all Unicode characters, contrary to `\tied-lyric`.

~	⸘	U+E551	lyricsElision
~x~	⸘	U+E550	lyricsElisionNarrow
~~	⸘	U+E552	lyricsElisionWide
_	—	U+E553	lyricsHyphenBaseline
%	⸘	U+E555	lyricsTextRepeat

### Examples:

"Siam navi~all'onde~~algenti"      Siam navi⸘all'onde⸘algenti

"Tutta la\_vita~ē~un mar %"      Tutta la\_vita⸘ē un mar ⸘

{ Che~~in ques -- ta~è~in quel -- l'al -- "tr\_on" -- "da %" }





# Analytics symbols

---

\ekm-analytics DEFINITION

Draw analytics symbols as markup. DEFINITION may consist of the following keys.

H	ℋ	U+E860	analyticsHauptstimme
CH	℣ℋ	U+E86A	analyticsChoralmelodie
RH	℞ℋ	U+E86B	analyticsHauptrhythmus
N	℟	U+E861	analyticsNebenstimme
[	ℭ	U+E862	analyticsStartStimme
]	ℭ	U+E863	analyticsEndStimme
Th	℥℥	U+E864	analyticsTheme
hT	℥℥	U+E865	analyticsThemeRetrograde
ihT	℥℥	U+E866	analyticsThemeRetrogradeInversion
iTh	℥℥	U+E867	analyticsThemeInversion
T	℥	U+E868	analyticsTheme1
iT	℥	U+E869	analyticsInversion1

Function theory symbols

`\ekm-func` DEFINITION

Draw a function theory symbol as markup. **DEFINITION** is a string of the form:

Paren Function , Bass , Soprano ^ Extra ... Paren

All parts are optional and may consist of the keys specified further below.

The bass / soprano symbol is placed below / above the function symbol.

The extra symbols are stacked vertically and raised to the right of the function symbol.

A leading / trailing parenthesis ( ) [ ] { } is placed separately before / after the entire symbol.

Used properties:

- `font-size` (0) for the function symbol.
- `func-size` (-4) relative to the font size for bass, soprano, and extra symbols.
- `func-skip` (2.5) for vertical distances.
- `func-space` (0.3) for horizontal space around the function symbol.

`\ekmFunc` DEFINITION











Set a function theory symbol as a music expression, for use in a `Lyrics` context. The symbol is drawn with a 4 steps smaller font size compared to `\ekm-func` . **DEFINITION** is a string as described above, with a further optional suffix:

- Starts an extender line after the symbol.
- . Stops an extender line at the symbol.
- + Inserts the symbol between notes with `\set stanza .`
- \* Dito but with the 4 steps larger font size of `\ekm-func` .

Note that the `Lyrics` context requires the `Text_spanner_engraver` to draw extender lines.

`\ekmFuncList` DEFINITION-LIST

Set a sequence of function theory symbols as music expressions, for use in a `Lyrics` context. **DEFINITION-LIST** is a list of strings as for `\ekmFunc` .

T		U+EA8B	functionTUpper
Tg			
Tp			
t		U+EA8C	functionTLower
D		U+EA7F	functionDUpper
/D		U+F644	functionSlashedD
Dp			
DD		U+EA81	functionDD
/DD		U+EA82	functionSlashedDD
d		U+EA80	functionDLower

S	Š	U+EA89	functionSUpper
Sg	Šg		
Sp	Šp		
SS	Šš	U+EA7D	functionSSUpper
s	š	U+EA8A	functionSLower
ss	šš	U+EA7E	functionSSLower
F	F	U+EA99	functionFUpper
G	G	U+EA83	functionGUpper
g	g	U+EA84	functionGLower
I	I	U+EA9A	functionIUpper
i	i	U+EA9B	functionILower
K	K	U+EA9C	functionKUpper
k	k	U+EA9D	functionKLower
L	L	U+EA9E	functionLUpper
l	l	U+EA9F	functionLLower
M	M	U+ED00	functionMUpper
m	m	U+ED01	functionMLower
N	N	U+EA85	functionNUpper
n	n	U+EA86	functionNLower
P	P	U+EA87	functionPUpper
p	p	U+EA88	functionPLower
r	r	U+ED03	functionRLower
V	V	U+EA8D	functionVUpper
v	v	U+EA8E	functionVLower
0	0	U+EA70	functionZero
:	:		
9	9	U+EA79	functionNine
<	<	U+EA7A	functionLessThan
>	>	U+EA7C	functionGreaterThan
-	-	U+EA7B	functionMinus
+	+	U+EA98	functionPlus
o	o	U+EA97	functionRing

