LilyPond
Il compositore tipografico per la musica

Frammenti
Il team di sviluppo di LilyPond

Questo documento contiene una selezione di frammenti LilyPond tratti dal LilyPond Snippet Repository (https://lsr.di.unimi.it) (LSR). I frammenti sono nel pubblico dominio.

Desideriamo ringraziare Sebastiano Vigna per la gestione del sito web e del database dell’LSR, e l’università di Milano per l’hosting.

Questo documento non è un sottoinsieme esatto dell’LSR: alcuni frammenti vengono dalla directory input/new dei sorgenti di LilyPond; i frammenti tratti dall’LSR vengono convertiti con convert-ly, perché l’LSR usa una versione stabile di LilyPond, mentre questo manuale è valido per la versione 2.24.3.

I frammenti sono raggruppati per categorie; le categorie elencate nell’indice corrispondono a una sezione del manuale di notazione di LilyPond. Un frammento potrebbe avere più di una categoria, e non tutte le categorie dell’LSR saranno presenti in questo documento.

Nella versione HTML di questo manuale, si può fare clic sul nome del file o sull’immagine di ciascun esempio per vedere il corrispondente file di input.

La documentazione completa si trova all’indirizzo https://lilypond.org/.

Questo documento è stato posto nel pubblico dominio.
Per la versione di LilyPond 2.24.3
Sommario

Pitches ........................................................................ 1
Un ambitus per voce .................................................. 1
Aggiungere un segno di ottava a una sola voce .......... 1
Teste di nota Aiken variante sottile ......................... 2
Altering the length of beamed stems ....................... 2
Ambitus ...................................................................... 3
Ambitus dopo armatura di chiave ........................... 3
Ambitus su più voci ................................................ 4
Applicazione degli stili delle teste di nota in base al grado della scala .... 4
Cambiare automaticamente la direzione del gambo della
nota centrale in base alla melodia .......................... 5
Changing ottava text ............................................... 6
Modifica dell’intervallo dell’ambitus ....................... 6
Changing the interval of lines on the stave ............. 7
Clefs can be transposed by arbitrary amounts ........ 8
Coloring notes depending on their pitch ................. 8
Creating a sequence of notes on various pitches .... 9
Creating custom key signatures ............................ 10
Direction of merged ‘fa’ shape note heads ............. 10
Force a cancellation natural before accidentals .... 11
Forcing a clef symbol to be displayed .................... 11
Generating random notes ....................................... 11
Nascondere le alterazioni delle note con legatura di valore .... 12
Keep change clefs full sized ................................ 12
Esempio di makam .................................................. 13
Modificare l’inclinazione dell’estensore dell’ottava ... 13
Armature di chiave non tradizionali ..................... 14
Numeri dentro le teste di nota .............................. 14
Modello per orchestra, coro e pianoforte ............. 16
Impedire l’inserimento automatico dei bequadri supplementari ... 19
Impedire l’inserimento dei segni di bequadro quando cambia l’armatura di chiave .... 20
Quoting another voice with transposition ............. 20
Separating key cancellations from key signature changes .... 21
Trasposizione delle altezze con numero minimo di alterazioni ... 21
Turkish Makam example ....................................... 23
Modifiche manuali della proprietà della chiave ......... 23
Using autochange with more than one voice ......... 25

Rhythms ................................................................. 27
Adding beams, slurs, ties etc. when using tuplet and non-tuplet rhythms .... 27
Adding drum parts ................................................. 27
Adjusting grace note spacing ................................ 28
Allineare i numeri di battuta ................................ 28
Note brevi alternative ........................................... 29
Appoggiatura or grace note before a bar line ........ 29
Automatic beam subdivisions ............................ 30
Automatically change durations .......................... 30
Estremità delle travature nel contesto Score .......... 31
Travature che attraversano le interruzioni di linea .......................... 32
Modificare la distanza delle travature angolari .................................. 32
Modificare la forma delle pause multiple ........................................... 33
Modifica del numero di punti di aumentazione per nota ........................ 33
Modificare il tempo senza mostrare l’indicazione metronomica ............... 33
Modifica del numero del gruppo irregolare ........................................ 34
Changing time signatures inside a polymetric section using \scaleDurations .......................... 34
Notazione per canti e salmi .................................................................. 35
Compound time signatures .................................................................... 36
Segni per la conduzione, segni di raggruppamento della misura .............. 36
Consistently left aligned bar numbers .................................................... 37
Controllare la visibilità della parentesi del gruppo irregolare ................. 38
Cow and ride bell example ................................................................. 39
Creare indicazioni metronomiche in modalità testuale ............................ 39
Disegnare manualmente le legature di valore ....................................... 40
Engraving tremolos with floating beams ............................................... 40
Inserire vari gruppi irregolari usando una sola volta il comando \tuple .......................... 41
Code e punte delle travature .............................................................. 41
Forcing rehearsal marks to start from a given letter or number ............... 43
Generating custom flags ...................................................................... 43
Ritmi di accompagnamento per chitarra .............................................. 44
Heavily customized polymetric time signatures .................................... 45
High and Low woodblock example ....................................................... 46
Making an object invisible with the `transparent property ...................... 47
Legature di portamento con complesse strutture di tratteggio .................... 48
Manually controlling beam positions ..................................................... 48
Merging multi-measure rests in a polyphonic part ................................. 49
Modifying tuplet bracket length ........................................................ 49
Spostare le note puntate in polifonia .................................................... 50
Regolazione della lunghezza delle pause multiple ................................ 50
Testo a margine delle pause multiple .................................................. 51
Numeri non predefiniti per i gruppi irregolari ........................................ 51
Numbering single measure rests .......................................................... 52
PartCombine e autoBeamOff ............................................................. 52
Percussion example ............................................................................ 53
Consentire l’interruzione del rigo all’interno di gruppi irregolari con travature .......................... 54
Positioning grace note beams at the height of normal note beams ........... 55
Posizionare gli abbellimenti con dello spazio fluttuante ......................... 55
Posizionamento delle pause multiple .................................................... 56
Positioning opposing fermatas on a bar line ........................................ 57
Preventing final mark from removing final tuplet .................................. 58
Mostrare i numeri di battuta a intervalli regolari .................................. 58
Printing bar numbers for broken measures ......................................... 59
Numeri di battuta racchiusi in rettangoli o cerchi ................................ 60
Printing bar numbers using modulo-bar-number-visible ........................ 60
Stampare i numeri di battuta a intervalli regolari variabili ....................... 61
Posizionare il metronomo e i numeri di chiamata sotto il rigo ................. 61
Printing music with different time signatures ...................................... 62
Mostrare il numero di battuta nella prima misura ................................... 65
Printing tuplet brackets on the note head side .................................... 65
Redefine le impostazioni predefinite globali degli abbellimenti ............... 66
Togliere i numeri di battuta da uno spartito ........................................ 67
Removing connecting bar lines on StaffGroup, PianoStaff, or GrandStaff .. 67
Stili di pausa ..................................................
Expressive marks .......................................................... 79

Adding beams, slurs, ties etc. when using tuplet and non-tuplet rhythms ........................................ 79
Adding parentheses around an expressive mark or chordal note ................................................................. 79
Aggiungere i segni di tempo per gli slussi lunghi .................................................. 79
Cambiare la forma dei portamenti indeterminati verso il basso o verso l’alto ........................................ 80
Allineare le estremità delle forcele alle direzioni di NoteColumn .......................................................... 81
Note brevi alternative .......................................................... 81
Asymmetric slurs .................................................................. 81
Breathing signs .................................................................... 82
Broken Crescendo Hairpin ..................................................... 83
Caesura ("railtracks") with fermata ........................................ 83
Center text below hairpin dynamics ..................................... 84
Nascondere la linea di estensione per le dinamiche testuali ................................................................. 85
Changing the appearance of a slur from solid to dotted or dashed ...................................................... 86
Cambiare il simbolo del segno di respiro ................................ 86
Modifica del numero di punti di aumentazione per nota ................................................................. 87
Combining dynamics with markup texts .................................. 87
Glissando contemporaneo .................................................... 87
Controlling spanner visibility after a line break ....................... 88
Controllo dell’ordine verticale degli script ............................... 88
Creare un gruppetto ritardato ............................................. 89
Creare degli arpeggi che attraversano note appartenenti a voci diverse ............................................. 89
Creare degli arpeggi che attraversano il rigo del pianoforte .............................................................. 90
Creare degli arpeggi che attraversano i righi in altri contesti .......................................................... 90
Creating double-digit fingerings ........................................... 91
Creating "real" parenthesized dynamics ................................... 91
Creating slurs across voices ............................................... 92
Creating text spanners ......................................................... 92
Estensione testuale della dinamica personalizzato .................. 93
Estensione testuale della dinamica personalizzato ............... 94
Glissandi can skip grobs ...................................................... 94
Hairpins with different line styles ........................................ 95
Nascondere la linea di estensione per le dinamiche testuali .......................................................... 95
Horizontally aligning custom dynamics (e.g. "sempre pp", "più f", "subito p") ........................................ 95
Inserire una cesura ................................................................. 99
Laissez vibrer ties .......................................................... 99
Line arrows ........................................................................ 100
Legature di portamento con complesse strutture di tratteggio .......................... 100
Modificare i valori predefiniti per le abbreviazioni delle articolazioni ........ 101
Moving slur positions vertically ........................................... 101
Spostare le estremità delle forcelle ........................................ 102
Positioning arpeggios .......................................................... 103
Posizionare il testo a margine dentro le legature di portamento .................... 103
Stampare le forcelle in vari stili ............................................ 103
Forcelle con notazione al niente ........................................... 104
Posizionare il metronomo e i numeri di chiamata sotto il rigo ....................... 104
Impostare il comportamento delle forcelle sulle stanghette ....................... 105
Impostare la lunghezza minima delle forcelle ...................................... 105
Showing the same articulation above and below a note or chord .................... 106
Snap-pizzicato or Bartok pizzicato ......................................... 106
Usare un segno di spunta come simbolo di respiro .................................... 107
Usare arpeggioBracket per rendere i divisi più visibili ............................... 107
Uso delle doppie legature di portamento per gli accordi legati .................... 108
Uso della proprietà whiteout .................................................................. 108
Vertical line as a baroque articulation mark ............................................. 108
Vertically aligning dynamics across multiple notes ..................................... 109

Repeats ................................................................. 110
Aggiungere le parentesi delle volte a altri righi ......................................... 110
Centered measure numbers .................................................... 110
Changing the default bar lines .................................................................. 111
Tremoli attraverso i righi ........................................................................ 112
Engraving tremolos with floating beams .................................................... 113
Ripetizioni con segni di percentuale isolati ............................................ 113
Measure counter .......................................................................... 113
Numbering groups of measures ........................................................... 114
Visibilità del conto della ripetizione con segno percentuale ........................ 115
Contatore della ripetizione con segno percentuale ..................................... 115
Positioning segno and coda (with line break) .......................................... 116
Impostare la doppia ripetizione predefinita per le volte ............................... 117
Accorciare le parentesi delle volte ..................................................... 117
Voce sotto gli accordi ........................................................................ 117
Voce multi staff ............................................................................ 118
Voce text markup using repeatCommands ............................................ 119

Simultaneous notes ......................................................... 120
Voci ulteriori per evitare le collisioni ..................................................... 120
Changing a single note’s size in a chord .................................................. 120
Modificare le indicazioni testuali di partCombine ..................................... 121
Clusters .................................................................................. 121
Combinare due parti sullo stesso rigo ..................................................... 122
Displaying complex chords .................................................................... 123
Forzare lo spostamento orizzontale delle note ........................................ 123
Making an object invisible with the ‘transparent property ......................... 124
Spostare le note puntate in polifonia ...................................................... 124
Suppressing warnings for clashing note columns .................................... 125
Two \partCombine pairs on one staff .................................................... 125
Staff notation

Un ambitus per voce .......................................................... 128
Adding an extra staff ...................................................... 128
Adding an extra staff at a line break ............................... 129
Adding indicators to staves which get split after a break ...... 130
Aggiungere citazioni orchestrali a una partitura vocale ....... 134
Aggiungere i segni di tempo per i glissandi lunghi .......... 136
Numeri di battuta alternativi ............................................. 137
Ambitus dopo armatura di chiave ................................. 138
Centered measure numbers .............................................. 138
Changing the default bar lines ....................................... 139
Changing the number of lines in a staff .......................... 140
Changing the staff size .................................................. 140
Creating blank staves .................................................... 141
Creating custom key signatures ........................................ 143
Creating double-digit fingerings .................................... 143
Cross staff stems ............................................................ 143
Mostrare la parentesi anche se c’è un solo rigo nel sistema 144
Extending a TrillSpanner .................................................. 145
Estendere i glissandi sulle volte delle ripetizioni .............. 145
Flat Ties ......................................................................... 147
Forcing measure width to adapt to MetronomeMark’s width ... 149
Glissandi can skip grobs .................................................. 150
Incipit ............................................................................. 150
Inserting score fragments above a staff, as markups ......... 155
Let TabStaff print the topmost string at bottom ............... 155
Letter tablature formatting .............................................. 156
Lasciare che i glissandi vadano a capo .............................. 157
Rendere alcune linee del rigo più spesse delle altre ........ 157
Measure counter ............................................................. 158
Formattazione mensurale (stanghette tra i righi) ............. 158
Modificare l’inclinazione dell’estensore dell’ottava ......... 159
Annidare i righi ............................................................... 159
Armature di chiave non tradizionali ............................. 160
Numbering groups of measures ...................................... 161
Modello per orchestra, coro e pianoforte ......................... 162
Print ChordNames with same root and different bass as slash and bass-note 165
Putting lyrics inside the staff .......................................... 167
Quoting another voice .................................................... 168
Quoting another voice with transposition ....................... 169
Removing brace on first line of piano score .................... 170
Eliminare la prima linea vuota ....................................... 170
Setting system separators .............................................. 172
Tick bar lines ................................................................. 174
Time signature in parentheses ....................................... 174
Time signature in parentheses - method 3 ....................... 174
Modifiche manuali della proprietà della chiave .............. 174
Two `partCombine pairs on one staff .............................. 176
Usare una parentesi quadra all’inizio di un gruppo di righi 178
Using autochange with more than one voice .................. 178
Using marklines in a Frenched score .............................. 179
Vertical aligned StaffGroups without connecting SystemStartBar 182
Volta sotto gli accordi .................................................... 190
Volta multi staff ............................................................ 191
Editorial annotations ........................................ 192

Adding fingerings to a score ........................................ 192
Adding links to objects ........................................ 192
Adding markups in a tablature ........................................ 194
Far sì che la diteggiatura appaia dentro il rigo ................. 194
Numeri di battuta alternativi ........................................ 195
Parentesi analitiche sopra il rigo .................................... 196
Parentesi analitiche con etichette .................................... 196
Applicazione degli stili delle teste di nota in base al grado della scala .................... 197
Blanking staff lines using the \whiteout command ...................... 198
Changing a single note’s size in a chord ......................... 198
Changing the appearance of a slur from solid to dotted or dashed ................. 199
Coloring notes depending on their pitch .......................... 199
Controllare il posizionamento delle diteggiature di un accordo .................... 200
Creare un gruppetto ritardato .................................. 201
Creating blank staves ........................................ 201
Creating double-digit fingerings .................................... 201
Direzione predefinita dei gambi sulla linea centrale del rigo ............... 203
Different font size settings for instrumentName and shortInstrumentName ............ 204
Drawing boxes around grobs ........................................ 205
Drawing circles around note heads .................................. 206
Drawing circles around various objects ................................ 206
Embedding native PostScript in a \markup block .................... 206
Modificare l’aspetto delle linee della griglia ...................... 207
Grid lines: emphasizing rhythms and notes synchronization .............. 208
Martellato e strappato ........................................ 209
Martellato e strappato usando accordi .................................. 209
Martellato e strappato usando le voci .................................. 210
Rendere alcune linee del rigo più spesse delle altre ..................... 210
Marking notes of spoken parts with a cross on the stem (Sprechstimme) ............ 210
Measure counter ........................................ 211
Measure spanner ........................................ 212
Numbering groups of measures .................................... 213
Positioning fingering indications precisely .......................... 214
Posizionare il testo a margine dentro le legature di portamento .................... 215
Printing text from right to left .................................... 215
String number extender lines ....................................... 215
Using PostScript to generate special note head shapes .................... 216
Uso della proprietà whiteout ....................................... 216

Text .............................................................. 218

Adding a QR code ........................................ 218
Adding markups in a tablature ....................................... 220
Adding the current date to a score .................................. 221
Adjusting lyrics vertical spacing .................................... 222
Aligning and centering instrument names ................................ 222
Aligning syllables with melisma ...................................... 224
Blanking staff lines using the \whiteout command .................... 224
Center text below hairpin dynamics .................................. 225
Changing ottava text ........................................ 225
Changing the default text font family ................................ 227
Combining dynamics with markup texts ................................ 228
Combinare due parti sullo stesso rigo ................................ 228
Vocal music .................................................. 247

Un ambitus per voce ......................................... 247
Adding indicators to staves which get split after a break .... 247
Aggiungere citazioni orchestrali a una partitura vocale ... 251
Adjusting lyrics vertical spacing .......................... 253
Aligning syllables with melisma ........................... 253
Ambitus ...................................................... 254
Ambitus dopo armatura di chiave .......................... 255
Ambitus su più voci ....................................... 255
Modello per notazione antica – trascrizione moderna di musica gregoriana .... 256
Modello per salmo anglicano ................................ 256
Arranging separate lyrics on a single line ................. 259
Changing stanza fonts ..................................... 260
Notazione per canti e salmi ................................ 261
Forcing hyphens to be shown .............................. 262
Formattazione delle sillabe del testo vocale ............... 262
How to put ties between syllables in lyrics ............... 262
Modello per inno ............................................ 263
Allineamento del testo vocale ............................. 265
Marking notes of spoken parts with a cross on the stem (Sprechstimme) .... 265
Ottenere la spaziatura del testo della vecchia versione 2.12 ........ 266
Modello per orchestra, coro e pianoforte ................... 269
Modello per pianoforte con melodia e testo .............. 272
Putting lyrics inside the staff ............................ 273
Modello per coro SATB - quattro righi .................... 274
Modello di rigo singolo con note, testo e accordi ......... 275
Single staff template with notes, lyrics, chords and frets .... 276
Modello di rigo singolo con note e testo ................. 277
Skips in lyric mode ....................................... 278
Skips in lyric mode (2) .................................. 278
Usare arpeggioBracket per rendere i divisi più visibili ........ 278
Usare le etichette per produrre musica mensurale e moderna dallo stesso sorgente ...................... 279
Allineare verticalmente gli ossia e il testo vocale ............................................................. 281
Vertically centered common lyrics ......................................................................................... 281
Modello per complesso vocale ............................................................................................... 283
Modello per gruppo vocale con riduzione per pianoforte automatica .................................. 285
Modello per gruppo vocale con testo allineato sotto e sopra i righi .................................. 287
Modello per gruppo vocale con strofa e ritornello ............................................................. 289

Chords ................................................................................................................................. 291
Adding a figured bass above or below the notes ................................................................. 291
Aggiungere stanghette al contesto ChordNames ............................................................... 291
Adjusting figured bass alteration glyphs ........................................................................... 292
Modifica del separatore dell’accordo .................................................................................. 292
Changing the chord names to German or semi-German notation ...................................... 292
Cambiare la posizione delle alterazioni del basso continuo ............................................. 293
Eccezioni dei nomi degli accordi ......................................................................................... 294
Nome dell’accordo di settima maggiore ............................................................................ 294
Chord names alternative ...................................................................................................... 295
Chords with stretched fingering for FretBoards and TabVoice ........................................ 305
Clusters .................................................................................................................................. 305
Controllare il posizionamento delle diteggiature di un accordo .......................................... 306
Cross-staff chords - beaming problems workaround ......................................................... 306
Customizing the chord grid style ......................................................................................... 307
Customizing the no-chord symbol ....................................................................................... 308
Displaying complex chords ................................................................................................. 308
Manually break figured bass extenders for only some numbers ........................................ 309
Print ChordNames with same root and different bass as slash and bass-note ................... 309
Mostrare gli accordi nei cambi ............................................................................................ 311
Canzoniere semplice ........................................................................................................... 312
Modello di rigo singolo con note, testo e accordi .............................................................. 312
Single staff template with notes, lyrics, chords and frets .................................................. 313
Modello di rigo singolo con note e accordi ........................................................................ 314
Vertically centering paired figured bass extenders ............................................................ 315
Volta sotto gli accordi .......................................................................................................... 315

Keyboards .......................................................................................................................... 317
Accordions-discount symbols ............................................................................................ 317
Simboli di registro della fisarmonica .................................................................................... 320
Changing the text for sustain markings .............................................................................. 321
Clusters ................................................................................................................................ 321
Controllare il posizionamento delle diteggiature di un accordo .......................................... 322
Creating slurs across voices ............................................................................................... 323
Cross-staff chords - beaming problems workaround ......................................................... 323
Tremolos attraverso i righi .................................................................................................... 324
Fine-tuning pedal brackets ................................................................................................. 325
Indicare accordi trasversali al rigo con la parentesi quadrata dell’arpeggio ..................... 325
Modello per combo jazz ........................................................................................................ 326
Laissez vibrer ties .................................................................................................................. 326
Modello per pianoforte (semplice) ..................................................................................... 333
Modello per pianoforte con testo al centro .......................................................................... 333
Modello per pianoforte con melodia e testo ....................................................................... 334
Removing brace on first line of piano score ...................................................................... 335
Using autochange with more than one voice ..................................................................... 336
Modello per gruppo vocale con riduzione per pianoforte automatica ................................ 337
### Percussion

- Adding drum parts ................................................. 340
- Cow and ride bell example .................................. 341
- Customized drum notation in printed and MIDI output .................................................................. 341
- Heavily customized polyrhythms time signatures ................................................................. 344
- High and Low woodblock example ......................................................................................... 345
- Modello per combo jazz ........................................... 346
- Percussion beaters .................................................. 352
- Percussion example .................................................. 355
- Printing music with different time signatures ......................................................................... 356
- Tam-tam example ..................................................... 359
- Tambourine example ................................................. 360

### Fretted strings

- Adding fingerings to a score .................................. 361
- Adding fingerings to tablatures .............................. 361
- Adding markups in a tablature ................................. 361
- Far si che la diteggiatura appaia dentro il rigo .... 362
- Automatic Fretboards Barre .................................. 363
- Cambiare l’orientamento della tastiera .................. 363
- Glissando di accordi in intavolatura ....................... 364
- Cambi di accordo nei diagrammi dei tasti ............... 365
- Chords with stretched fingering for FretBoards and TabVoice ............................................. 366
- Controllare il posizionamento delle diteggiature di un accordo ............................................ 366
- Personalizzare la tastiera del diagramma dei tasti ................................................................. 367
- Personalizzare il diagramma dei tasti di tipo markup ............................................................ 368
- Definire diagrammi dei tasti predefiniti per altri strumenti .................................................... 370
- Faking a hammer in tablatures .............................. 372
- Diteggiature, indicazioni di corda e diteggiature della mano destra ....................................... 372
- Flamenco notation ................................................... 372
- Fret diagrams explained and developed ............... 376
- Tabelle alternative per i diagrammi dei tasti .......... 383
- Armonici su corde premute in intavolatura .......... 385
- Guitar slides ............................................................ 386
- Ritmi di accompagnamento per chitarra ............... 387
- Martellato e strappato ............................................. 388
- Martellato e strappato usando accordi ................... 388
- Martellato e strappato usando le voci .................... 389
- How to change fret diagram position ..................... 389
- Modello per combo jazz ........................................... 390
- Laissez vibre tie ....................................................... 396
- Let TabStaff print the topmost string at bottom ..... 397
- Letter tablature formatting .................................... 397
- Armonici su corde a vuoto in intavolatura .......... 398
- Posizionamento delle diteggiature della mano destra ......................................................... 400
- Polifonia in intavolatura ........................................... 401
- Slides in tablature ..................................................... 402
- Comportamento di gambi e travature in intavolatura ............................................................. 402
- String number extender lines ............................... 403
Unfretted strings .......................... 404
  Creating slurs across voices .......................... 404
  Dotted harmonics ........................................ 404
  Snap-pizzicato or Bartok pizzicato ....................... 405
  Modello per quartetto d’archi (semplice) .................. 405
  Modello per quartetto d’archi con parti separate .......... 406

Winds ........................................... 410
  Cambiare la dimensione dei diagrammi per legni ............. 410
  Simboli di diteggiatura per strumenti aerofoni ............. 410
  Flute slap notation ...................................... 411
  Diagrammi grafici e testuali per i legni .................... 412
  Grafico della diteggiatura per flauto dolce ................ 412
  Elenco delle chiavi dei diagrammi per legni ................ 414
  Elenco dei diagrammi per i legni .......................... 414

Ancient notation .................................. 418
  Adding a figured bass above or below the notes .............. 418
  Ancient fonts ........................................... 418
  Modello per notazione antica – trascrizione moderna di musica gregoriana 423
  Ancient time signatures ................................ 424
  Notazione per canti e salmi ................................ 424
  Custodes ............................................... 424
  Incipit .................................................. 425
  Formattazione mensurale (stanghette tra i righi) ............ 430
  Stili di pausa .......................................... 430
  Usare le etichette per produrre musica mensurale e moderna dallo stesso sorgente 432
  Vertical line as a baroque articulation mark .................. 433

World music ...................................... 434
  Improvvisazione araba .................................... 434
  Esempio di makam ....................................... 434
  Printing text from right to left ............................ 434
  Turkish Makam example ................................... 435

Contexts and engravers ............................ 436
  Adding a figured bass above or below the notes .............. 436
  Adding an extra staff .................................... 436
  Adding an extra staff at a line break ....................... 437
  Cambiare automaticamente la direzione del gambo della nota centrale in base alla melodia ...................... 438
  Centered measure numbers ................................ 439
  Impostare l’output MIDI su un canale per voce .............. 440
  Changing time signatures inside a polymetric section using \scaleDurations ................. 441
  Notazione per canti e salmi ................................ 441
  Creating blank staves .................................... 442
  Creating custom key signatures ............................. 444
  Cross staff stems ........................................ 444
  Defining an engraver in Scheme: ambitus engraver ........... 445
  Displaying a whole GrandStaff system if only one of its staves is alive .................. 451
  Engravers one-by-one .................................... 453
  Formattazione mensurale (stanghette tra i righi) ............ 458
Tweaks and overrides ........................................ 471

Annidare i righi ........................................... 459
Numbering groups of measures ............................ 460
Print ChordNames with same root and different bass as slash and bass-note ............ 461
Togliere i numeri di battuta da uno spartito ........ 463
Usare una parentesi quadra all'inizio di un gruppo di righi ...................... 464
Using marklines in a Frenched score .................... 464
Modello per gruppo vocale con testo allineato sotto e sopra i righi ............ 467
Modello per gruppo vocale con strofa e ritornello .......................... 468

Adding links to objects .................................. 471
Adding markups in a tablature ............................ 471
Aggiungere i segni di ottava a una sola voce .......................... 473
Parentesi analitiche sopra il rigo .................................. 477
Parentesi analitiche con etichette .................................. 478
Asymmetric slurs ......................................... 479

Changing a single note’s size in a chord .................. 480
Changing beam thickness and spacing ...................... 480
Modificare la forma delle pause multiple .................... 480
Changing properties for individual grobs ...................... 481
Nascondere la linea di estensione per le dinamiche testuali ...................... 481
Changing the default text font family ...................... 482

Changing the staff size .................................... 483
Modificare il tempo senza mostrare l’indicazione metronomica .................... 483
Changing the text for sustain markings ...................... 484
Controlling spanner visibility after a line break ...................... 484
Controllo dell’ordine verticale degli script .......................... 485
Controllare la visibilità della parentesi del gruppo irregolare .................. 485

Creare un gruppetto ritardato .................................. 486
Creating custom key signatures ............................. 487
Creating double-digit fingerings .................................. 487
Creating text spanners ...................................... 487
Cross-staff chords - beaming problems workaround ...................... 488
Cross staff stems ........................................ 489

Personalizzare la tastiera del diagramma dei tasti .................................. 491
Personalizzare il diagramma dei tasti di tipo markup .......................... 492
Mostrare la parentesi anche se c'è un solo rigo nel sistema .................... 494
Displaying grob ancestry .................................... 494
Dotted harmonics ........................................ 496

Drawing circles around various objects ...................... 497
Estensore testuale della dinamica personalizzato ...................... 498

Extending a TrillSpanner .................................... 499
Estendere i glissandi sulle volte delle ripetizioni .......................... 499
Fine-tuning pedal brackets .................................... 501
Flat Ties ........................................ 501
Generating custom flags ................................................................. 504
Fret diagrams explained and developed ......................................... 504
Generating custom flags ................................................................. 511
Glissandi can skip grobs ................................................................. 512
Hairpins with different line styles ............................................... 512
Horizontally aligning custom dynamics (e.g. "sempre pp", "più f", "subito p") ................................................................. 513
How to change fret diagram position ............................................. 516
Inserire una cesura ...................................................................... 517
Keep change clefs full sized ......................................................... 518
Line arrows .................................................................................. 518
Making an object invisible with the 'transparent property .............. 519
Lasciare che i glissandi vadano a capo .......................................... 519
Manually controlling beam positions ............................................ 524
Measure-centered bar numbers ...................................................... 520
Formatazione mensurale (stanghette tra i righi) .............................. 521
Modificare l’inclinazione dell’estensore dell’ottava ....................... 522
Spostare le note puntate in polifonia ............................................ 522
Moving slur positions vertically ...................................................... 523
Annidare i righi ........................................................................... 524
Overriding articulations by type ................................................... 524
Overriding articulations of distinct type ........................................ 526
Visibilità del conto della ripetizione con segno percentuale .......... 528
Positioning arpeggios .................................................................. 528
Posizionamento delle pause multiple ........................................... 529
Posizionare il testo a margine dentro le legature di portamento ...... 530
Numeri di battuta racchiusi in rettangoli o cerchi ......................... 530
Posizionare il metronomo e i numeri di chiamata sotto il rigo ....... 530
Printing note names with and without an octave marker ............... 531
Printing tuplet brackets on the note head side .............................. 532
Proportional strict notespacing ..................................................... 532
Removing brace on first line of piano score .................................. 533
Removing connecting bar lines on StaffGroup, PianoStaff, or GrandStaff ................................................................. 534
Eliminare la prima linea vuota ...................................................... 534
Stili di pausa .............................................................................. 536
Rhythmic slashes ....................................................................... 537
Separating key cancellations from key signature changes ............. 537
Impostare il comportamento delle forcelle sulle stanghette .......... 538
Setting system separators .............................................................. 538
Showing the same articulation above and below a note or chord ...... 541
String number extender lines ......................................................... 541
Suppressing warnings for clashing note columns ......................... 542
Time signature in parentheses ...................................................... 542
Time signature in parentheses - method 3 ..................................... 542
Indicazione di tempo che mostra solo il numeratore (invece della frazione) ................................................................. 543
Tuplet bracket and change staff .................................................... 543
Modifiche manuali della proprietà della chiave ............................ 544
Modificare l’aspetto degli abbellimenti di un intero brano .......... 546
Using alternative flag styles .......................................................... 546
Using ly:grob-object to access grobs with \	weak ......................... 547
Using PostScript to generate special note head shapes ................. 548
Using the \tweak command to tweak individual grobs ................. 549
Dinamiche e segni testuali allineati verticalmente ........................ 549
Allineare verticalmente gli ossia e il testo vocale ........................ 550
Vertically centering paired figured bass extenders .................................................. 551

**Paper and layout** ................................................................. 552
Aligning and centering instrument names ................................................................. 552
Arranging separate lyrics on a single line ................................................................. 553
Book parts ......................................................................................... 554
Changing the staff size ......................................................................................... 558
Clip systems ................................................................................................. 559
Creating blank staves ......................................................................................... 562
Demonstrating all headers .................................................................................... 563
Setting system separators ..................................................................................... 564
Table of contents ................................................................................................. 567
Vertical aligned StaffGroups without connecting SystemStartBar ......................... 568

**Titles** ......................................................................................... 577
Adding the current date to a score ........................................................................... 577
Aligning and centering instrument names ................................................................ 577
Demonstrating all headers ..................................................................................... 579
Outputting the version number .............................................................................. 580

**Spacing** ......................................................................................... 581
Adjusting lyrics vertical spacing ............................................................................. 581
Far sì che la diteggiatura appaia dentro il rigo ......................................................... 581
Page label ........................................................................................................... 582
Proportional strict notes spacing ........................................................................... 583
Dinamiche e segni testuali allineati verticalmente ................................................... 583
Allineare verticalmente gli ossia e il testo vocale .................................................... 584

**MIDI** ............................................................................................. 586
Impostare l'output MIDI su un canale per voce ........................................................ 586
Modificare il tempo senza mostrare l'indicazione metronomica ................................. 587
Creare dinamiche personalizzate nell'output MIDI .................................................. 587
Customized drum notation in printed and MIDI output ............................................. 588
Demo MidiInstruments ........................................................................................... 590
Modificare l'equalizzazione predefinita degli strumenti MIDI .................................... 594

**Templates** ....................................................................................... 596
Modello per notazione antica – trascrizione moderna di musica gregoriana .............. 596
Modello per salmo anglicano ................................................................................... 596
Modello per inno .................................................................................................... 599
Modello per combo jazz ........................................................................................... 601
Modello per orchestra, coro e pianoforte .................................................................. 607
Modello per pianoforte ( semplice) ........................................................................... 611
Modello per pianoforte con testo al centro ............................................................... 612
Modello per pianoforte con melodia e testo ............................................................. 613
Modello per coro SATB - quattro righi ..................................................................... 614
Score for diatonic accordion .................................................................................... 615
Modello di rigo singolo con note, testo e accordi ...................................................... 620
Single staff template with notes, lyrics, chords and frets .......................................... 621
Modello di rigo singolo con note e accordi ............................................................... 622
Modello di rigo singolo con note e testo .................................................................. 622
Modello di rigo singolo con solo note ..................................................................... 623
Modello per quartetto d'archi (semplice) .................................................. 624
Modello per quartetto d'archi con parti separate.............................................. 625
Modello per complesso vocale ................................................................. 628
Modello per gruppo vocale con riduzione per pianoforte automatica............... 630
Modello per gruppo vocale con testo allineato sotto e sopra i righi .................. 632
Modello per gruppo vocale con strofa e ritornello ....................................... 634
Pitches

Sezione “Pitches” in Guida alla Notazione

Un ambitus per voce

L’ambitus può essere specificato per voce. In tal caso occorre spostarlo manualmente per evitare collisioni.

```latex
\new Staff <<
  \new Voice \with {
    \consists "Ambitus_engraver"
  } \relative c'' {
    \override Ambitus.X-offset = #2.0
    \voiceOne
    c4 a d e
    f1
  }
  \new Voice \with {
    \consists "Ambitus_engraver"
  } \relative c' {
    \voiceTwo
    es4 f g as
    b1
  }
>>
```

Aggiungere un segno di ottava a una sola voce

Se il rigo ha più di una voce, l’ottavazione in una voce trasporrà la posizione delle note in tutte le voci per la durata della parentesi dell’ottava. Se si intende applicare l’ottavazione a una sola voce, occorre spostare l’incisore Ottava_spanner_engraver nel contesto Voice.

```latex
\layout {
  \context {
    \Staff {
      \remove Ottava_spanner_engraver
    }
  }
  \context {
    \Voice {
      \consists Ottava_spanner_engraver
    }
  }
}

\clef bass
<< { <g d'>1~ q2 <c' e'> } 
\`
```
Teste di nota Aiken variante sottile

Le teste di nota Aiken bianche sono più difficili da leggere a dimensioni del rigo più piccole, specialmente coi tagli addizionali. Perdendo lo spazio bianco interno le fa apparire come note di un quarto.