(	(	U+EA91	functionParensLeft
)	)	U+EA92	functionParensRight
[	[	U+EA8F	functionBracketLeft
]	]	U+EA90	functionBracketRight
{	<	U+EA93	functionAngleLeft
}	>	U+EA94	functionAngleRight
..	..	U+EA95	functionRepetition1
..+	..+	U+EA96	functionRepetition2
b	b	U+ED60	csymAccidentalFlat
#	#	U+ED62	csymAccidentalSharp
bb	bb	U+ED64	csymAccidentalDoubleFlat
x	x	U+ED63	csymAccidentalDoubleSharp
=	=	U+ED61	csymAccidentalNatural
~			

The keys `b` `#` `bb` `x` `=` draw standard accidentals for chord symbols.

The key `~` draws a space with the dimensions of functionZero (U+EA70) . This is especially useful for empty extra symbols.

## Example 1:

Uses `\ekm-func` in text scripts to attach function theory symbols to chords and spacer rest.

Sets `\textLengthOn` and `TextScript.staff-padding` for a consistent vertical alignment.

```
\relative c' {
  \textLengthOn
  \override TextScript.staff-padding = #6
  <c e g bes>2_\markup \ekm-func "D^7 "
  <e g bes! c>_\markup \ekm-func "(D,3^7)"

  \override TextScript.staff-padding = #11
  <c e g c>4_\markup \ekm-func "T____"
  <g e' g c>_\markup \ekm-func "D^4^6"
  s_\markup \ekm-func "^-^-"
  <g d' g b>_\markup \ekm-func "^3^5"

  \key es \major
  \override TextScript.staff-padding = #7
  <g' b d>1_\markup \ekm-func "V#"
  <f as c e>_\markup \ekm-func "IV^7#"
  <ces es as!>_\markup \ekm-func "VI,b"
}
```

The musical notation shows a sequence of chords and a rest on a treble clef staff in common time. The chords are: D<sup>7</sup> (D<sup>7</sup>), T, D<sup>4-3</sup>, V<sup>#</sup>, IV<sup>7#</sup>, and VI<sub>b</sub>. The function theory symbols are: D<sup>7</sup>, (D<sup>7</sup>), T, D<sup>4-3</sup>, V<sup>#</sup>, IV<sup>7#</sup>, and VI<sub>b</sub>. The symbols are aligned with the chords and the rest.

## Example 2:

Uses `\ekmFuncList` in a Lyrics context to synchronise function theory symbols to music and to ensure a consistent vertical alignment. The Lyrics context requires the `Text_spanner_engraver` and is aligned to a `NullVoice` context.

The sample is taken from [lsr.di.unimi.it/LSR/Item?id=967](http://lsr.di.unimi.it/LSR/Item?id=967) by Klaus Blum and adapted for Esmuflily.

```
funcSoprano = \relative c'' {
  e4 e e( d)
  c4 d d2
  d4 e8 d c4 c
  d8( c) <b g>4 c2
}

funcAltTenor = \relative c'' {
  <c g>4 <bes g> <a f>2
  <a d,>4 <c a> <c a>( <b g>)
  <b e,>2 <g e>4 <a f>
  <a d,>4 d,8( f) <g e>2
}

funcBass = \relative c {
  \clef bass
  c4 cis d2
  f4 fis g2
  gis2 bes4 a8 g
  fis4 g c,2
}

funcAligner = \relative c {
  c4 cis d d
  f4 fis g g
  gis4 gis8 gis bes4 a8 g
  fis8 fis g g c,2
}

funcSymbols = \lyricmode {
  \set stanza = #"C major:"
  \ekmFuncList #'(
    "T,,3" " (*" "/D,3^7^9>" ")*" "Sp^9-" "^8."
    "S^5^6" "(D,3^7)" "D^2^4-" "^1^3."
    "(D,3^7-" "^8" "^7." "_" [Tp] +" "(D,7)" "S,3-" " ,2."
    "DD,3^8-" "^7." "D^5-" "^7." "T"
  )
}

\layout {
  \context {
    \Lyrics
    \consists "Text_spanner_engraver"
    \override StanzaNumber.font-family = #'sans
    \override StanzaNumber.font-series = #'medium
  }
}
```

```

\new GrandStaff
<<
  \new Staff
    \new Voice \partCombine \funcSoprano \funcAltTenor

  \new Staff
  <<
    \new Voice \funcBass
    \new NullVoice = "funcaligner" \funcAligner
    \new Lyrics \lyricsto "funcaligner" \funcSymbols
  >>
>>