\score { 
  \aikenHeads
  \override Stem.details.beamed-lengths = #'(2)
  a8[ a] a16[ a] a32[ a]
  \override Stem.details.beamed-lengths = #'(8 10 12)
  a8[ a] a16[ a] a32[ a] r8
  \override Stem.details.beamed-lengths = #'(8)
  a8[ a]
\overridestemdetails.beamed-lengths = #'(8.5)
a8[ a]
\revertstemdetails.beamed-lengths
a8[ a] a16[ a] a32[ a] r16
}

Ambitus

Ambitus indicate pitch ranges for voices.

Accidentals only show up if they are not part of the key signature. AmbitusNoteHead grobs also have ledger lines.

\layout{
  \context{
    \Voice
    \consists "Ambitus_engraver"
  }
}

<<
\newStaff{
  \relative c' {
    \time 2/4
    c4 f'
  }
}
\newStaff{
  \relative c' {
    \time 2/4
    \key d \major
    cis4 as'
  }
}
>>

Ambitus dopo armatura di chiave

Per impostazione predefinita, gli ambitus sono posizionati a sinistra della chiave. La funzione \ambitusAfter permette di cambiare questo posizionamento. La sintassi è \ambitusAfter
grob-interface (vedi Graphical Object Interfaces (http://lilypond.org/doc/v2.22/Documentation/internals/graphical-object-interfaces) per un elenco dei possibili valori per grob-interface.)

Un caso d’uso comune è il posizionamento dell’ambitus tra l’armatura di chiave e l’indicazione di tempo.

\new Staff \with {
  \consists Ambitus_ engraver
} \relative {
  \ambitusAfter key-signature
  \key d \major
  es'8 g bes cis d2
}

\new Staff \with {
  \consists "Ambitus_ engraver"
}
<<
\new Voice \relative c' { 
  \voiceOne 
  c4 a d e 
  f1 
}
\new Voice \relative c' { 
  \voiceTwo 
  es4 f g as 
  b1 
}
>>

Applicazione degli stili delle teste di nota in base al grado della scala

La proprietà \shapeNoteStyles può essere usata per definire vari stili di teste di nota per ogni grado della scala (definita dall’armatura di chiave o dalla proprietà \tonic). Questa proprietà richiede un insieme di simboli, che può essere puramente arbitrario (sono permessi espressioni geometriche come triangle, cross e xcircle) o basato sull’antica tradizione tipografica americana (sono consentiti anche alcuni nomi di nota latini).

Detto questo, per imitare gli antichi canzonieri americani, ci sono vari stili predefiniti disponibili attraverso dei comodi comandi come \aikenHeads o \sacredHarpHeads.
Questo esempio mostra modi diversi di ottenere teste di nota di varie forme e illustra la possibilità di trasporre una melodia senza perdere la corrispondenza tra le funzioni armoniche e gli stili delle teste.

fragment = {
  \key c \major
  c2 d
e2 f
g2 a
  b2 c
}

\new Staff {
  \transpose c d
  \relative c' {
    \set shapeNoteStyles = ##(do re mi fa
      #f la ti)

    \fragment
  }

  \break

  \relative c' {
    \set shapeNoteStyles = ##(cross triangle fa #f
      mensural xcircle diamond)

    \fragment
  }
}

Cambiare automaticamente la direzione del gambo della nota centrale in base alla melodia

LilyPond può modificare la direzione del gambo della nota centrale di un rigo in modo che seguа la melodia: occorre aggiungere l’incisore Melody_engraver al contesto Voice.

La proprietà di contesto suspendMelodyDecisions può essere usata per disattivare questo comportamento localmente.

\relative c'' {
  \time 3/4
  a8 b g f b g |
  \set suspendMelodyDecisions = ##t
    a b g f b g |
  \unset suspendMelodyDecisions
    c b d c b c |}
Changing ottava text

Internally, \ottava sets the properties ottavation (for example, to 8va or 8vb) and middleCPosition. To override the text of the bracket, set ottavation after invoking \ottava.

Short text is especially useful when a brief ottava is used.

{  
  c'2
  \ottava #1
  \set Staff.ottavation = #'8"
  c''2
  \ottava #0
  c'1
  \ottava #1
  \set Staff.ottavation = #'Text"
  c''1
}

Modifica dell’intervallo dell’ambitus

È possibile cambiare le impostazioni predefinite dell’intervallo tra le teste di nota dell’ambitus e la linea che le collega.

\layout {  
  \context {  
    \Voice  
    \consists "Ambitus_engraver"
  }
}

\new Staff {  
  \time 2/4
  % Default setting
  c'4 g''
}
Changing the interval of lines on the stave

staffLineLayoutFunction is used to change the position of notes. This snippet shows setting its value to ly:pitch-semitones in order to produce a chromatic scale with the distance between each space and line of the stave equal to one semitone.

```
scale = \relative c' { 
a4 ais b c 
cis4 d dis e 
f4 fis g gis 
a1 }
```

```
\new Staff \with { 
  \remove "Accidental_engraver"
  staffLineLayoutFunction = #ly:pitch-semitones
}
```
Clefs can be transposed by arbitrary amounts

Clefs can be transposed by arbitrary amounts, not just by octaves.

Coloring notes depending on their pitch

It is possible to color note heads depending on their pitch and/or their names: the function used in this example even makes it possible to distinguish enharmonics.

%Association list of pitches to colors.
#(define color-mapping
(list
(cons (ly:make-pitch 0 0 NATURAL) (x11-color 'red))
(cons (ly:make-pitch 0 0 SHARP) (x11-color 'green))
(cons (ly:make-pitch 0 1 FLAT) (x11-color 'green))
(cons (ly:make-pitch 0 2 NATURAL) (x11-color 'red))
(cons (ly:make-pitch 0 2 SHARP) (x11-color 'green))
(cons (ly:make-pitch 0 3 FLAT) (x11-color 'red))
(cons (ly:make-pitch 0 3 NATURAL) (x11-color 'green))
(cons (ly:make-pitch 0 4 SHARP) (x11-color 'red))
(cons (ly:make-pitch 0 5 NATURAL) (x11-color 'green))
(cons (ly:make-pitch 0 5 FLAT) (x11-color 'red))
(cons (ly:make-pitch 0 6 SHARP) (x11-color 'red))
(cons (ly:make-pitch 0 1 NATURAL) (x11-color 'blue))
(cons (ly:make-pitch 0 3 SHARP) (x11-color 'blue)))
(cons (ly:make-pitch 0 4 FLAT) (x11-color 'blue))
(cons (ly:make-pitch 0 5 SHARP) (x11-color 'blue))
(cons (ly:make-pitch 0 6 FLAT) (x11-color 'blue)))

% Compare pitch and alteration (not octave).
#(define (pitch-equals? p1 p2)
  (and
   (= (ly:pitch-alteration p1) (ly:pitch-alteration p2))
   (= (ly:pitch-notename p1) (ly:pitch-notename p2))))

#(define (pitch-to-color pitch)
  (let ((color (assoc pitch color-mapping pitch-equals?)))
    (if color
      (cdr color))))

#(define (color-notehead grob)
  (pitch-to-color
   (ly:event-property (event-cause grob) 'pitch)))

\score {
  \new Staff \relative c' { 
    \override NoteHead.color = #color-notehead
    c8 b d dis ees f g aes
  }
}

Creating a sequence of notes on various pitches

In music that contains many occurrences of the same sequence of notes at different pitches, the following music function may prove useful. It takes a note, of which only the pitch is used.

This example creates the rhythm used throughout Mars, from Gustav Holst’s The Planets.

rhythm =
#(define-music-function (p) (ly:pitch?)
  "Make the rhythm in Mars (the Planets) at the given pitch"
  #{ \tuplet 3/2 { $p 8 8 8 } 4 4 8 8 4 #})

\new Staff {
  \time 5/4 
  \rhythm c'
  \rhythm c''
  \rhythm g
}
Creating custom key signatures

LilyPond supports custom key signatures. In this example, print for D minor with an extended range of printed flats.

\new Staff \with {
  \override StaffSymbol.line-count = #8
  \override KeySignature.flat-positions = #'((-7 . 6))
  \override KeyCancellation.flat-positions = #'((-7 . 6))
  \override KeySignature.sharp-positions = #'((-6 . 7))
  \override KeyCancellation.sharp-positions = #'((-6 . 7))

  \override Clef.stencil = #
  \(\text{\texttt{\{\lambda grob\{}}\texttt{\grob-interpret-markup grob}}\)
  \#\{ \markup\combine
    \musicglyph "clefs.C"
    \translate #'((-3 . -2)
    \musicglyph "clefs.F"
  \})
  \clefPosition = #3
  \middleCPosition = #3
  \middleCClefPosition = #3
}

\key d\minor
f bes, f bes,

Direction of merged ‘fa’ shape note heads

Using property NoteCollision.fa-merge-direction, the direction of ‘fa’ shape note heads (‘fa’, ‘faThin’, etc.) can be controlled independently of the stem direction if two voices with the same pitch and different stem directions are merged. If this property is not set, the ‘down’ glyph variant is used.

\clef bass
<< \aikenHeads
  f2
  \override Staff.NoteCollision.fa-merge-direction = #UP
  f2 \}
\\ \{ \aikenHeads
  f2
  f2 \}
>>
Force a cancellation natural before accidentals

The following example shows how to force a natural sign before an accidental.

```latex
\relative c' { 
  \key es \major 
  bes c des 
  \tweak Accidental.restore-first \#t 
  eis 
}
```

Forcing a clef symbol to be displayed

When a clef sign has already been displayed and it has not been changed to a different clef, then repeating the `\clef` command will be ignored by LilyPond, since it is not a change of clef. It is possible to force the clef to be redisplayed using the command `\set Staff.forceClef = \#t`.

```latex
\relative c' { 
  \clef treble 
  c1 
  \clef treble 
  c1 
  \set Staff.forceClef = \#t 
  c1 
  \clef treble 
  c1 
}
```

Generating random notes

This Scheme-based snippet generates 24 random notes (or as many as required), based on the current time (or any randomish number specified instead, in order to obtain the same random notes each time): i.e., to get different random note patterns, just change this number.

```scheme
\score { 
  $(let ((random-state (seed->random-state (current-time)))))
    (make-sequential-music
      (map (lambda (x)
            (let ((idx (random 12 random-state)))
              (make-event-chord
                (list
                  (make-music 'NoteEvent
```
Nascondere le alterazioni delle note con legatura di valore all'inizio di un nuovo sistema

Questo frammento mostra come nascondere le alterazioni delle note unite alla figura precedente mediante una legatura di valore all'inizio di un nuovo sistema
\relative c'' { 
  \override Accidental.hide-tied-accidental-after-break = ##t 
  cis1~ cis~ 
  \break 
  cis 
}

Keep change clefs full sized

When a clef is changed, the clef sign displayed is smaller than the initial clef. This can be overridden with full-size-change.
\relative c' { 
  \clef "treble" 
  c1 
  \clef "bass" 
  c1 
  \clef "treble" 
  c1 
  \override Staff.Clef.full-size-change = ##t 
  \clef "bass" 
  c1 
  \clef "treble" 
  c1 
  \revert Staff.Clef.full-size-change 
  \clef "bass"
Esempio di makam

Makam è un tipo di melodia proveniente dalla Turchia che usa alterazioni microtonali di 1/9. Consultare il file di inizializzazione ‘ly/makam.ly’ per vedere come sono definiti i nomi delle altezze e le alterazioni.

\% Initialize makam settings
\include "makam.ly"

\relative c' {
  \set Staff.keyAlterations = #'((6 . ,(- KOMA)) (3 . ,BAKIYE))
c4 cc db fk
gbm4 gfc gfb efk
fk4 db cc c
}

Modificare l’inclinazione dell’estensore dell’ottava

È possibile cambiare l’inclinazione dell’estensore dell’ottava.

\relative c'' {
  \override Staff.OttavaBracket.stencil = #ly:line-spanner::print
  \override Staff.OttavaBracket.bound-details = 
    #'((left . ((Y . 0)
      (attach-dir . ,LEFT)
      (padding . 0)
      (stencil-align-dir-y . ,CENTER))))
    (right . ((Y . 5.0) ; Change the number here
      (padding . 0)
      (attach-dir . ,RIGHT)
      (text . , (make-draw-dashed-line-markup
        (cons 0 -1.2)))))
  \override Staff.OttavaBracket.left-bound-info =
    #ly:horizontal-line-spanner::calc-left-bound-info-and-text
  \override Staff.OttavaBracket.right-bound-info =
    #ly:horizontal-line-spanner::calc-right-bound-info
  \ottava #1
  c1
  c''''1
}
Armature di chiave non tradizionali

Il comando \key comunemente usato imposta la proprietà keyAlterations del contesto Staff. Per creare armature di chiave non standard, tale proprietà va impostata esplicitamente.

Il formato di questo comando è una lista:

\set Staff.keyAlterations = #*(((ottava . grado) . alterazione))

dove, per ogni elemento della lista, ottava indica l’ottava (0 è l’ottava dal Do centrale al Si precedente), grado indica la nota all’interno dell’ottava (0 significa Do e 6 significa Si) e alterazione può essere ,SHARP ,FLAT ,DOUBLE-SHARP etc.

Altrimenti, usando il formato breve per ogni elemento della lista, (grado . alterazione), ciò indica che la stessa alterazione deve essere presente in tutte le ottave. Per le scale microtonalidove un “diesis” non è 100 centesimi, alterazione si riferisce alla proporzione di un duecentesimo di tono intero.

\include "arabic.ly"
\relative do' {
  \set Staff.keyAlterations = #*((0 . ,SEMI-FLAT)
                             (1 . ,SEMI-FLAT)
                             (2 . ,FLAT)
                             (5 . ,FLAT)
                             (6 . ,SEMI-FLAT))
%
\set Staff.extraNatural = ##f
  re reb \dwn reb resd
  dod dob dosd \dwn dob |
  dobsd sb dob do do |
}

Numeri dentro le teste di nota

Le teste di nota con nome della nota usano la proprietà note-names dell’oggetto NoteHead per determinare cosa appaia all’interno della testa. È possibile sovrascrivere questa proprietà e mostrare numeri corrispondenti ai gradi della scala.

Si può creare un semplice incisore che faccia questo per ogni oggetto testa di nota che incontra.

#(define Ez_numbers_engraver
  (make-engraver
    (acknowledgers
      ((note-head-interface engraver grob source-engraver)
        (let* ((context ((ly:translator-context engraver))
                    (tonic-pitch (ly:context-property context 'tonic))
                    (tonic-name (ly:pitch-notename tonic-pitch)))
          (note-head-interface engraver grob source-engraver))
    (context ((ly:translator-context engraver))
      (ly:context-property context 'tonic))
    (ly:pitch-notename tonic-pitch))
  )
)
(grob-pitch
  (ly:event-property (event-cause grob) 'pitch))
(grob-name (ly:pitch-notename grob-pitch))
(delta (modulo (- grob-name tonic-name) 7))
(note-names
  (make-vector 7 (number->string (1+ delta)))))

(ly:grob-set-property! grob 'note-names note-names))))))

#(set-global-staff-size 26)

\layout {
  \ragged-right = ##t
  \context {
    \Voice
      \consists \Ez_numbers_engraver
  }
}

\relative c' {
  \easyHeadsOn
  c4 d e f
  g4 a b c \break

  \key a \major
  a,4 b cis d
  e4 fis gis a \break

  \key d \dorian
  d,4 e f g
  a4 b c d
}

\timesig 4/4
\clef bass
\timesig 3/4
\clef bass
\timesig 5/4
\clef bass
Modello per orchestra, coro e pianoforte

Questo modello mostra come usare i contesti annidati StaffGroup e GrandStaff per creare sottogruppi degli strumenti dello stesso tipo. Mostra anche come usare \transpose in modo che le variabili mantengano la musica per gli strumenti traspositori nell’intonazione reale.

\(\texttt{#(set-global-staff-size 17)}\)
\(\texttt{paper}\)
\(\quad \texttt{indent} = 3.0\ cm \ % \ add \ space \ for \ instrumentName\)
\(\quad \texttt{short-indent} = 1.5\ cm \ % \ add \ less \ space \ for \ shortInstrumentName\)

\(\texttt{fluteMusic} = \texttt{relative c'} \ { \texttt{key g \ major g'1 b} }\)

\(\% \ \text{Pitches as written on a manuscript for Clarinet in A}\)
\(\% \ \text{are transposed to concert pitch.}\)

\(\texttt{clarinetMusic} = \texttt{\transpose c' a}\)
\(\quad \texttt{relative c''} \ { \texttt{key bes \ major bes1 d} }\)

\(\texttt{trumpetMusic} = \texttt{relative c} \ { \texttt{key g \ major g'1 b} }\)

\(\% \ \text{Key signature is often omitted for horns}\)

\(\texttt{hornMusic} = \texttt{\transpose c' f}\)
\(\quad \texttt{relative c} \ { \texttt{d'1 fis} }\)

\(\texttt{percussionMusic} = \texttt{relative c} \ { \texttt{key g \ major g1 b} }\)

\(\texttt{sopranoMusic} = \texttt{relative c'} \ { \texttt{key g \ major g'1 b} }\)

\(\texttt{sopranoLyrics} = \texttt{\lyricmode} \ { \texttt{Lyr -- ics} }\)

\(\texttt{altoIMusic} = \texttt{relative c'} \ { \texttt{key g \ major g'1 b} }\)

\(\texttt{altoIIMusic} = \texttt{relative c'} \ { \texttt{key g \ major g'1 b} }\)

\(\texttt{altoILyrics} = \texttt{\sopranoLyrics}\)

\(\texttt{altoIILyrics} = \texttt{\lyricmode} \ { \texttt{Ah -- ah} }\)

\(\texttt{tenorMusic} = \texttt{relative c'} \ { \texttt{clef "treble_8" key g \ major g1 b} }\)

\(\texttt{tenorLyrics} = \texttt{\sopranoLyrics}\)

\(\texttt{pianoRHMusic} = \texttt{relative c} \ { \texttt{key g \ major g'1 b} }\)

\(\texttt{pianoLHMusic} = \texttt{relative c} \ { \texttt{clef bass key g \ major g1 b} }\)

\(\texttt{violinIMusic} = \texttt{relative c'} \ { \texttt{key g \ major g'1 b} }\)

\(\texttt{violinIIMusic} = \texttt{relative c'} \ { \texttt{key g \ major g'1 b} }\)
violaMusic = relative c { clef alto \key g \major g'1 b }

celloMusic = relative c { clef bass \key g \major g1 b }

bassMusic = relative c { clef "bass_8" \key g \major g,1 b }

\score { 
  \new StaffGroup = "StaffGroup_woodwinds" <<
  \new Staff = "Staff_flute" \with { instrumentName = "Flute" }
  \fluteMusic

  \new Staff = "Staff clarinet" \with {
    instrumentName = \markup { \concat { "Clarinet in B" \flat } }
  }

  % Declare that written Middle C in the music
  % to follow sounds a concert B flat, for
  % output using sounded pitches such as MIDI.
  %\transposition bes

  % Print music for a B-flat clarinet
  \transpose bes c' \clarinetMusic
>

\new StaffGroup = "StaffGroup_brass" <<
  \new Staff = "Staff_hornI" \with { instrumentName = "Horn in F" }
  % \transposition f
  \transpose f c' \hornMusic

  \new Staff = "Staff_trumpet" \with { instrumentName = "Trumpet in C" }
  \trumpetMusic
>

\new RhythmicStaff = "RhythmicStaff_percussion"
  \with { instrumentName = "Percussion" }
<<
  \percussionMusic
>

\new PianoStaff \with { instrumentName = "Piano" }
<<
  \new Staff { \pianoRHMusic }
  \new Staff { \pianoLHMusic }
>>

\new ChoirStaff = "ChoirStaff_choir" <<
  \new Staff = "Staff_soprano" \with { instrumentName = "Soprano" }
  \new Voice = "soprano"
  \sopranoMusic

  \new Lyrics \lyricsto "soprano" { \sopranoLyrics }
  \new GrandStaff = "GrandStaff_altos"
  \with { \accepts Lyrics } <<
\new Staff = "Staff_altoI" \with { instrumentName = "Alto I" }
\new Voice = "altoI"
\altoIMusic

\new Lyrics \lyricsto "altoI" { \altoILyrics }
\new Staff = "Staff_altoII" \with { instrumentName = "Alto II" }
\new Voice = "altoII"
\altoIIIMusic

\new Lyrics \lyricsto "altoII" { \altoIIILyrics }

\new Staff = "Staff_tenor" \with { instrumentName = "Tenor" }
\new Voice = "tenor"
\tenorMusic

\new Lyrics \lyricsto "tenor" { \tenorLyrics }

\new StaffGroup = "StaffGroup_strings" <<
\new GrandStaff = "GrandStaff_violins" <<
\new Staff = "Staff_violinI" \with { instrumentName = "Violin I" }
\violiniMusic

\new Staff = "Staff_violinII" \with { instrumentName = "Violin II" }
\violiniIMusic

\new Staff = "Staff_viola" \with { instrumentName = "Viola" }
\violaMusic

\new Staff = "Staff_cello" \with { instrumentName = "Cello" }
\celloMusic

\new Staff = "Staff_bass" \with { instrumentName = "Double Bass" }
\bassMusic

>>

\layout { }
Impedire l’inserimento automatico dei bequadri supplementari

Secondo le norme tipografiche tradizionali, un segno di bequadro viene inserito prima di un diesis o di un bemolle se un precedente doppio diesis o bemolle sulla stessa nota è cancellato. Per cambiare questo comportamento e seguire la pratica contemporanea, si imposta la proprietà extraNatural su f (falso) nel contesto Staff.

\relative c' {  
  a|es|es|a|es|a  
  \set Staff.extraNatural = ##f  
  a|es|es|a|es|a  
}
Impedire l’inserimento dei segni di bequadro quando cambia l’armatura di chiave

Quando l’armatura di chiave cambia, vengono inseriti automaticamente i segni di bequadro per annullare le alterazioni di precedenti armature. Si può evitare questo comportamento impostando su \f (falso) la proprietà printKeyCancellation nel contesto Staff.

\relative c' {
\key d \major
a4 b cis d
\key g \minor
a4 bes c d
\set Staff.printKeyCancellation = ##f
\key d \major
a4 b cis d
\key g \minor
a4 bes c d
}

Quoting another voice with transposition

Quotations take into account the transposition of both source and target. In this example, all instruments play sounding middle C; the target is an instrument in F. The target part may be transposed using \transpose. In this case, all the pitches (including the quoted ones) are transposed.

\addQuote clarinet {
  \transposition bes
  \repeat unfold 8 { d'16 d' d'8 }
}

\addQuote sax {
  \transposition es'
  \repeat unfold 16 { a8 }
}

quoteTest = {
  % french horn
  \transposition f
  g'4
  << \quoteDuring "clarinet" { \skip 4 } s4~"clar." >>
  << \quoteDuring "sax" { \skip 4 } s4~"sax." >>
  g'4
}

{
  \new Staff \with {
    instrumentName = \markup { \column { Horn "in F" } }
  }
  \quoteTest
  \transpose c' d' << \quoteTest s4~"up a tone" >>
Separating key cancellations from key signature changes

By default, the accidentals used for key cancellations are placed adjacent to those for key signature changes. This behavior can be changed by overriding the 'break-align-orders property of the BreakAlignment grob.

The value of 'break-align-orders is a vector of length 3, with quoted lists of breakable items as elements. This example only modifies the second list, moving key-cancellation before staff-bar; by modifying the second list, break alignment behavior only changes in the middle of a system, not at the beginning or the end.

```latex
\new Staff {
  \override Score.BreakAlignment.break-align-orders =
  ##((left-edge ambitus breathing-sign clef staff-bar
    key-cancellation key-signature time-signature custos)

  (left-edge ambitus breathing-sign clef key-cancellation
    staff-bar key-signature time-signature custos)

  (left-edge ambitus breathing-sign clef key-cancellation
    key-signature staff-bar time-signature custos))

\key des \major
\c'1
\bar "||"
\key bes \major
\c'1
}
```

Trasposizione delle altezze con numero minimo di alterazioni

Questo esempio usa del codice Scheme per imporre delle modifiche enarmoniche alle note che permettano di avere il numero minimo di alterazioni. In questo caso si applica la seguente regola:

- Le doppie alterazioni devono essere eliminate
- Si diesis -> Do
- Mi diesis -> Fa
- Do bemolle -> Si
- Fa bemolle -> Mi

In questo modo vengono scelti i suoni enarmonici più semplici.

```scheme
#:define (naturalize-pitch p)
  (let ((o (ly:pitch-octave p))
        (a (* 4 (ly:pitch-alteration p))))
```
Pitches

;;; alteration, a, in quarter tone steps,
;;; for historical reasons
(n (ly:pitch-notename p))

(cond
  ((and (> a 1) (or (eqv? n 6) (eqv? n 2)))
   (set! a (- a 2))
   (set! n (+ n 1)))
  ((and (< a -1) (or (eqv? n 0) (eqv? n 3)))
   (set! a (+ a 2))
   (set! n (- n 1))))

(cond
  ((> a 2) (set! a (- a 4)) (set! n (+ n 1)))
  ((< a -2) (set! a (+ a 4)) (set! n (- n 1)))))

(if (< n 0) (begin (set! o (- o 1)) (set! n (+ n 7)))
  (if (> n 6) (begin (set! o (+ o 1)) (set! n (- n 7)))
    (ly:make-pitch o n (/ a 4))))

#(define (naturalize music)
  (let ((es (ly:music-property music 'elements))
         (e (ly:music-property music 'element))
         (p (ly:music-property music 'pitch)))
    (if (pair? es)
        (ly:music-set-property! music 'elements
          (map naturalize es)))
    (if (ly:music? e)
        (ly:music-set-property! music 'element
          (naturalize e)))
    (if (ly:pitch? p)
        (begin
          (set! p (naturalize-pitch p))
          (ly:music-set-property! music 'pitch p))
        music)))

naturalizeMusic =
#(define-music-function (m)
  (ly:音乐 m)
  (naturalize m))

music = \relative c' { c4 d e g }

\score {
  \new Staff {
    \transpose c ais { \music }
    \naturalizeMusic \transpose c ais { \music }
    \transpose c deses { \music }
    \naturalizeMusic \transpose c deses { \music }
  }
  \layout { }
}
Turkish Makam example

This template uses the start of a well-known Turkish Saz Semai that is familiar in the repertoire in order to illustrate some of the elements of Turkish music notation.

\paper {\tagline = ##f}

% Initialize makam settings
\include "turkish-makam.ly"

\header {
  \title = "Hüseyni Saz Semaisi"
  \composer = "Lavtacı Andon"
}

\relative {
  \set Staff.extraNatural = ##f
  \set Staff.autoBeaming = ##f

  \key a \huseyni
  \time 10/8

  a'4 g'16 [fb] e8. [d16] d [c d e] c [d c8] bfc |
  a16 [bfc a8] bfc c16 [d c8] d16 [e d8] e4 fb8 |
  d4 a'8 a16 [g fb e] fb8 [g] a8. [b16] a16 [g] |
  g4 g16 [fb] fb8. [e16] e [g fb e] e4 r8 |
}

Hüseyni Saz Semaisi

Lavtacı Andon

Modifiche manuali della proprietà della chiave

Cambiando il glifo della chiave, la sua posizione o l’ottavazione non cambia la posizione delle note successive nel rigo. Per far sì che le armature di chiave si trovino sulle linee del rigo corrette, bisogna specificare anche middleCPosition, con valori positivi o negativi che spostano il Do centrale rispettivamente su o giù in senso relativo alla linea centrale del rigo.

Per esempio, \clef "treble_8" equivale a impostare clefGlyph, clefPosition (che regola la posizione verticale della chiave), middleCPosition e clefTransposition. Viene stampata una chiave quando cambia una di queste proprietà, eccetto middleCPosition.
Gli esempi seguenti mostrano le possibilità date dall’impostazione manuale di tali proprietà. Sulla prima linea le modifiche manuali preservano il posizionamento relativo standard di chiavi e note, mentre sulla seconda linea non lo fanno.

{%
% The default treble clef
\key f \major
c'1
%
% The standard bass clef
\set Staff.clefGlyph = #"clefs.F"
\set Staff.clefPosition = #2
\set Staff.middleCPosition = #6
\set Staff.middleCClefPosition = #6
\key g \major
c'1
%
% The baritone clef
\set Staff.clefGlyph = #"clefs.C"
\set Staff.clefPosition = #4
\set Staff.middleCPosition = #4
\set Staff.middleCClefPosition = #4
\key f \major
c'1
%
% The standard choral tenor clef
\set Staff.clefGlyph = #"clefs.G"
\set Staff.clefPosition = #-2
\set Staff.clefTransposition = #-7
\set Staff.middleCPosition = #1
\set Staff.middleCClefPosition = #1
\key f \major
c'1
%
% A non-standard clef
\set Staff.clefPosition = #0
\set Staff.clefTransposition = #0
\set Staff.middleCPosition = #-4
\set Staff.middleCClefPosition = #-4
\key g \major
c'1 \break
%
% The following clef changes do not preserve
% the normal relationship between notes, key signatures
% and clefs:
%
\set Staff.clefGlyph = #"clefs.F"
\set Staff.clefPosition = #2
c'1
\set Staff.clefGlyph = #"clefs.G"
c'1
\set Staff.clefGlyph = #"clefs.C"
c'1
\set Staff.clefTransposition = #7
c'1
\set Staff.clefTransposition = #0
Using autochange with more than one voice

Using autochange with more than one voice.

```latex
\score
{\new PianoStaff
  \new Staff = "up" {
    \new Staff = "up" {
      \set Staff.clefPosition = #0
c'1
      \set Staff.middleCPosition = #0
c'1
      \set Timing.beamExceptions = #'()
      \set Timing.beatStructure = #'(4)
      \new Voice {
        \voiceOne
        \autoChange
        \relative c' {
          g8 a b c d e f g
          g,,8 a b c d e f g
        }
      }
      \new Voice {
        \voiceTwo
        \autoChange
        \relative c' {
          g8 a b c d e f g
          g,,8 a b c d e f g
        }
      }
    }
  }
  \new Staff = "down" {
    \clef bass
  }
}
```
Rhythms

Sezione “Rhythms” in Guida alla Notazione

Adding beams, slurs, ties etc. when using tuplet and non-tuplet rhythms

LilyPond syntax can involve many unusual placements for parentheses, brackets etc., which might sometimes have to be interleaved.

For example, when entering a manual beam, the left square bracket has to be placed after the starting note and its duration, not before. Similarly, the right square bracket should directly follow the note which is to be at the end of the requested beaming, even if this note happens to be inside a tuplet section.

This snippet demonstrates how to combine manual beaming, manual slurs, ties and phrasing slurs with tuplet sections (enclosed within curly braces).

```lilypond
\time 2/4
\tuplet 5/4 { e'32\( a b d' e' \) a'4.\})
```

Adding drum parts

Using the powerful pre-configured tools such as the \drummode function and the DrumStaff context, inputting drum parts is quite easy: drums are placed at their own staff positions (with a special clef symbol) and have note heads according to the drum. Attaching an extra symbol to the drum or restricting the number of lines is possible.

```lilypond
drh = \drummode {
cymc4."crash" hhc16"h.h." hh hhc8 hho hhc8 hh16 hh
hhc4 r4 r2
}
drl = \drummode {
bd4 sn8 bd bd4 << bd ss >>
bd8 tommh tommh bd toml toml bd tomfh16 tomfh
}
timb = \drummode {
timb4 ssh timl8 ssh r timh r4
ssh8 timl r4 cb8 cb
}
\score {
<<
  \new DrumStaff \with {
    instrumentName = "timbales"
  }
}
```
### Adjusting grace note spacing

The space given to grace notes can be adjusted using the `spacing-increment` property of `Score.GraceSpacing`.

```latex
\begin{verbatim}
graceNotes = {
  \grace { c4 c8 c16 c32 }
  c8
}
\relative c'' {
  c8
  \graceNotes
  \override Score.GraceSpacing.spacing-increment = #2.0
  \graceNotes
  \revert Score.GraceSpacing.spacing-increment
  \graceNotes
}
\end{verbatim}
```

### Allineare i numeri di battuta

Per impostazione predefinita i numeri di battuta sono allineati a destra rispetto al loro oggetto genitore. Di solito si tratta del margine sinistro della linea oppure, se i numeri appaiono all’interno della linea, del lato sinistro della stanghetta. I numeri possono essere posizionati anche direttamente sopra la stanghetta oppure allineati a sinistra della stanghetta.
Rhythms

\relative c' { 
\set Score.currentBarNumber = #111 
\override Score.BarNumber.break-visibility = #all-visible 
% Increase the size of the bar number by 2 
\override Score.BarNumber.font-size = #2 
% Print a bar number every second measure 
\set Score.barNumberVisibility = #(every-nth-bar-number-visible 2) 
c1 | c1 
% Center-align bar numbers 
\override Score.BarNumber.self-alignment-X = #CENTER 
c1 | c1 
% Left-align bar numbers 
\override Score.BarNumber.self-alignment-X = #LEFT 
c1 | c1 
}

Note brevi alternative

Le note brevi sono disponibili anche con due linee verticali su ciascun lato della testa invece di una sola e in stile barocco.

\relative c'' { 
\time 4/2 
c\breve | 
\override Staff.NoteHead.style = #'altdefault 
b\breve 
\override Staff.NoteHead.style = #'baroque 
b\breve 
\revert Staff.NoteHead.style 
a\breve 
}

Appoggiatura or grace note before a bar line

By default, appoggiaturas and grace notes that occur on the first beat of a measure are printed after the bar line. They can however be printed before, simply by adding an invisible BarLine and then the visible one, as demonstrated here.