```

C major:  $\overset{3}{T} \left( \overset{9}{\underset{3}{D^7}} \right) \overset{9}{S_p} \overset{8}{\text{---}}$   $\overset{6}{S^5} \left( \overset{4}{\underset{3}{D^7}} \right) \overset{3}{D^2} \overset{1}{\text{---}}$   $\left( \overset{7}{\underset{3}{D^7}} \overset{8}{\text{---}} \overset{7}{\text{---}} \right) [T_p] \left( \underset{7}{D} \right) \overset{3}{S} \overset{2}{\text{---}}$   $\overset{8}{\underset{3}{D^7}} \overset{7}{\text{---}} \overset{5}{D^5} \overset{7}{\text{---}} T$

## Arrows and arrow heads

---

\ekm-arrow STYLE ORIENTATION

Draw an arrow, an arrow head, or a geometric shape according to **ORIENTATION** as markup. The style can be one of the following symbols. For most of the styles, diagonal and / or “bilateral” orientations are not supported, e.g. there are no both-sided arrow heads.

black	↑	U+EB60	arrowBlackUp
white	↑	U+EB68	arrowWhiteUp
open	↑	U+EB70	arrowOpenUp
simple	↑	U+2191	
double	⇑	U+21D1	
triple	⇓	U+290A	
quadruple	⇓	U+27F0	
black-wide	⇑	U+2B06	
white-wide	⇑	U+21E7	
triangle	↑	U+2B61	
triangle-bar	↑	U+2B71	
two-headed	↕	U+2BED	
dashed	⋈	U+21E1	
triangle-dashed	⋈	U+2B6B	
opposite	↕	U+21C5	
triangle-opposite	↕	U+2B81	
paired	⇑	U+21C8	
triangle-paired	⇑	U+2B85	
bent-tip	↗	U+21B1	
long-bent-tip	↗	U+2BA3	
curving	↗	U+2934	
black-head	▲	U+EB78	arrowheadBlackUp
white-head	△	U+EB80	arrowheadWhiteUp
open-head	^	U+EB88	arrowheadOpenUp
equilateral-head	▲	U+2B9D	
three-d-head	▲	U+2B99	
black-triangle	▲	U+25B2	
white-triangle	△	U+25B3	
black-small-triangle	▲	U+25B4	
white-small-triangle	▲	U+25B5	
half-circle	◐	U+2BCA	
circle-half-black	◑	U+25D3	
square-half-black	◼	U+2B12	
diamond-half-black	◔	U+2B18	
circle-quarters	◕	U+25D4	



Examples:

ORIENTATION	#N	#NE	#E	#SE	#S	#SW	#W	#NW	#NS	#NESW	#EW	#SENW
black	↑	↗	→	↘	↓	↙	←	↖	↕		↔	
simple	↑	↗	→	↘	↓	↙	←	↖	↕	↗↖	↔	↘↙
bent-tip	↗	↗	↗	↘	↘	↙	↙	↖				
open-head	^	7	>	v	v	^	<	7				
black-triangle	▲	▴	▶	▹	▼	▾	◀	▸				
circle-quarters	●		◐		◑		◒					

`\ekm-arrow-head` AXIS DIRECTION FILLED  
Draw an arrow head as markup, i.e. `black-head` if `FILLED` is a true value, else `open-head`.




























## Percussion symbols














---

\ekm-beater STYLE ORIENTATION

























Draw a percussion beater according to **ORIENTATION** as markup. The style can be one of the following symbols. The suffix separated by – is optional. If it is not specified or unknown the first matching style in the list is drawn. Styles in the list without –... actually have the suffix `-normal` which need not be specified.