{ 
R1 
%% default 
\appoggiatura d''8 c''4 r2. 
%% cheated 
\appoggiatura { \bar "" d''8 \bar "|" } c''4 r2. 
}
Automatic beam subdivisions

Beams can be subdivided automatically. By setting the property `subdivideBeams`, beams are subdivided at beat positions (as specified in `baseMoment`).

```latex
\new Staff {
  \relative c'' {
    <<
    \voiceOne
    \set subdivideBeams = ##t
    b32[ a g f c' b a g
    b32"subdivide beams" a g f c' b a g]
    }\new Voice {
    \voiceTwo
    b32."default"[ a g f c' b a g
    b32 a g f c' b a g]
    }>
  \oneVoice
  \set baseMoment = #(ly:make-moment 1/8)
  \set beatStructure = 2,2,2,2
  b32"baseMoment 1 8"[ a g f c' b a g]
  \set baseMoment = #(ly:make-moment 1/16)
  \set beatStructure = 4,4,4,4
  b32"baseMoment 1 16"[ a g f c' b a g]
}
```

Automatically change durations

`shiftDurations` can be used to change the note lengths of a piece of music.

It takes two arguments - the scaling factor as a power of two, and the number of dots to be added as a positive integer.

```latex
\paper { indent = 0 }

\music = \relative c'' { a1 b2 c4 d8 r }
\score {
  \new Voice {
    \time 4/2
```
Estremità delle travature nel contesto Score

Le regole relative alle estremità delle travature definite nel contesto Score si applicano a tutti i righi, ma possono essere modificate anche ai livelli Staff e Voice:

```plaintext
\relative c'' {  
  \time 5/4  
  \set Score.baseMoment = #(ly:make-moment 1/8)  
  \set Score.beatStructure = 3,4,3  
  \new Staff {  
    c8 c c c c c c c c  
  }  
  \new Staff {  
    \set Staff.beatStructure = 6,4  
    c8 c c c c c c c  
  }  
  \new Staff {  
    % Inherit beaming from Score context  
    \voiceOne  
  }  
}  
```
Travature che attraversano le interruzioni di linea

Le interruzioni di linea sono di norma proibite quando le travature attraversano la stanghetta di una battuta. Si può cambiare questo comportamento nel modo seguente:

```latex
\relative c'' {
\override Beam.breakable = ##t
\new Voice {
\voiceTwo
\set Voice.beatStructure = 6,4
a8 a a a a a a a a a a
}
}\noteheads.s2/noteheads.s2
\timesig.C44/clefs.G/noteheads.s2/noteheads.s2/noteheads.s2/noteheads.s2/noteheads.s2/noteheads.s2/noteheads.s2/noteheads.s2
\flags.d3\timesig.C44/clefs.G/noteheads.s2/noteheads.s2/noteheads.s2/noteheads.s2/noteheads.s2/noteheads.s2/noteheads.s2/noteheads.s2
\flags.d3\timesig.C44/clefs.G
```

Modificare la distanza delle travature angolari


```latex
\new Voice {
\override Beam.auto-knee-gap = #6
f8 f''8 f8 f''8
}```
f8 f''8 f8 f''8

}\}

Modificare la forma delle pause multiple

Se la pausa multipla dura dieci misure o un numero inferiore a dieci, nel rigo apparirà una serie di pause di longa e di breve (chiamate in tedesco “Kirchenpausen” - pause ecclesiastiche); altrimenti apparirà una semplice linea. Il numero predefinito di dieci può essere cambiato sovrascrivendo la proprietà `expand-limit`..

```
relative c' \compressMMRests { R1*2 | R1*5 | R1*9 }
\override MultiMeasureRest.expand-limit = #3 R1*2 | R1*5 | R1*9 }
```

Modifica del numero di punti di aumentazione per nota

Il numero di punti di aumentazione su una singola nota può essere modificato in modo indipendente dai punti posizionati dopo la nota.

```
relative c' { c4.. a16 r2 | \override Dots.dot-count = #4 c4.. a16 r2 | \override Dots.dot-count = #0 c4.. a16 r2 | \revert Dots.dot-count c4.. a16 r2 | }
```

Modificare il tempo senza mostrare l’indicazione metronomica

Per cambiare il tempo del file MIDI senza che appaia l’indicazione metronomica, basta renderla invisibile.

```
\score { \new Staff \relative c' { \tempo 4 = 160
```
c4 e g b
c4 b d c
\set Score.tempoHideNote = ##t
\tempo 4 = 96
d,4 fis a cis
d4 cis e d
}
\layout { }
\midi { }

\relative c'' {
\tuplet 3/2 { c8 c c }
\tuplet 3/2 { c8 c c }
\override TupletNumber.text = #tuplet-number::calc-fraction-text
\tuplet 3/2 { c8 c c }
\omit TupletNumber
\tuplet 3/2 { c8 c c }
}

Changing time signatures inside a polymetric section using \scaleDurations

The measureLength property, together with measurePosition, determines when a bar line is needed. However, when using \scaleDurations, the scaling of durations makes it difficult to change time signatures. In this case, measureLength should be set manually, using the \ly:make-moment callback. The second argument must be the same as the second argument of \scaleDurations.

\layout {
\context {
\Score
\remove "Timing_translator"
}
\context {
\Staff
\consists "Timing_translator"
}
Notazione per canti e salmi

Questa forma di notazione è utilizzata per i salmi, dove i versi non sono sempre della stessa lunghezza.

stemOff = \hide Staff.Stem
stemOn = \undo \stemOff

\score {
\new Staff \with { \remove "Time_signature_engraver" }
{
\key g \minor
\cadenzaOn
\stemOff a'\breve bes'4 g'4
\stemOn a'2 \section
\stemOff a'\breve g'4 a'4
\stemOn f'2 \section
\stemOff a'\breve\markup { \italic flexe }
\stemOn g'2 \fine
}
}
Compound time signatures

Odd 20th century time signatures (such as “5/8”) can often be played as compound time signatures (e.g. “3/8 + 2/8”), which combine two or more inequal metrics.

LilyPond can make such music quite easy to read and play, by explicitly printing the compound time signatures and adapting the automatic beaming behavior.

\relative c' {
  \compoundMeter #'((2 8) (3 8))
  c8 d e fis gis
  c8 fis, gis e d
  c8 d e4 gis8
}

Segni per la conduzione, segni di raggruppamento della misura

Il raggruppamento delle pulsazioni all’interno della misura è regolato dalla proprietà di contesto beatStructure. I valori di beatStructure per varie indicazioni di tempo vengono stabiliti in scm/time-signature-settings.scm. Questi valori possono essere impostati o modificati con \set. Altrimenti, si può usare \time per impostare sia l’indicazione di tempo che la struttura delle pulsazioni. Per farlo si specifica il raggruppamento interno delle pulsazioni in una misura in una lista di numeri (nella sintassi di Scheme) prima dell’indicazione di tempo.

\time agisce nel contesto Timing, dunque non reimposterà i i valori di beatStructure e baseMoment che sono impostati in altri contesti di più basso livello, come Voice.

Se si include l’incisore Measure_grouping_engraver in uno dei contesti che regolano l’aspetto, appariranno i segni di raggruppamento della misura. Tali segni facilitano la lettura di musica moderna ritmicamente complessa. Nell’esempio la misura di 9/8 è raggruppata in due diversi schemi usando due metodi differenti, mentre la misura di 5/8 è raggruppata in base alle impostazioni predefinite in scm/time-signature-settings.scm:

\score {  
  \new Voice \relative c'' {  
     \time 9/8
     g8 g d d g g a( bes g) |
     \set Timing.beatStructure = 2,2,2,3
     g8 g d d g g a( bes g) |
     \time 4,5 9/8
     g8 g d d g g a( bes g) |
     \time 5/8
     a4. g4 |
  }
  \layout {
    \context {  
      \Staff
      \consists "Measure_grouping_engraver"
    }
  }
}
Consistently left aligned bar numbers

When left aligning bar numbers, overlapping problems may occur with Staves brackets.

The snippet solves this by keeping right aligned the first bar number following line breaks.

```latex
\texttt{consistentlyLeftAlignedBarNumbers} = {
  \texttt{\override Score.BarNumber.break-visibility = #end-of-line-invisible}
  \texttt{\override Score.BarNumber.self-alignment-X = #(lambda (grob)
    (let ((break-dir (ly:item-break-dir grob)))
      (if (= break-dir RIGHT) RIGHT LEFT)))}
}

\texttt{new ChoirStaff} \<-
\texttt{new Staff} {
  \texttt{relative c'} {
    \texttt{set Score.barNumberVisibility = #(every-nth-bar-number-visible 3)}
    \texttt{bar ""}
    \texttt{consistentlyLeftAlignedBarNumbers}
    \texttt{set Score.currentBarNumber = #112}
    \texttt{repeat unfold 8 \{ R1 \}}
    \texttt{break}
    \texttt{repeat unfold 9 \{ R1 \}}
    \texttt{break}
    \texttt{repeat unfold 7 \{ R1 \}}
  }
}
\texttt{new Staff} {
  \texttt{relative c'} {
    \texttt{repeat unfold 24 \{ R1 \}}
  }
}
\texttt{>>}

\texttt{\layout {}
  \texttt{indent = #0}
  \texttt{ragged-right = ##t}
  \texttt{ragged-last = ##t}
\texttt{}}
```
Controllare la visibilità della parentesi del gruppo irregolare

Il comportamento predefinito relativo alla visibilità della parentesi quadra del gruppo irregolare è di mostrare una parentesi a meno che non ci sia una travatura della stessa lunghezza del gruppo.

Per controllare la visibilità di tale parentesi, si imposta la proprietà ‘bracket-visibility’ su #t (mostra sempre la parentesi), #’if-no-beam (mostra la parentesi solo se non c’è una travatura, che è il comportamento predefinito) o #f (non mostrare mai la parentesi). L’ultima opzione equivale a omettere l’oggetto @code{TupletBracket} dall’output.

```latex
\relative c' {
    \tuplet 3/2 { c16[ d e ] f8}
    \tuplet 3/2 { c8 d e }
    \tuplet 3/2 { c4 d e }
}
```

```latex
\new Voice {
\relative c' {
    \override Score.TextMark.non-musical = ##f
    \textMark "default" \music
    \override TupletBracket.bracket-visibility = #'if-no-beam
    \textMark \markup \typewriter "'if-no-beam" \music
    \override TupletBracket.bracket-visibility = ##t
    \textMark \markup \typewriter "#t" \music
    \override TupletBracket.bracket-visibility = ##f
    \textMark \markup \typewriter "#f" \music
    \omit TupletBracket
    \textMark \markup \typewriter "omit" \music
}
Cow and ride bell example

Two different bells, entered with 'cb' (cowbell) and 'rb' (ridebell).

```latex
\paper { tagline = ##f }
\new DrumStaff \with { instrumentName = "Different Bells" }

\drummode {
  \set DrumStaff.drumStyleTable = #(alist->hash-table mydrums)
  \set DrumStaff.clefPosition = 0.5
  \override DrumStaff.StaffSymbol.line-positions = #'(-2 3)
  \override Staff.BarLine.bar-extent = #'(-1.0 . 1.5)

  \time 2/4
  rb8 8 cb8 16 rb16-> ~ |
  16 8 16 cb8 8 |
}
```

Different Bells \[\begin{array}{c}
\text{rb}6 8 \text{cb}8 16 \text{rb}16-> ~ |
16 8 16 \text{cb}8 8 |
\end{array}\]

Creare indicazioni metronomiche in modalità testuale

Si possono creare nuove indicazioni metronomiche in modalità testuale, ma non modificheranno il tempo del file MIDI.

```latex
\relative c' {
  \tempo \markup {
    \concat {
      \smaller \general-align #Y #DOWN \note {16.} #1
      " = "
      \smaller \general-align #Y #DOWN \note {8} #1
    }
  }
  c1
  c4 c' c,2
```
Disegnare manualmente le legature di valore

Le legature di valore possono essere disegnate a mano cambiando la proprietà `tie-configuration` dell’oggetto `TieColumn`. Il primo numero indica la distanza dal centro del rigo nell’unità di metà spazio rigo, mentre il secondo numero indica la direzione (1 = su, -1 = giù).

Si noti che LilyPond fa una distinzione tra valori precisi e imprecisi per il primo numero. Se si usa un valore preciso (ovvero un intero o una frazione come `/4.5/`), il valore serve come posizione verticale approssimata che viene ulteriormente aggiustata da LilyPond per far sì che la legatura di valore eviti le linee del rigo. Se si usa un valore impreciso, come una virgola mobile, viene usato per la posizione verticale senza ulteriori regolazioni.

```
\relative c' { 
  \override TieColumn.tie-configuration = #'((0.0 1) (-2.0 1) (-4.0 1)) 
  \relative c' { 
    \repeat tremolo 8 { a32 f } 
    \override Beam.gap-count = #1 
    \repeat tremolo 8 { a32 f } 
    \override Beam.gap-count = #2 
    \repeat tremolo 8 { a32 f } 
    \override Beam.gap-count = #3 
    \repeat tremolo 8 { a32 f } 
    \override Beam.gap-count = #3 
    \override Beam.gap = #1.33 
  } 
}
```

Engraving tremolos with floating beams

If a tremolo’s total duration is less than a quarter-note, or exactly a half-note, or between a half-note and a whole-note, it is normally typeset with all beams touching the stems. Certain engraving styles typeset some of these beams as centered floating beams that do not touch the stems. The number of floating beams in this type of tremolo is controlled with the ‘gap-count’ property of the `Beam` object, and the size of the gaps between beams and stems is set with the ‘gap’ property.

```
\relative c' { 
  \repeat tremolo 8 { a32 f } 
  \override Beam.gap-count = #1 
  \repeat tremolo 8 { a32 f } 
  \override Beam.gap-count = #2 
  \repeat tremolo 8 { a32 f } 
  \override Beam.gap-count = #3 
  \repeat tremolo 8 { a32 f } 
  \override Beam.gap-count = #3 
  \override Beam.gap = #1.33 
}
```
Inserire vari gruppi irregolari usando una sola volta il comando \tuplet

La proprietà tupletSpannerDuration imposta la durata di ognuno dei gruppi irregolari compresi tra parentesi dopo il comando \tuplet. In questo modo si possono inserire molti gruppi irregolari consecutivi all’interno di una singola espressione \tuplet, risparmiando così tempo e spazio.

Ci sono vari modi per impostare tupletSpannerDuration. Il comando \tupletSpan la imposta su una certa durata e poi la annulla quando invece di una durata viene specificato \default. Altrimenti si può usare un argomento opzionale con \tuplet.

\relative c’ {
\time 2/4
\tuplet 3/2 { c8"\tupletSpan 4" c c c c c }
\tupletSpan \default
\tuplet 3/2 4 { c8"\tupletSpan \default" c c c c c }
\tuplet 3/2 4 {...} c c c c c }

Code e punte delle travature

È possibile ottenere delle codette su note isolate e dei tratti di suddivisione all’estremità della travatura con una combinazione di stemLeftBeamCount, stemRightBeamCount e una coppia di indicatori della travatura [].

Per ottenere delle codette rivolte a destra, si usa la coppia di indicatori [] e si imposta stemLeftBeamCount a zero (vedi Example 1).

Per ottenere delle codette rivolte a sinistra, si imposta invece stemRightBeamCount (Example 2).

Perché i tratti di suddivisione alla fine di un gruppo di note unite da travatura siano rivolti a destra, si imposta stemRightBeamCount su un valore positivo. Perché i tratti di suddivisione all’inizio di un gruppo di note unite da travatura siano rivolti a sinistra, si imposta invece stemLeftBeamCount (Example 3).
Talvolta, ad esempio per una nota isolata circondata da pause, ha senso avere una coda che punti sia a destra che a sinistra. Lo si può fare con una coppia di indicatori di travatura [ ] da soli (Example 4).

(Nota che \set stemLeftBeamCount è sempre equivalente a \once \set. In altre parole, le impostazioni che definiscono il conteggio delle travature non “permangono”, quindi la coppia di code attaccate al 16[ ] solitario nell’ultimo esempio non hanno nulla a che fare con l’impostazione \set di due note prima.)

\score {
<<
  % Example 1
  \new RhythmicStaff {
    \set stemLeftBeamCount = #0
    c16
    r8.
  }
  % Example 2
  \new RhythmicStaff {
    r8.
    \set stemRightBeamCount = #0
    16
  }
  % Example 3
  \new RhythmicStaff {
    16 16
    \set stemRightBeamCount = #2
    16 r r
    \set stemLeftBeamCount = #2
    16 16 16
  }
  % Example 4
  \new RhythmicStaff {
    16 16
    \set stemRightBeamCount = #2
    16 r16
    16[]
    r16
    \set stemLeftBeamCount = #2
    16 16
  }
>>
}
Forcing rehearsal marks to start from a given letter or number

This snippet demonstrates how to obtain automatic ordered rehearsal marks, but from the letter or number desired.

\relative c'' {
  c1 \mark \default
  c1 \mark \default
  c1 \mark \default
  c1 \mark #14
  c1 \mark \default
  c1 \mark \default
  c1 \mark \default
}

Generating custom flags

The stencil property of the Flag grob can be set to a custom scheme function to generate the glyph for the flag.