Most of the beaters have predefined glyphs for the orientations `N S NE NW`, the others only for `N S` or `N`. The remaining orientations are achieved by flipping or by rotating through 90 or 30 degrees.

xyl-soft		U+E770	pictBeaterSoftXylophoneUp
xyl-medium		U+E774	pictBeaterMediumXylophoneUp
xyl-hard		U+E778	pictBeaterHardXylophoneUp
xyl-wood		U+E77C	pictBeaterWoodXylophoneUp
glsp-soft		U+E780	pictBeaterSoftGlockenspielUp
glsp-hard		U+E784	pictBeaterHardGlockenspielUp
timpani-soft		U+E788	pictBeaterSoftTimpaniUp
timpani-medium		U+E78C	pictBeaterMediumTimpaniUp
timpani-hard		U+E790	pictBeaterHardTimpaniUp
timpani-wood		U+E794	pictBeaterWoodTimpaniUp
yarn-soft		U+E7A2	pictBeaterSoftYarnUp
yarn-medium		U+E7A6	pictBeaterMediumYarnUp
yarn-hard		U+E7AA	pictBeaterHardYarnUp
gum-soft		U+E7BB	pictGumSoftUp
gum-medium		U+E7BF	pictGumMediumUp
gum-hard		U+E7C3	pictGumHardUp
bass-soft		U+E798	pictBeaterSoftBassDrumUp
bass-medium		U+E79A	pictBeaterMediumBassDrumUp
bass-hard		U+E79C	pictBeaterHardBassDrumUp
bass-metal		U+E79E	pictBeaterMetalBassDrumUp
bass-double		U+E7A0	pictBeaterDoubleBassDrumUp
hammer-plastic		U+E7CD	pictBeaterHammerPlasticUp
hammer-wood		U+E7CB	pictBeaterHammerWoodUp
hammer-metal		U+E7CF	pictBeaterHammerMetalUp
stick		U+E7E8	pictDrumStick
stick-snare		U+E7D1	pictBeaterSnareSticksUp
stick-jazz		U+E7D3	pictBeaterJazzSticksUp

triangle		U+E7D5	pictBeaterTriangleUp
triangle-plain		U+E7EF	pictBeaterTrianglePlain
wound-soft		U+E7B7	pictWoundSoftUp
wound-hard		U+E7B3	pictWoundHardUp
hand		U+E7E3	pictBeaterHand
hand-finger		U+E7E4	pictBeaterFinger
hand-fist		U+E7E5	pictBeaterFist
hand-fingernail		U+E7E6	pictBeaterFingernails
superball		U+E7AE	pictBeaterSuperballUp
metal		U+E7C7	pictBeaterMetalUp
brass		U+E7D9	pictBeaterBrassMalletsUp
brushes		U+E7D7	pictBeaterWireBrushesUp
mallet		U+E7DF	pictBeaterMallet

Examples:

ORIENTATION	#N	#NE	#E	#SE	#S	#SW	#W	#NW
xyl-medium								
bass-metal								
hand-finger								

Electronic music symbols

```
\ekm-fader LEVEL ORIENTATION
\ekm-midi LEVEL ORIENTATION
```

Draw a fader (volume control) and a MIDI controller, respectively, as markup. For the thumb position, the level is rounded to the nearest integral percent value, limited to 100. If this is a multiple of 20, the respective precomposed glyph is used. Else the empty control and the thumb glyphs are combined. Note that they are [Ekmelos](#) specific for the MIDI controller.

- LEVEL ≥ 0 is a percent value.
- LEVEL < 0 is a decibel (dB) value, e.g. -6.0 is equivalent to 50.

It is drawn as a label next to the control according to [ORIENTATION](#) or #f for no label.