#(define-public (weight-flag grob)
  (let* ((stem-grob (ly:grob-parent grob X))
         (log (- (ly:grob-property stem-grob 'duration-log) 2))
         (is-up? (eqv? (ly:grob-property stem-grob 'direction) UP))
         (yext (if is-up? (cons (* log -0.8) 0) (cons 0 (* log 0.8))))
         (flag-stencil (make-filled-box-stencil '(-0.4 . 0.4) yext))
         (stroke-style (ly:grob-property grob 'stroke-style))
         (stroke-stencil (if (equal? stroke-style "grace")
                           (make-line-stencil 0.2 -0.9 -0.4 0.9 -0.4)
                           empty-stencil)))
  (ly:stencil-add flag-stencil stroke-stencil)))

% Create a flag stencil by looking up the glyph from the font
#(define (inverted-flag grob)
(let* ((stem-grob (ly:grob-parent grob X))
   (dir (if (eqv? (ly:grob-property stem-grob 'direction) UP) "d" "u"))
   (flag (retrieve-glyph-flag "" dir "" grob))
   (line-thickness (ly:staff-symbol-line-thickness grob))
   (stem-thickness (ly:grob-property stem-grob 'thickness))
   (stem-width (* line-thickness stem-thickness))
   (stroke-style (ly:grob-property grob 'stroke-style))
   (stencil (if (null? stroke-style)
     flag
     (add-stroke-glyph flag stem-grob dir stroke-style "")))
   (rotated-flag (ly:stencil-rotate-absolute stencil 180 0 0)))

snippetexamplenotes =
{
  \autoBeamOff c'8 d'16 c'32 d'64 \acciaccatura \{c'8\} d'64
}

{
  \time 1/4
  \textMark "Normal flags"
  \snippetexamplenotes

  \textMark "Custom flag: inverted"
  \override Flag.stencil = #inverted-flag
  \snippetexamplenotes

  \textMark "Custom flag: weight"
  \override Flag.stencil = #weight-flag
  \snippetexamplenotes

  \textMark "Revert to normal"
  \revert Flag.stencil
  \snippetexamplenotes
}

Ritmi di accompagnamento per chitarra

Per la musica per chitarra, è possibile mostrare i ritmi di accompagnamento, insieme alle note della melodia e ai nomi e ai diagrammi degli accordi.

\include "predefined-guitar-fretboards.ly"
<<
\new ChordNames {
  \chordmode {
    c1 | f | g | c
  }
}
Heavily customized polymetric time signatures

Though the polymetric time signature shown was not the most essential item here, it has been included to show the beat of this piece (which is the template of a real Balkan song!).

melody = \relative c'' {
\key g \major
\compoundMeter #'((3 8) (2 8) (2 8) (3 8) (2 8) (2 8) (2 8) (2 8)
(2 8) (2 8) (3 8) (2 8) (2 8))
c8 c c d4 c8 c b c b a4 g fis8 e d c b’ c d e4- fis8 g \break
c,4. d4 c4. c4 d2 d4. e4- d4
\break
c4. d4 c4. c4 d c2 d4. e4- d4
\break
c4. d4 c4. c4 d c2 d4. e4- d4
\break
}

\new DrumStaff \drummode\
\repeat volta 2 {
  bd4.\markup { Drums } sn4 bd \bar ";"
  sn4. bd4 sn \bar ";"
  bd sn bd4. sn4 bd
}

\new Staff \with {
  \instrumentName = \markup { \concat { "B" \flat " Sop." } }
}

{\melody \drum }

High and Low woodblock example

Two Woodblocks, entered with 'wbh' (high woodblock) and 'wbl' (low woodblock). The length of the barline has been altered with an \override command otherwise it would be too short. The positions of the two stafflines also have to be explicitly defined.
\paper { \tagline = ##f }
% These lines define the position of the woodblocks in the stave;
% if you like, you can change it or you can use special note heads
% for the woodblocks.
#(define mydrums '((hiwoodblock default #f 3)
           (lowoodblock default #f -2)))

woodstaff = {
  % This defines a staff with only two lines.
  % It also defines the positions of the two lines.
  \override Staff.StaffSymbol.line-positions = #'(-2 3)

  % This is necessary; if not entered,
  % the barline would be too short!
  \override Staff.BarLine.bar-extent = '#'(-1.0 . 1.5)
  % small correction for the clef:
  \set DrumStaff.clefPosition = 0.5
}

\new DrumStaff {
  % with this you load your new drum style table
  \set DrumStaff.drumStyleTable = #(alist->hash-table mydrums)
  \woodstaff

  \drummode {
    \time 2/4
    wb18 16 16 8-> 8 |
    wb18 16 16-> - 16 16 r8 |
  }
}

Making an object invisible with the 'transparent property

Setting the transparent property will cause an object to be printed in “invisible ink”; the object is not printed, but all its other behavior is retained. The object still takes up space, it takes part in collisions, and slurs, ties and beams can be attached to it.

This snippet demonstrates how to connect different voices using ties. Normally, ties only connect two notes in the same voice. By introducing a tie in a different voice, and blanking the first up-stem in that voice, the tie appears to cross voices.

\relative {
  \time 2/4
  \relative & \once \hide Stem
  \once \override Stem.length = #8
  b'8 ~ 8\noBeam
  \once \hide Stem
  \once \override Stem.length = #8
Legature di portamento con complesse strutture di tratteggio

Le legature di portamento possono avere schemi di tratteggio complessi definendo la proprietà dash-definition. dash-definition è una lista di dash-elements. Un dash-element è una lista di parametri che definiscono il comportamento del tratteggio per un segmento della legatura.

La legatura di portamento è definita come il parametro \( t \) della curva di bezier che va da 0 sul margine sinistro della legatura fino a 1 su quello destro. dash-element è una lista di \( (\text{inizio-}t \ \text{fine-}t \ \text{frazione-trattino} \ \text{punto-trattino}) \). La regione della legatura di portamento che va da \( \text{inizio-}t \ a \ \text{fine-}t \) avrà una frazione frazione-trattino di ogni punto-trattino nero. punto-trattino viene definito in spazi rigo. frazione-trattino è impostato su 1 per una legatura di portamento continua.

\[
\text{\relative c' \ { \\
\once \override Slur.dash-definition = #'((0 0.3 0.1 0.75)  \\
  (0.3 0.6 1 1)  \\
  (0.65 1.0 0.4 0.75))}
\]
\text{c4( d e f)}

\[
\text{\once \override Slur.dash-definition = #'((0 0.25 1 1)  \\
  (0.3 0.7 0.4 0.75)  \\
  (0.75 1.0 1 1))}
\]
\text{c4( d e f)}

Manually controlling beam positions

Beam positions may be controlled manually, by overriding the positions setting of the Beam grob.

\[
\text{\relative c' \ { \\
\time 2/4  \\
% from upper staff-line (position 2) to center (position 0) \\
\override Beam.positions = #'(2 . 0) \\
c8 c  \\
% from center to one above center (position 1)}
\]
\override Beam.positions = #'(0 . 1)
c8 c
}

\begin{music}
\begin{notation}
\clef g clef
\end{notation}
\end{music}

Merging multi-measure rests in a polyphonic part

When using multi-measure rests in a polyphonic staff, the rests will be placed differently depending on the voice they belong to. However they can be printed on the same staff line, using the following setting.

\normalPos = \revert MultiMeasureRest.direction

{
   <<
   {
      c''1
      R1
      c''1
      \normalPos
      R1
   }
   \\%
   {
      c'1
      R1
      c'1
      \normalPos
      R1
   }
   >>
}

\begin{music}
\begin{notation}
\clef g clef
\end{notation}
\end{music}

Modifying tuplet bracket length

Tuplet brackets can be made to run to prefatory matter or the next note. Default tuplet brackets end at the right edge of the final note of the tuplet; full-length tuplet brackets extend farther to the right, either to cover all the non-rhythmic notation up to the following note, or to cover only the whitespace before the next item of notation, be that a clef, time signature, key signature, or another note. The example shows how to switch tuplets to full length mode and how to modify what material they cover.

\new RhythmicStaff {\%
\set tupletFullLength = ##t
\%
\set tupletFullLengthNote = ##t
}
Spostare le note puntate in polifonia

Quando una nota puntata della voce più alta viene spostata per evitare una collisione con una nota di un'altra voce, il comportamento predefinito è spostare la nota più alta a destra. Tale comportamento può essere modificato tramite la proprietà prefer-dotted-right di NoteCollision.

Regolazione della lunghezza delle pause multiple

Le pause multiple hanno una lunghezza che dipende dalla loro durata totale e tale lunghezza è regolata da MultiMeasureRest.space-increment. Nota che il valore predefinito è 2.0.
Testo a margine delle pause multiple

Il testo a margine di una pausa multipla viene centrato sopra o sotto di essa. Se il testo è lungo, la misura non si espanderà. Per espandere la pausa multipla in modo che si allinei col testo, conviene usare un accordo vuoto con del testo attaccato prima della pausa multipla.

Il testo così attaccato a una nota spaziatrice viene allineato a sinistra della posizione in cui la nota sarebbe posta nella misura, ma se la lunghezza della misura è determinata dalla lunghezza del testo, il testo verrà centrato.

\relative c' {
  \compressMMRests {
    \textLengthOn
    \markup { [MAJOR GENERAL] }
    R1*19
    \markup { italic { Cue: ... it is yours } }
    \markup { k }  
    R1*30 \markup { [MABEL] }
    \textLengthOff
    c4 \markup { CHORUS } d f c
  }
}

[CMAJOR GENERAL] [30]

Numeri non predefiniti per i gruppi irregolari

LilyPond fornisce anche funzioni di formattazione che permettono di creare numeri di gruppi irregolari diversi dalla frazione vera e propria, così come di aggiungere un valore di nota al numero o alla frazione di un gruppo irregolare.

\relative c'' {
  \once \override TupletNumber.text =  
    #(tuplet-number::non-default-tuplet-denominator-text 7)
  \tuplet 3/2 { c4. c4. c4. c4. }  
  \once \override TupletNumber.text =  
    #(tuplet-number::non-default-tuplet-fraction-text 12 7)
  \tuplet 3/2 { c4. c4. c4. c4. }  
  \once \override TupletNumber.text =  
    #(tuplet-number::append-note-wrapper
      (tuplet-number::non-default-tuplet-fraction-text 12 7)
      (ly:make-duration 3 0)))
  \tuplet 3/2 { c4. c4. c4. c4. }  
  \once \override TupletNumber.text =  
    #(tuplet-number::append-note-wrapper
      tuplet-number::calc-denominator-text
      (ly:make-duration 2 0))

Numbering single measure rests

Multi measure rests show their length by a number except for single measures. This can be changed by setting `restNumberThreshold`.

```
{ 
\compressEmptyMeasures 
R1 R1*10 R1*11 \bar "\|\|" 
\set restNumberThreshold = 0 
R1 R1*10 R1*11 \bar "\|\|" 
\set restNumberThreshold = 10 
R1 R1*10 R1*11 
}
```

PartCombine e autoBeamOff

La funzione \autoBeamOff, se usata insieme a \partCombine, può essere difficile da comprendere. 
È preferibile usare invece 
\set Staff.autoBeaming = ##f per assicurarsi che la disposizione delle travature sia disabilitata per tutto il rigo.
\partCombine funziona con 3 voci – gambo in su singolo, gambo in giù singolo, gambo in su unito.
L’uso di \autoBeamOff all’interno del primo argomento di partCombine ha effetto sulla voce che è attiva al momento in cui la funzione viene elaborata, ovvero sul gambo in su singolo o sul gambo in giù unito. L’uso di \autoBeamOff nel secondo argomento avrà effetto sulla voce che ha il gambo in giù singolo.

Per poter usare \autoBeamOff per impedire tutte le disposizioni automatiche delle travature, se usato con \partCombine, è necessario richiamare tre volte la funzione \autoBeamOff.

\[
\%\set\text{Staff.autoBeaming} = \#f \% \text{turns off all autobeaming} \\
\partCombine \\
\{ \\
\autoBeamOff \% \text{applies to split up stems} \\
\\repeat\text{unfold} 4 a'16 \\
\%\autoBeamOff \% \text{applies to combined up stems} \\
\\repeat\text{unfold} 4 a'8 \\
\\repeat \text{unfold} 4 a'16 \\
\} \\
\{ \\
\autoBeamOff \% \text{applies to down stems} \\
\\repeat\text{unfold} 4 f'8 \\
\\repeat\text{unfold} 8 f'16 \mid \\
\}
\]

\[\]

Percussion example

A short example taken from Stravinsky’s L’Histoire du soldat.

\#(define mydrums '((bassdrum default #f 4) 
(snare default #f -4) 
(tambourine default #f 0)))

global = 
\{ 
\time 3/8 s4. 
\time 2/4 s2\times2 
\time 3/8 s4. 
\time 2/4 s2 
\}

drumsA = 
\{ 
\context DrumVoice << 
\{ \global \} 
\{ \drummode \ 
\autoBeamOff 
\stemDown sn8 \stemUp tamb s8 | 
\stemDown sn4 \stemDown sn4 | 
\stemUp tamb8 \stemDown sn8 \stemUp sn16 \stemDown sn \stemUp sn8 | 
\stemDown sn8 \stemUp tamb s8 | 
\stemUp sn4 s8 \stemUp tamb 
\}


\[\]
Consentire l’interruzione del rigo all’interno di gruppi irregolari con travature

Questo esempio artificioso mostra come permettere interruzioni del rigo sia manuali che automatiche all’interno di un gruppo irregolare con travature. Si noti che le travature di questi gruppi irregolari fuori dal ritmo devono essere disposte manualmente.

\layout {
  context {
Positioning grace note beams at the height of normal note beams

When notes are placed on ledger lines, their beams are usually centred on the stave. Grace note beams are shorter and grace notes on ledger lines may well have beams outside the stave. You can override this beaming for grace notes.

Posizionare gli abbellimenti con dello spazio fluttuante

Se si imposta la proprietà 'strict-grace-spacing', le colonne musicali degli abbellimenti 'fluttuano', ovvero si scollegano dalle note normali: prima vengono spaziate le note normali, poi le colonne musicali degli abbellimenti vengono messe a sinistra delle colonne delle note principali.
Posizionamento delle pause multiple

Diversamente dalle pause normali, non esiste un comando predefinito per cambiare la posizione sul rigo di un simbolo di pausa multipla di qualsiasi tipo connettendolo a una nota. Tuttavia, nella musica polifonica le pause multiple nelle voci dispari e pari sono separate verticalmente. Il posizionamento delle pause multiple si controlla nel modo seguente:

\relative c'' {
  \override Score.SpacingSpanner.strict-grace-spacing = ##t
  \new Staff \new Voice {
    \afterGrace c4 { c16{ c8 c16 } }
    \afterGrace c8{ \grace { b16 d } c8 }
    c4 r
  }
  \new Staff {
    c16 c c c c c c c c4 r
  }
}

\relative c'' {
  % Multi-measure rests by default are set under the fourth line
  R1
  % They can be moved using an override
  \override MultiMeasureRest.staff-position = #-2
  R1
  \override MultiMeasureRest.staff-position = #0
  R1
  \override MultiMeasureRest.staff-position = #2
  R1
  \override MultiMeasureRest.staff-position = #3
  R1
  \override MultiMeasureRest.staff-position = #6
  R1
  \revert MultiMeasureRest.staff-position
  \break
  % In two Voices, odd-numbered voices are under the top line
  << { R1 } \ \ { a1 } >>
  % Even-numbered voices are under the bottom line
  << { a1 } \ \ { R1 } >>
  % Multi-measure rests in both voices remain separate
  << { R1 } \ \ { R1 } >>
% Separating multi-measure rests in more than two voices
% requires an override
<< \{ R1 \} \\ \{ R1 \} \\ 
   \once \override MultiMeasureRest.staff-position = #0 
   \{ R1 \} >>

% Using compressed bars in multiple voices requires another override
% in all voices to avoid multiple instances being printed
\compressMMRests
<< \revert MultiMeasureRest.direction 
   \{ R1*3 \} \\ 
   \revert MultiMeasureRest.direction 
   \{ R1*3 \} >>

\new GrandStaff << 
\new Staff \music 
\new Staff \music 
>>

\new Staff \music
\new Staff \music

\begin{verbatim}
\begin{music}
\times 4 \c4 \times 4 \c4 \times 4 \c4 \times 4 \c4 \times 4 \c4 \\
\end{music}
\end{verbatim}

\begin{verbatim}
\begin{music}
\times 4 \c4 \times 4 \c4 \times 4 \c4 \times 4 \c4 \\
\end{music}
\end{verbatim}

Positioning opposing fermatas on a bar line

This snippet demonstrates a command that prints fermatas both above and below a bar line. If there would not otherwise be a bar line, it adds a double bar line. Semantically, the command codes a longer-than-normal caesura, which might be considered misuse depending on the situation.

twoWayFermata = {
   \once \set Staff.caesuraType = #'(underlying-bar-line , "||")
   \once \set Staff.caesuraTypeTransform = ##f
   \caesura \fermata _ /fermata
}

\begin{verbatim}
\begin{music}
  f'1 \twoWayFermata \\
  R1 \\
  f'2 \twoWayFermata f'2 \\
  R1 \\
  b'1 \twoWayFermata \fine
\end{music}
\end{verbatim}

Preventing final mark from removing final tuplet

The addition of a final mark can result in the loss of a final tuplet marking. This can be overcome by setting `TupletBracket.full-length-to-extent` to false.

% due to issue 2362 a long mark such as
% \textEndMark "Composed Feb 2007 - Feb 2008"
% cannot be used here.

\new Staff {\set tupletFullLength = ##t}
  \time 1/8
  \tuplet 3/2 8 \{ c'16 c' c' c' c' c' c' c' c' \}
  \tweak direction #DOWN \textEndMark "1234"
}

\new Staff {\set tupletFullLength = ##t}
  \override TupletBracket.full-length-to-extent = ##f
  \time 1/8
  \tuplet 3/2 8 \{ c'16 c' c' c' c' c' c' c' c' \}
  \tweak direction #DOWN \textEndMark "1234"
}

Mostrare i numeri di battuta a intervalli regolari

Impostando la proprietà `barNumberVisibility` si possono far apparire i numeri di battuta a intervalli regolari. In questo esempio vengono mostrati ogni due misure eccetto alla fine della linea.