Used properties:

- label-format (#f): #f uses "~a%" for percent and "~adB" for decibel values.
- font-size (0)
- label-size (-4) relative to the font size.
- padding (0.3)

0		U+EB2E	elecVolumeLevel0
	:		
100		U+EB33	elecVolumeLevel100
or		U+EB2C	elecVolumeFader
		U+EB2D	elecVolumeFaderThumb
0		U+EB36	elecMIDIController0
	:		
100		U+EB3B	elecMIDIController100
or		U+F6D2	elecMIDIController
		U+F6D3	elecMIDIControllerThumb

Examples:

ORIENTATION	#N	#NE	#E	#SE	#S	#SW	#W	#NW
\ekm-fader								
	-0.18dB	-0.9dB	-2.5dB	-4.4dB	-6dB	-8dB	-17dB	-25dB
\ekm-midi								
	-0.18dB	-0.9dB	-2.5dB	-4.4dB	-6dB	-8dB	-17dB	-25dB

Other symbols










---

\ekm-fermata STYLE

Draw a fermata as markup. The style can be one of the following symbols.



Used property:

- direction

default		U+E4C0	fermataAbove
		U+E4C1	fermataBelow
short		U+E4C4	fermataShortAbove
		U+E4C5	fermataShortBelow
long		U+E4C6	fermataLongAbove
		U+E4C7	fermataLongBelow
veryshort		U+E4C2	fermataVeryShortAbove
		U+E4C3	fermataVeryShortBelow
verylong		U+E4C8	fermataVeryLongAbove
		U+E4C9	fermataVeryLongBelow
extrashort		U+F69E	fermataExtraShortAbove
		U+F69F	fermataExtraShortBelow
extralong		U+F6A0	fermataExtraLongAbove
		U+F6A1	fermataExtraLongBelow
henzeshort		U+E4CC	fermataShortHenzeAbove
		U+E4CD	fermataShortHenzeBelow
henzelong		U+E4CA	fermataLongHenzeAbove
		U+E4CB	fermataLongHenzeBelow

\ekm-eyeglasses DIRECTION

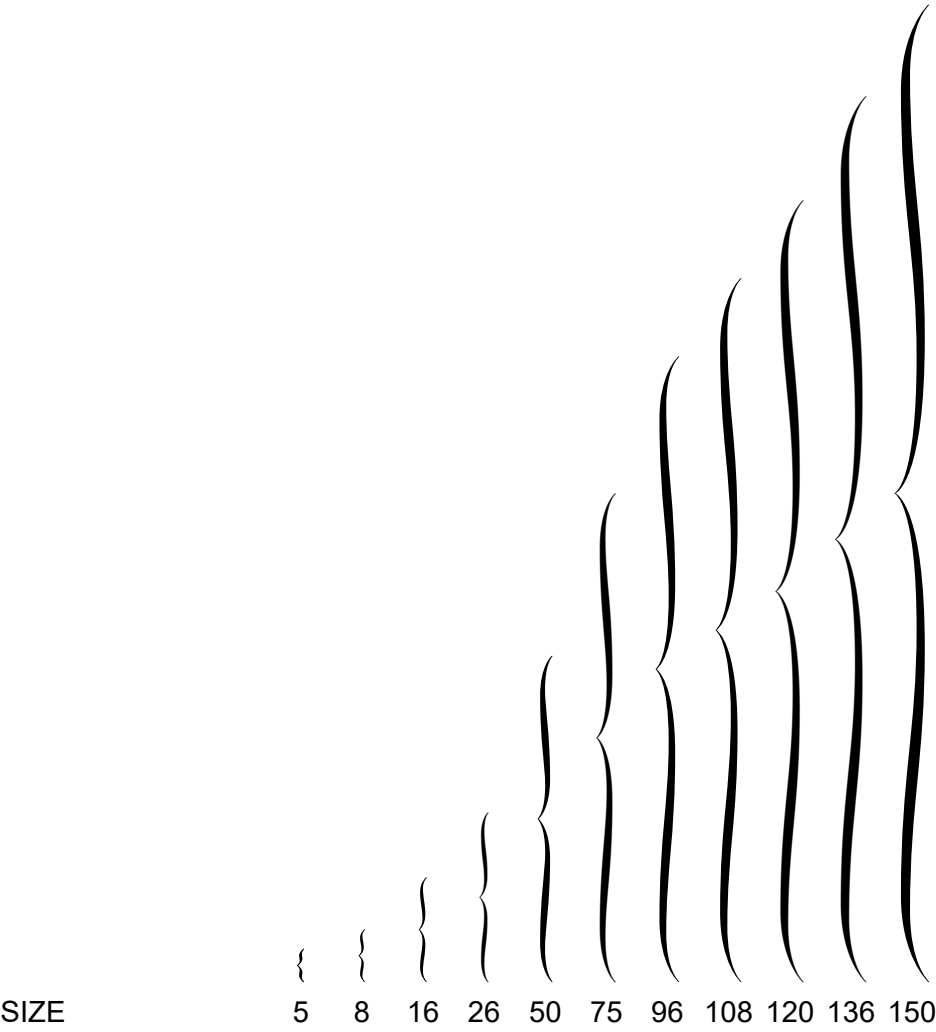
Draw eyeglasses as markup.