\relative c' {\set Score.currentBarNumber = #11}
  \override Score.BarNumber.break-visibility = #end-of-line-invisible
  \set Score.barNumberVisibility = #{every-nth-bar-number-visible 2}
Printing bar numbers for broken measures

By default a BarNumber of a broken measure is not repeated at the beginning of the new line. Use first-bar-number-invisible-save-broken-bars for barNumberVisibility to get a parenthesized BarNumber there.

```
\layout {
  \context {
    \Score
      barNumberVisibility = #first-bar-number-invisible-save-broken-bars
      \override BarNumber.break-visibility = ##(#f #t #t)
  }
}
```

```
\relative c' {
  c1 | d | e | f2 \bar "" \break
  fis | gl | e2 \bar "" \break
  <> "reenabled default"
  % back to default -
  % \unset Score.barNumberVisibility would do so as well
  \set Score.barNumberVisibility =
    #first-bar-number-invisible-and-no-parenthesized-bar-numbers
  es | d1 | c
}
```
Numeri di battuta racchiusi in rettangoli o cerchi

I numeri di battuta possono apparire anche all’interno di rettangoli o cerchi.
\relative c’ {
  % Prevent bar numbers at the end of a line and permit them elsewhere
  \override Score.BarNumber.break-visibility = #end-of-line-invisible
  \set Score.barNumberVisibility = #(every-nth-bar-number-visible 4)
}

% Increase the size of the bar number by 2
\override Score.BarNumber.font-size = #2

% Draw a box round the following bar number(s)
\override Score.BarNumber.stencil
  = #(make-stencil-boxer 0.1 0.25 ly:text-interface::print)
\repeat unfold 5 { c1 }

% Draw a circle round the following bar number(s)
\override Score.BarNumber.stencil
  = #(make-stencil-circler 0.1 0.25 ly:text-interface::print)
\repeat unfold 4 { c1 } \bar "|."

Printing bar numbers using modulo-bar-number-visible

If the remainder of the division of the current BarNumber by the first argument of modulo-bar-number-visible equals its second argument print the BarNumber.

Useful to print the BarNumber at certain distances, p.e:

• \( \text{modulo-bar-number-visible 3 2} \) -> prints 2,5,8
• \( \text{modulo-bar-number-visible 4 2} \) -> prints 2,6,10
• \( \text{modulo-bar-number-visible 3 1} \) -> prints 3,5,7
• \( \text{modulo-bar-number-visible 5 2} \) -> prints 2,7,12

\layout {
  \context {
    \Score
    \override BarNumber.break-visibility = ##(#f #t #t)
    barNumberVisibility = #(modulo-bar-number-visible 3 2)
  }
}

\relative c’ {
  c1 | d | e | f \break
  g1 | e | d | c
}

### Rhythms 60

Numeri di battuta racchiusi in rettangoli o cerchi

I numeri di battuta possono apparire anche all’interno di rettangoli o cerchi.
\relative c’ {
  % Prevent bar numbers at the end of a line and permit them elsewhere
  \override Score.BarNumber.break-visibility = #end-of-line-invisible
  \set Score.barNumberVisibility = #(every-nth-bar-number-visible 4)
}

% Increase the size of the bar number by 2
\override Score.BarNumber.font-size = #2

% Draw a box round the following bar number(s)
\override Score.BarNumber.stencil
  = #(make-stencil-boxer 0.1 0.25 ly:text-interface::print)
\repeat unfold 5 { c1 }

% Draw a circle round the following bar number(s)
\override Score.BarNumber.stencil
  = #(make-stencil-circler 0.1 0.25 ly:text-interface::print)
\repeat unfold 4 { c1 } \bar "|."

Printing bar numbers using modulo-bar-number-visible

If the remainder of the division of the current BarNumber by the first argument of modulo-bar-number-visible equals its second argument print the BarNumber.

Useful to print the BarNumber at certain distances, p.e:

• \( \text{modulo-bar-number-visible 3 2} \) -> prints 2,5,8
• \( \text{modulo-bar-number-visible 4 2} \) -> prints 2,6,10
• \( \text{modulo-bar-number-visible 3 1} \) -> prints 3,5,7
• \( \text{modulo-bar-number-visible 5 2} \) -> prints 2,7,12

\layout {
  \context {
    \Score
    \override BarNumber.break-visibility = ##(#f #t #t)
    barNumberVisibility = #(modulo-bar-number-visible 3 2)
  }
}

\relative c’ {
  c1 | d | e | f \break
  g1 | e | d | c
}
Stampare i numeri di battuta a intervalli regolari variabili

Tramite la funzione di contesto \{set-bar-number-visibility\} si possono modificare gli intervalli dei numeri di battuta.

\relative c' {  
  \override Score.BarNumber.break-visibility = #end-of-line-invisible  
  \context Score \applyContext #(set-bar-number-visibility 4)  
  \repeat unfold 10 c'\1  
  \repeat unfold 10 c 
}

Posizionare il metronomo e i numeri di chiamata sotto il rigo

Di norma, il metronomo e i numeri di chiamata vengono posizionati sopra il rigo. Per metterli sotto il rigo basta impostare correttamente la proprietà \texttt{direction} di \texttt{MetronomeMark} o \texttt{RehearsalMark}.

\layout {  
  indent = 0  
  ragged-right = ##f 
}

{  
  \override Score.MetronomeMark.direction = #DOWN  
  \tempo 8. = 120  
  c''\1  
  \override Score.RehearsalMark.direction = #DOWN  
  \mark \default  
  c''\1 
}

\tempo J. = 120

A
Printing music with different time signatures

In the following snippet, two parts have a completely different time signature, yet remain synchronized.

The bar lines can no longer be printed at the Score level; to allow independent bar lines in each part, the Timing_translator is moved from the Score context to the Staff context.

If bar numbers are required, the Bar_number_engraver should also be moved, since it relies on properties set by the Timing_translator; a \with block can be used to add bar numbers to the relevant staff.

```latex
\paper {
  \indent = #0
  \ragged-right = ##t
}

global = { \time 3/4 { s2.*3 } \bar "" \break { s2.*3 } }

layout {
  \context {
    \Score
    \remove "Timing_translator"
    \remove "Bar_number_engraver"
    \override SpacingSpanner.uniform-stretching = ##t
    \override SpacingSpanner.strict-note-spacing = ##t
    proportionalNotationDuration = #(ly:make-moment 1/64)
  }
  \context {
    \Staff
    \consists "Timing_translator"
  }
  \context {
    \Voice
    \remove "Forbid_line_break_engraver"
    tupleFullLength = ##t
  }
}

Bassklarinette = \new Staff \with {
  \consists "Bar_number_engraver"
  \barNumberVisibility = #((every-nth-bar-number-visible 2)
  \override BarNumber.break-visibility = #end-of-line-invisible
} <<

\global {
  \bar "|
  \clef treble
  \time 3/8
  d''4.
  \bar "|
  \time 3/4
  r8 des''2( c''8)
  \bar "|
}
\time 7/8
r4. ees'2 ~
\bar "|
\time 2/4
\tupletUp
\tuplet 3/2 \{ ees''4 r4 d''4 ~ \}
\bar "|
\time 3/8
\tupletUp
\tuplet 4/3 \{ d''4 r4 \}
\bar "|
\time 2/4
e''2
\bar "|
\time 3/8
es''4.
\bar "|
\time 3/4
r8 d''2 r8
\bar "|
}\>

**Perkussion = \new StaffGroup <<
\new Staff <<
\global {
  \bar "|
  \clef percussion
  \time 3/4
  r4 c'2 ~
  \bar "|
c'2.
  \bar "|
R2.
  \bar "|
r2 g'4 ~
  \bar "|
g'2. ~
  \bar "|
g'2.
}\>

}
\new Staff <<
\global {
  \bar "|"
  \clef percussion
  \time 3/4
  R2.

  \bar "|"
  g'2. ~

  \bar "|"
  g'2.

  \bar "|"
  r4 g'2 ~

  \bar "|"
  g'2 r4

  \bar "|"
  g'2.
}
>>
>>

\score {
  <<
    \Bassklarinette
    \Perkussion
  >>

Mostrare il numero di battuta nella prima misura

Il primo numero di battuta di una partitura viene soppresso se è inferiore o uguale a ‘1’. Se si imposta barNumberVisibility su all-bar-numbers-visible, verrà mostrato il numero di battuta della prima misura e di tutte quelle successive. Si noti che perché funzioni è necessario inserire una stanghetta invisibile prima della prima nota.

```
\layout {
    indent = 0
    ragged-right = ##t
}

\relative c' {
    \set Score.barNumberVisibility = #all-bar-numbers-visible
    c1 | d | e | f \break
    g1 | e | d | c
}
```

Printing tuplet brackets on the note head side

Whichever option you choose for controlling the tuplet bracket visibility, it will show or hide the tuplet bracket irrespectively of tuplet bracket placement (stem side or note head side). However,
when placing the tuplet bracket on the note head side some authors recommend always printing the tuplet bracket. The option visible-over-note-heads can be used to achieve this.

```
music = \relative c' {  
  \tupletNeutral \tuplet 3/2 { c16[ d e } f8]  
  \tupletUp \tuplet 3/2 { c8 d e }  
}  
```

\new Voice {  
  \relative c' {  
    \time 2/4  
    \override TupletBracket.visible-over-note-heads = ##t  
    \override Score.TextMark.non-musical = ##f  
    { \textMark \markup "default" \music }  
    \override TupletBracket.bracket-visibility = #'if-no-beam  
    { \textMark \markup "'if-no-beam" \music }  
  }  
}

![Musical notation example]

Redefine the global default abbelliments settings

The global default abbelliments settings are saved in the following identifiers:

- startGraceMusic
- stopGraceMusic
- startAcciaccaturaMusic
- stopAcciaccaturaMusic
- startAppoggiaturaMusic
- stopAppoggiaturaMusic

They are defined in the file ly/grace-init.ly. Redefining them can achieve various effects.

```
startAcciaccaturaMusic = {  
  <>  
  \override Flag.stroke-style = "grace"  
  \slurDashed  
}  
```

```
stopAcciaccaturaMusic = {  
  \revert Flag.stroke-style  
  \slurSolid  
  <>  
}  
```

```
\relative c'' {  
  \acciaccatura d8 c1  
}  
```
Togliere i numeri di battuta da uno spartito

I numeri di battuta possono essere tolti rimuovendo l’incisore Bar_number_engraver dal contesto Score.

```latex
\layout {
  \context {
    \Score
    \omit BarNumber
    \% or:
    \% remove "Bar_number_engraver"
  }
}

\relative c' {
  c4 c c c \break
  c4 c c c
}
```

Removing connecting bar lines on StaffGroup, PianoStaff, or GrandStaff

By default, bar lines in StaffGroup, PianoStaff, or GrandStaff groups are connected between the staves, i.e. a SpanBar is printed. This behaviour can be overridden on a staff-by-staff basis.

```latex
\relative c' {
  \new StaffGroup <<
    \new Staff {
      e1 | e
      \once \override Staff.BarLine.allow-span-bar = ##f
      e1 | e | e
    }
    \new Staff {
      c1 | c | c
      \once \override Staff.BarLine.allow-span-bar = ##f
      c1 | c
    }
    \new Staff {
      a1 | a | a | a | a
    }
  >>
```

\[ \text{\( \begin{array}{c}
C C C C \\text{break} \\
C C C C
\end{array} \)} \]
Stili di pausa

Esistono vari stili di pausa.

\new Staff \relative c { 
\omit Score.TimeSignature
\cadenzaOn
\override Staff.Rest.style = #'mensural
r\maxima~\markup \typewriter { mensural }
\bar ""
\break

\override Staff.Rest.style = #'neomensural
r\maxima~\markup \typewriter { neomensural }
\bar ""
\break

\override Staff.Rest.style = #'classical
r\maxima~\markup \typewriter { classical }
\bar ""
\break

\override Staff.Rest.style = #'z
r\maxima~\markup \typewriter { z-style }
\bar ""
\break

\override Staff.Rest.style = #'default
r\maxima~\markup \typewriter { default }
\bar ""
\break

\override Staff.Rest.style = #'mensural
Reverting default beam endings

To typeset beams grouped 3-4-3-2 in 12/8 it is necessary first to override the default beam endings in 12/8, and then to set up the new beaming endings:

```
\relative c' { 
  \time 12/8

  % Default beaming
  a8 a a a a a a a a a a

  % Set new values for beam endings
  \set Score.beatStructure = 3,4,3,2
  a8 a a a a a a a a a a a a
}
```

Rhythmic slashes

In “simple” lead-sheets, sometimes no actual notes are written, instead only “rhythmic patterns” and chords above the measures are notated giving the structure of a song. Such a feature is for example useful while creating/transcribing the structure of a song and also when sharing lead sheets with guitarists or jazz musicians.

The standard support for this using \repeat percent is unsuitable here since the first beat has to be an ordinary note or rest.

This example shows two solutions to this problem, by redefining ordinary rests to be printed as slashes. (If the duration of each beat is not a quarter note, replace the r4 in the definitions with a rest of the appropriate duration).

```
% Macro to print single slash
rs = {
  \once \override Rest.stencil = #ly:percent-repeat-interface::beat-slash
  \once \override Rest.thickness = #0.48
  \once \override Rest.slope = #1.7
```
r4
}

% Function to print a specified number of slashes
comp = #((define-music-function (count) (integer?))
    #
    \override Rest.stencil = #ly:percent-repeat-interface::beat-slash
    \override Rest.thickness = #0.48
    \override Rest.slope = #1.7
    \repeat unfold $count { r4 }
    \revert Rest.stencil
#
)

\score {
    \relative c' {
        c4 d e f |
        \rs \rs \rs \rs |
        \comp #4 |
    }  
}

\relative c'' { a1 | a }
\new Lyrics \lyricmode { \skip 1 bla1 }
>>

\relative c' { a1 | a }
\new Lyrics \lyricmode { \skip 1 bla1 }

Skips in lyric mode (2)

Although \skip skips cannot be used in \lyricmode (it is taken to be a literal “s”, not a space),
double quotes (""") or underscores (_) are available.

So for example:

<<
\relative c'' { a4 b c d }
\new Lyrics \lyricmode { a4 "" _ gap }
>>

Skips in lyric mode

The \skip syntax for skips is only available in note mode and chord mode. In other situations, for
example, when entering lyrics, using the \skip command is recommended.
Stemlets

In some notational conventions beams are allowed to extend over rests. Depending on preference, these beams may drop 'stemlets' to help the eye appreciate the rhythm better, and in some modern music the rest itself is omitted and only the stemlet remains.

This snippet shows a progression from traditional notation, to beams over the rest, to stemlets over the rest, to stemlets alone. Stemlets are generated by overriding the 'stemlet-length property of Stem, while rests are hidden by setting 'transparent = ##t.

Some \markup elements are included in the source to highlight the different notations.

```
paper { ragged-right = ##f }

{  
c'16\markup { traditional } d' r f'  
g'16[\markup { beams over rests } f' r d']

% N.B. use Score.Stem to set for the whole score.  
\override Staff.Stem.stemlet-length = #0.75

  c'16[\markup { stemlets over rests } d' r f']
  g'16[\markup { stemlets and no rests } f']
  \once \hide Rest
  r16 d']
}
```

Travatura che segue strettamente il battito

Si possono impostare i tratti di suddivisione della travatura in modo che siano rivolti verso la relativa pulsazione. La prima travatura fa sì che non spuntino i tratti di suddivisione (comportamento predefinito); la seconda travatura è orientata verso la pulsazione.