LEFT		U+EC62	miscEyeglasses
RIGHT		U+F65F	miscEyeglassesRight

\ekm-brace SIZE DIRECTION

Draw a brace as markup with the size (height) in staff units. It makes use of the [Ekmelos](#) specific size variants, each for a specific range of sizes.

0 – 5	U+F706	brace005
	U+F707	reversedBrace005
6 – 11	U+E000	brace
	U+E001	reversedBrace
12 – 23	U+F708	brace020
	U+F709	reversedBrace020
24 – 39	U+F70A	brace040
	U+F70B	reversedBrace040
40 – 55	U+F70C	brace060
	U+F70D	reversedBrace060
56 – 71	U+F70E	brace080
	U+F70F	reversedBrace080
72 – 87	U+F710	brace100
	U+F711	reversedBrace100
88 – 103	U+F712	brace120
	U+F713	reversedBrace120
104 – 119	U+F714	brace140
	U+F715	reversedBrace140
120 – 135	U+F716	brace160
	U+F717	reversedBrace160
136 – ...	U+F718	brace180
	U+F719	reversedBrace180



\ekm-note-by-number STYLE LOG DOTS DIRECTION

Draw a note with augmentation dots as markup. It does not support stem lengths. The style can be one of the following symbols or one of the [notehead style symbols](#) . LOG can be in the range -2 (or -1) thru 10. Some styles have notes only for LOG ≤ 5 (32nd note) with stem up.

note

U+F637

noteheadLongaUp

U+F638

noteheadLongaDown

U+E1D0

noteDoubleWhole

U+E1D2

noteWhole

U+E1D3

noteHalfUp


U+E1D4

noteHalfDown

:

U+E1E6

note1024thDown



metronome

U+ECA0

metNoteDoubleWhole

U+ECA2

metNoteWhole

U+ECA3

metNoteHalfUp


U+ECA4

metNoteHalfDown

:

U+ECB6

metNote1024thDown



straight

U+F637

noteheadLongaUp

U+E1D0

noteDoubleWhole

U+E1D2

noteWhole

U+E1D3

noteHalfUp

U+E1D5

noteQuarterUp

U+F683


note8thUpStraight

U+F686

note16thUpStraight

U+F689

note32ndUpStraight



short

U+F637

noteheadLongaUp

U+E1D0

noteDoubleWhole

U+E1D2

noteWhole

U+E1D3

noteHalfUp

U+E1D5

noteQuarterUp

U+F684

note8thUpShort

U+F687

note16thUpShort

U+F68A

note32ndUpShort



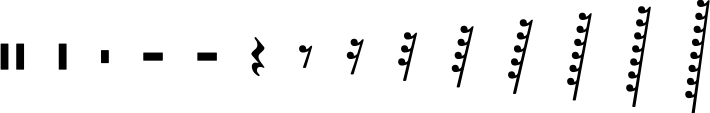
beamed	U+F637	noteheadLongaUp
	U+E1D0	noteDoubleWhole
	U+E1D2	noteWhole
	U+E1D3	noteHalfUp
	U+E1D5	noteQuarterUp
	U+F685	note8thUpBeamed
	U+F688	note16thUpBeamed
	U+F68B	note32ndUpBeamed



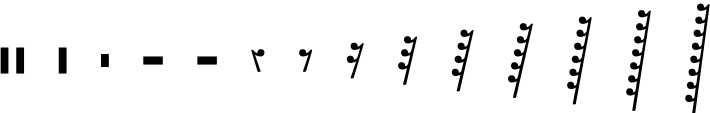
```
\ekm-rest-by-number STYLE LOG DOTS
```

Draw a rest with augmentation dots as markup. It does not support multi-measure-rests. The style can be one of the following symbols. LOG can be in the range -3 thru 10.

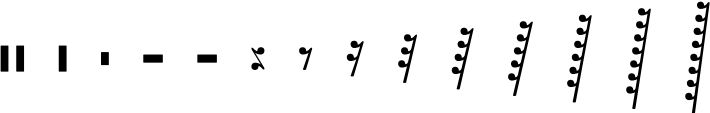
default	U+E4E0	restMaxima
	:	
	U+E4E5	restQuarter
	:	
	U+E4ED	rest1024th



classical	U+E4E0	restMaxima
	:	
	U+E4F2	restQuarterOld
	:	
	U+E4ED	rest1024th



z	U+E4E0	restMaxima
	:	
	U+E4F6	restQuarterZ
	:	
	U+E4ED	rest1024th





`\ekm-metronome COUNT`

Draw COUNT metronome strokes as markup, i.e. the glyph `noteTick` (U+F614) which is [Ekmelos](#) specific. COUNT is a positive integer.

Used property:

- `stroke-space (1)`

`\ekmMetronome MUSIC`

Attach metronome strokes to each note, chord, or rest in MUSIC as a horizontally centered markup above the staff, using `\ekm-metronome`. The number of strokes equals the number of quarter note values of the respective duration (possibly rounded up).

## Examples:

`\ekmMetronome ...`

`c4`

`c2`

`c2.`

`<g c>1`

`r4`

`r1*5/4`

`R1`



## General markup commands

---

These are the basic functions of Esmuflily to draw music symbols.

`\ekm-str STRING`

Draw the string, e.g. from `(ly:wide-char->utf-8)` for a single code point.

`\ekm-char CODEPOINT`

Draw the glyph of the code point, or nothing (the empty string) for zero.

`\ekm-char ##xE048`



`\ekm-char ##xEB27`



`\ekm-charf CODEPOINT FEATURES`

Draw the glyph of the code point with font features. FEATURES is either a list of one or more strings or the number of a stylistic alternate. #1 and #' (1) and #' ("salt 1") are equivalent. #0 and #' () do not set font features, i.e. they behave like `\ekm-char`.

`\ekm-charf ##xE4A0 #0`



`\ekm-charf ##xE4A0 #1`



`\ekm-charf ##xE4A0 #2`



`\ekm-cchar CENTER CODEPOINT`

Draw the glyph of the code point, centered horizontally if CENTER is 1 or 3 (bit 0), and vertically if CENTER is 2 or 3 (bit 1).

`\ekm-chars CODEPOINT-LIST`

Draw the glyphs of the code points in the list concatenated to a string, or nothing (the empty string) for an empty list. This can be used to draw a ligature by specifying the code points of the corresponding character sequence.

`\ekm-chars #'(#xE260 #xE2B4 #xE2B2)`



`\ekm-chars #'(#xE262 #xE566 #xEAA6 #xEAA5)`



`\ekm-chars #'(#xE1F0 #xE1F7 #xE1FC #xE1F7 #xE1F4)`



`\ekm-text EXTEXT`

Draw **EXTEXT**. Depending on the argument type, it behaves like `\ekm-char`, `\ekm-charf`, or `\ekm-chars`, or it draws markup.

`\ekm-text #'(#xE4A0 0)`



`\ekm-text #'(#xE4A0 2)`



`\ekm-text #'(#xE4A0 #xE4A4)`



`\ekm-ctext CENTER EXTEXT`

Draw **EXTEXT** . Markup is centered like `\ekm-cchar` . A list of code points is centered only horizontally. A single code point (possibly with font features) is never centered. This command is intended to draw symbols on stem.

`\ekm-line EXTEXT-LIST`

Draw the **EXTEXTs** in the list in a horizontal line.

Used properties:

- `word-space`
- `text-direction`

`\ekm-line #'(#xE046 "al fine")`

D.C. al fine

`\ekm-line #'(#xE6D0 "with" #xE78E)`

 with 

`\ekm-line #'((#xE6D0 1) "with" #xE78E)`

 with 




`\ekm-def MAP DEFINITION`

Draw a text according to **DEFINITION** .

MAP is an alist of **EXTEXTs** mapped onto key strings. A key which is a prefix of other keys must be arranged after them in MAP, i.e. the correct order is "abc", "ab", "a" . A common key ( " " , "\_" , etc.) can be overridden. The special value `#f` draws nothing, i.e. the key is simply ignored.

```
#(define my-map `(
  (".|:" . #xE040)
  ("tr#~" . (#xE262 #xE566 #xEAA6 #xEAA5))
  ("timp" . (#xE6D0 1))
  (" " . #f)
  ("w" . "with")
  ("box/" . , (markup #:box #:ekm-beater 'timpani-medium NE))
))
```

`\ekm-def #my-map #".|:___tr#~"`

:  

`\ekm-def #my-map #"timp w box/"`

 with 

`\ekm-label ORIENTATION LABEL ARG`

Combine a markup with another markup placed as a label next to it according to **ORIENTATION** (= `#f` ignores the label).


Used properties:

- `font-size (0)`
- `label-size (-4) relative to the font size.`
- `padding (0.3)`

`\ekm-label #SE \ekm-char ##xE836 "G"`

G<sup>3</sup>

`\ekm-label #NW "Medium" \ekm-char ##xE78E`

Medium 

\ekm-number CODEPOINT NUMBER

Draw the integer number as a decimal digit string. CODEPOINT is either the code point of digit 0, or a vector with the code points of digit 0 - 9.

\ekm-number ##xE880 #1234

***1234***

\ekm-number ##xEA70 #2345

**2345**

```
#(define my-dig '#(
  #xEA50 #xEA51 #xEA52 #xEA54 #xEA55
  #xEA57 #xEA5B #xEA5D #xEA60 #xEA61))
\ekm-number #my-dig #3456
```

**3456**

\ekm-combine CODEPOINT X Y CODEPOINT2

Combine the glyphs of the code points, where the second is translated by X,Y scaled.

\ekm-combine ##xECA5 #-0.5 #1.0 ##xE56E



\ekm-combine ##xEA7F #0.3 #0 ##xE87B



## Extended text

---

Some commands accept an EXTEXT value (or a list of EXTEXTs) which is one of the following:

A single code point (integer). See `\ekm-char`.

```
##xE695
```

A list of a single code point followed by font features, i.e. one or more strings or a number less than 32 of a stylistic alternate. Higher values are treated as code points. See `\ekm-charf`.

```
#' (#xE626 "salt 2")
```

```
#' (#xE626 2)
```

A list of one or more code points. See `\ekm-chars`.

```
#' (#xE260 #xE567 #xE262)
```

Any markup. Note that the commands `\ekmTremolo` and `\ekmStem` interpret some strings to draw predefined symbols.

```
#"poco a poco"
```

```
#(markup #:box #:ekm-char #xED19)
```

# Definition string

---

Some commands and properties accept a DEFINITION value. This is a string of one or more keys, each consisting of one or more characters. Their corresponding values (mostly single glyphs) are stacked in a line. Any other character in the string produces a warning and only the text created so far is drawn.

## Common keys

These keys are always applicable but can be overridden in the MAP specified with `\ekm-def`.

<code>&lt;space&gt;</code>	U+0020	space
<code>—</code>	U+200A	hairspace
<code>—</code>	U+2009	thinspace
<code>—</code>	U+2002	enspace
<code>—</code>	U+2003	emspace

## Orientation

---

Some commands accept an ORIENTATION value. This is the sum of axis (0, 1, or  $\pm 0.5$  for diagonal) and direction ( $\pm 1$ ). The following symbols are defined for the 12 possible values. The last four values are intended for “bilateral” orientations. Currently, only `\ekm-arrow` supports them for a few styles.

An unsupported value is substituted with N .

N	2	Y	+ UP
NE	1.5	0.5	+ UP
E	1	X	+ RIGHT
SE	0.5	-0.5	+ RIGHT
S	0	Y	+ DOWN
SW	-0.5	0.5	+ DOWN
W	-1	X	+ LEFT
NW	-1.5	-0.5	+ LEFT
NS	-2	Y	+ -3
NESW	-2.5	0.5	+ -3
EW	-3	X	+ -3
SENW	-3.5	-0.5	+ -3