```
\relative c' {  
  \time 6/8  
a8. a16 a a  
  \set strictBeatBeaming = ##t  
a8. a16 a a  
}  
```
Suddividere le travature

Le travature di note consecutive di un sedicesimo (o più brevi) non vengono suddivise, ovvero i tre (o più) tratti della travatura si estendono, senza spezzarsi, sugli interi gruppi di note. Questo comportamento può essere modificato in modo da suddividere le travature in sottogruppi attraverso la proprietà `subdivideBeams`. Se impostata, le travature che comprendono più sottogruppi verranno suddivise a intervalli definiti dal valore attuale di `baseMoment`, riducendo le travature multiple al numero di travature che indica il valore metrico della suddivisione. Se il gruppo successivo alla suddivisione è più breve del valore metrico corrente (di solito perché la travatura è incompleta), il numero di travature riflette il gruppo di suddivisione più lungo possibile. Tuttavia, se rimane una sola nota dopo la divisione, questa restrizione non viene applicata. Si noti che `baseMoment`, se non impostata esplicitamente, equivale a uno fratto il denominatore dell’attuale indicazione di tempo. Deve quindi essere impostata su una frazione che stabilisca la durata del sottogruppo di travature; lo si può fare usando la funzione `ly:make-moment`, come è mostrato in questo frammento di codice. Inoltre quando `baseMoment` cambia, anche `beatStructure` deve essere modificato per accordarsi con `baseMoment`:

```latex
\relative c' { 
  c32[ c c c c c c c ]
  \set subdivideBeams = ##t 
  c32[ c c c c c c c ]

  \% Set beam sub-group length to an eighth note 
  \set baseMoment = #(ly:make-moment 1/8) 
  \set beatStructure = 2,2,2,2 
  c32[ c c c c c c c ]

  \% Set beam sub-group length to a sixteenth note
  \set baseMoment = #(ly:make-moment 1/16) 
  \set beatStructure = 4,4,4,4 
  c32[ c c c c c c c ]

  \% Shorten beam by 1/32
  \set baseMoment = #(ly:make-moment 1/8) 
  \set beatStructure = 2,2,2,2 
  c32[ c c c c c c c ] r32

  \% Shorten beam by 3/32
  \set baseMoment = #(ly:make-moment 1/8) 
  \set beatStructure = 2,2,2,2 
  c32[ c c c c ] r16.
  r2
}
```

Tam-tam example

A tam-tam example, entered with ‘tt’

```latex
#(define mydrums '((tamtam default #f 0)))
```
\new DrumStaff \with \{ instrumentName = #'Tam tam' \}

\drummode \{
  \set DrumStaff.drumStyleTable = #(alist->hash-table mydrums)
  \override Staff.StaffSymbol.line-positions = #'( 0 )
  \override Staff.BarLine.bar-extent = #'(-1.5 . 1.5)

  tt 1 \pp \laissezVibrer
\}

Tam tam \ działalności

Tam tam example

A tambourine example, entered 'tamb'
\paper \{ tagline = ##f \}

#(define mydrums '((tambourine default #f 0)))

\new DrumStaff \with \{ instrumentName = #'Tambourine' \}

\drummode \{
  \set DrumStaff.drumStyleTable = #(alist->hash-table mydrums)
  \override Staff.StaffSymbol.line-positions = #'( 0 )
  \override Staff.BarLine.bar-extent = #'(-1.5 . 1.5)

  \time 6/8
  tamb8. 16 8 8 8 8 |
  tamb4. 8 8 8 |
  \% the trick with the scaled duration and the shorter rest
  \% is necessary for the correct ending of the trill-span!
  tamb2.*5/6 \startTrillSpan s8 \stopTrillSpan |
\}

Tambourine \_acquire

Three-sided box

This example shows how to add a markup command to get a three sided box around some text (or other markup).

\% New command to add a three sided box, with sides north, west and south
\% Based on the box-stencil command defined in scm/stencil.scm
\% Note that ";;" is used to comment a line in Scheme

#(define-public (NWS-box-stencil stencil thickness padding)
  "Add a box around STENCIL, producing a new stencil."
  (let* ((x-ext (interval-widen (ly:stencil-extent stencil X) padding))
         (y-ext (interval-widen (ly:stencil-extent stencil Y) padding))
         (y-rule (make-filled-box-stencil (cons 0 thickness) y-ext))
         (x-rule (make-filled-box-stencil stencil X)
                 y-rule)
         (x-rule (make-filled-box-stencil stencil X)
                 y-rule)))
(interval-widen x-ext thickness) (cons 0 thickness)))

;; (set! stencil (ly:stencil-combine-at-edge stencil X 1 y-rule padding))
(set! stencil (ly:stencil-combine-at-edge stencil X LEFT y-rule padding))
(set! stencil (ly:stencil-combine-at-edge stencil Y UP x-rule 0.0))
(set! stencil (ly:stencil-combine-at-edge stencil Y DOWN x-rule 0.0))

stencil)

% The corresponding markup command, based on the \box command defined
% in scm/define-markup-commands.scm

(define-markup-command (NWS-box layout props arg) (markup?)
  #:properties ((thickness 0.1) (font-size 0) (box-padding 0.2))
  "Draw a box round \var{arg}. Looks at \code{thickness},
  \code{box-padding} and \code{font-size} properties to determine line
  thickness and padding around the markup."
  (let ((pad (* (magstep font-size) box-padding))
        (m (interpret-markup layout props arg)))
    (NWS-box-stencil m thickness pad)))

% Test it:

\relative c' {
  c1:\markup { \NWS-box ABCD }
  c1:\markup { \NWS-box \note {4} #1.0 }
}

\[ \text{\texttt{ABCD}} \]

\textbf{Time signature in parentheses}

The time signature can be enclosed within parentheses.

\relative c' {
  \override Staff.TimeSignature.stencil = #(lambda (grob)
  (bracketify-stencil (ly:time-signature::print grob) Y 0.1 0.2 0.1))
  \time 2/4
  a4 b8 c
}

\[ \text{\texttt{ABCD}} \]

\textbf{Time signature in parentheses - method 3}

Another way to put the time signature in parenthesis

\relative c' {
  \override Staff.TimeSignature.stencil = #(lambda (grob)
  (parenthesize-stencil (ly:time-signature::print grob) 0.1 0.4 0.4 0.1 ))
  \time 2/4
  a4 b8 c
}

\[ \text{\texttt{ABCD}} \]
Indicazione di tempo che mostra solo il numeratore (invece della frazione)

Talvolta un’indicazione di tempo non deve mostrare la frazione intera (ad esempio 7/4), ma solo il numeratore (numero 7 in questo caso). Si può ottenere facilmente con \override Staff.TimeSignature.style = #'single-digit, che cambia lo stile in modo permanente. Con \revert Staff.TimeSignature.style, questa impostazione può essere annullata. Per applicare lo stile a cifra singola (single-digit) a una sola indicazione di tempo, si usa il comando \override preceduto da \once.

\relative c' { 
  \time 3/4 
c4 c c 
  \override Staff.TimeSignature.style = #'single-digit 
  \time 2/4 
c4 c 
  \time 3/4 
c4 c c 
  \revert Staff.TimeSignature.style 
  \time 2/4 
c4 c 
  \time 5/4 
c4 c c c c 
  \time 2/4 
c4 c 
}

Modificare l’aspetto degli abbellimenti di un intero brano

L’aspetto di tutte le espressioni contenute nei blocchi \grace di un brano può essere modificato con le funzioni add-grace-property e remove-grace-property. L’esempio seguente toglie la definizione della direzione di Stem nell’abbellimento, in modo che gli abbellimenti non siano sempre rivolti in su, e barra le teste di nota.

\relative c' { 
  \new Staff { 
    $$\{\text{remove-grace-property 'Voice 'Stem 'direction}\}$$ 
    $$\{\text{add-grace-property 'Voice 'NoteHead 'style 'cross}\}$$ 
    \new Voice { 

User defined time signatures

New time signature styles can be defined. The time signature in the second measure should be upside down in both staves.

```lisp
#(add-simple-time-signature-style 'topsy-turvy
  (lambda (fraction)
    (make-rotate-markup 180 (make-compound-meter-markup fraction))))
```

Using alternative flag styles

Alternative styles of flag on eighth and shorter notes can be displayed by overriding the stencil property of Flag. Valid values are modern-straight-flag, old-straight-flag and flat-flag.

```latex
\score { \relative c' { \time 2/4 \testnotes }
```
Usare il gambo barrato degli abbellimenti con le teste normali

Il gambo barrato presente nelle acciaccature può essere applicato in altre situazioni.

\relative c'' {
   \override Flag.stroke-style = "grace"
   c8 (d2) e8 (f4)
}

Usare le legature di valore con un arpeggio

Le legature di valore vengono usate talvolta per scrivere un arpeggio. In questo caso, le due note da legare devono non essere consecutive. Per ottenere tale risultato occorre impostare la proprietà tieWaitForNote su #t. Questa funzionalità serve anche a legare un tremolo a un accordo e in generale qualsiasi coppia di note consecutive.

\relative c' {
   \set tieWaitForNote = #t
   \grace { c16[ ~ e ~ g ] - } <c, e g>2
   \repeat tremolo 8 { c32 ~ c' - } <c c,>1
e8 ~ c ~ a ~ f ~ <e' c a f>2
   \tieUp
c8 ~ a
   \tieDown
   \tieDotted
}
g8 - c g2
Expressive marks

Sezione “Expressive marks” in Guida alla Notazione

Adding beams, slurs, ties etc. when using tuplet and non-tuplet rhythms

LilyPond syntax can involve many unusual placements for parentheses, brackets etc., which might sometimes have to be interleaved.

For example, when entering a manual beam, the left square bracket has to be placed after the starting note and its duration, not before. Similarly, the right square bracket should directly follow the note which is to be at the end of the requested beaming, even if this note happens to be inside a tuplet section.

This snippet demonstrates how to combine manual beaming, manual slurs, ties and phrasing slurs with tuplet sections (enclosed within curly braces).

```
{ r16[ g16 \tuplet 3/2 { r16 e'8 } ]
g16[ a \tuplet 3/2 { b d e' } ]
g8[ a \tuplet 3/2 { b d' } e' ] - }
\time 2/4
\tuplet 5/4 { e'32\( a b d' e' \) a'4.\}
}
```

Adding parentheses around an expressive mark or chordal note

The \parenthesize function is a special tweak that encloses objects in parentheses. The associated grob is Parenthood.

```
\relative c' {
  c2-\parenthesize ->
  \override Parentheses.padding = #0.1
  \override Parentheses.font-size = #-4
  <d \parenthesize f a>2
}
```

Aggiungere i segni di tempo per i glissandi lunghi

I battiti saltati nei glissandi molto lunghi vengono talvolta segnalati con delle indicazioni di tempo, che consistono solitamente in dei gambi privi di teste di nota. Questi gambi possono essere usati anche per contenere segni di espressione intermedi.

Se i gambi non si allineano bene al glissando, può essere necessario riposizionarli leggermente.

glissandoSkipOn = {

Cambiare la forma dei portamenti indeterminati verso il basso o verso l’alto

La proprietà `shortest-duration-space` può essere modificata per cambiare la forma dei portamenti indeterminati verso il basso o verso l’alto.

```latex
\relative c’’ { 
\override Score.SpacingSpanner.shortest-duration-space = #4.0
\c2-\bendAfter #5
\c2-\bendAfter #-4.75
\c2-\bendAfter #8.5
\c2-\bendAfter #-6
}
```

\[ < \]

\[ f > \]
Allineare le estremità delle forcelle alle direzioni di NoteColumn

Le estremità delle forcelle possono essere allineate alle direzioni LEFT, CENTER o RIGHT dei grob NoteColumn modificando la proprietà endpoint-alignments, che è costituita da una coppia di numeri che rappresentano le estremità sinistra e destra della forcella. endpoint-alignments devono essere direzioni (-1, 0 o 1). Valori diversi verranno trasformati emettendo un avviso. Non ha effetto quando l’estremità destra di una forcella termina su una pausa; in questo caso termina sempre sul margine sinistro della pausa.

```
\override Hairpin.endpoint-alignments = #'(1 . -1)
\override Hairpin.endpoint-alignments = #'(LEFT . CENTER)
```

Note brevi alternative

Le note brevi sono disponibili anche con due linee verticali su ciascun lato della testa invece di una sola e in stile barocco.

```
\relative c'' { \time 4/2 c'\breve | \override Staff.NoteHead.style = #'altdefault b\breve \override Staff.NoteHead.style = #'baroque b\breve \revert Staff.NoteHead.style a\breve }
```

Asymmetric slurs

Slurs can be made asymmetric to match an asymmetric pattern of notes better.

```
slurNotes = { d,8( a' d f a f' d, a) }
\relative c' { \stemDown \slurUp \slurNotes \once \override Slur.eccentricity = #3.0 \slurNotes }
```
Breathing signs

Breathing signs are available in different tastes: commas (default), ticks, vees and “railroad tracks” (caesura).

\new Staff \relative c' { 
  \key es \major 
  \time 3/4 
  % this bar contains no \breathe 
  << { g4 as g } \( \{ \es4 \bes \es \} \) >> |
  % Modern notation: 
  % by default, \breathe uses the rcomma, just as if saying:
  % \override BreathingSign.text = 
  % #(make-musicglyph-markup "scripts.rcomma")
  << { g4 as g } \( \{ \es4 \breathe \bes \es \} \) >> |

  % rvarcomma and lvarcomma are variations of the default rcomma 
  % and lcomma 
  % N.B.: must use Staff context here, since we start a Voice below
  \override Staff.BreathingSign.text = 
    \markup { \musicglyph "scripts.rvarcomma" }
  << { g4 as g } \( \{ \es4 \breathe \bes \es \} \) >> |

  % raltcomma and laltcomma are alternative variations of the 
  % default rcomma and lcomma
  \override Staff.BreathingSign.text = 
    \markup { \musicglyph "scripts.raltcomma" }
  << { g4 as g } \( \{ \es4 \breathe \bes \es \} \) >> |

  % vee
  \override BreathingSign.text = 
    \markup { \musicglyph "scripts.upbow" }
  es8[ d es f g] \breathe f |

  % caesura
  \override BreathingSign.text = 
    \markup { \musicglyph "scripts.caesura.curved" }
  es8[ d] \breathe es[ f g f] |
  es2 r4 \bar "||" 
}
Broken Crescendo Hairpin

In order to make parts of a crescendo hairpin invisible, the following method is used: A white rectangle is drawn on top of the respective part of the crescendo hairpin, making it invisible. The rectangle is defined as postscript code within a text markup.

The markup command `with-dimensions` tells LilyPond to consider only the bottom edge of the rectangle when spacing it against the hairpin. The property `staff-padding` prevents the rectangle from fitting between the hairpin and staff.

Make sure the hairpin is in a lower layer than the text markup to draw the rectangle over the hairpin.

```lilypond
\relative c' {
  <<
    { 
      \dynamicUp
      r2 r16 c'8. \pp r4 
    }
    \\ 
    { 
      \override DynamicLineSpanner.layer = #0 
      des,2\mf< - 
      \override TextScript.layer = #2 
      \once \override TextScript.staff-padding = #6 
      \once \override TextScript.vertical-skylines = #'( ) 
      des16 \markup \with-dimensions #'(2 . 7) #'(0 . 0) 
      \with-color \#white 
      \filled-box #'(2 . 7) #'(0 . 2) #0 
      r8. des4 - des16->\sff r8. 
    }
  >>
}
```

Caesura ("railtracks") with fermata

A caesura is sometimes denoted by a double “railtracks” breath mark with a fermata sign positioned above. This snippet shows an optically pleasing combination of railtracks and fermata.

```lilypond
\relative c'' {
  c2.
  % construct the symbol
  \override BreathingSign.text = \markup { 
    \override #(direction . 1) 
    \override #(baseline-skip . 1.8) 
    \dir-column { 
      \translate #'(0.155 . 0) 
      \center-align \musicglyph "scripts.caesura.curved"
      \center-align \musicglyph "scripts.ufermata"
  }
```
Center text below hairpin dynamics

This example provides a function to typeset a hairpin (de)crescendo with some additional text below it, such as “molto” or “poco”. The added text will change the direction according to the direction of the hairpin. The Hairpin is aligned to DynamicText.

The example also illustrates how to modify the way an object is normally printed, using some Scheme code.

\paper { tagline = ##f }

\hspace{4em}

hairpinWithCenteredText =
#(define-music-function (text) (markup?)
  #(\once \override Voice.Hairpin.after-line-breaking =
    #(lambda (grob)
      (let* ((stencil (ly:hairpin::print grob))
             (par-y (ly:grob-parent grob Y))
             (dir (ly:grob-property par-y 'direction))
             (staff-line-thickness
              (ly:output-def-lookup (ly:grob-layout grob) 'line-thickness))
             (new-stencil (ly:stencil-aligned-to
                           (ly:stencil-combine-at-edge
                            (ly:stencil-aligned-to stencil X CENTER) Y dir)
                           (ly:stencil-aligned-to
                            (grob::has-interface par-x
                             (grob-interpreter grob)
                             (make-fontsize-markup
                              (magnification->font-size
                               (+ (ly:staff-symbol-staff-space grob)
                                (ly:staff-line-thickness 2))
                              text)) X CENTER)))
             (staff-space (ly:output-def-lookup
                           (ly:grob-layout grob) 'staff-space))
             (par-x (ly:grob-parent grob X))
             (dyn-text (grob::has-interface par-x 'dynamic-text-interface))
             (dyn-text-stencil-x-length
              (if dyn-text
               (ly:image-stencil-sprite
class (ly:output-def-lookup (ly:grob-layout grob) 'image)
              X CENTER))))
      X LEFT)))
    (ly:image-stencil-sprite
class (ly:output-def-lookup (ly:grob-layout grob) 'image)
    X CENTER)
  )

\breathe c4
\% set the breathe mark back to normal
\revert BreathingSign.text
c2. \breathe c4
\bar "|."
Nascondere la linea di estensione per le dinamiche testuali

Il testo usato per i crescendo e i decrescendo può essere cambiato modificando le proprietà di contesto crescendoText e decrescendoText.

\relative c' {
  \set crescendoText = \markup { \italic { cresc. poco } }
  \set crescendoSpanner = #'text
  \override DynamicTextSpanner.style = #'dotted-line
  a2\< a
  a2 a
  a2 a\mf
}

Changing the appearance of a slur from solid to dotted or dashed

The appearance of slurs may be changed from solid to dotted or dashed.

\relative c' {
  c4( d e c)
  \slurDotted
  c4( d e c)
  \slurSolid
  c4( d e c)
  \slurDashed
  c4( d e c)
  \slurSolid
  c4( d e c)
}

Cambiare il simbolo del segno di respiro

Il glifo del respiro può essere modificato sovrascrivendo la proprietà text dell’oggetto di formattazione BreathingSign con qualsiasi testo incluso in un blocco markup.

\relative c'' {
  c2
  \override BreathingSign.text = \markup { \musicglyph "scripts.rvarcomma" }
  \breathe
  d2
}
Modifica del numero di punti di aumentazione per nota

Il numero di punti di aumentazione su una singola nota può essere modificato in modo indipendente dai punti posizionati dopo la nota.

\relative c' {  
  c4.. a16 r2 |  
  \override Dots.dot-count = #4  
  c4.. a16 r2 |  
  \override Dots.dot-count = #0  
  c4.. a16 r2 |  
  \revert Dots.dot-count  
  c4.. a16 r2 | 
}

Combining dynamics with markup texts

Some dynamics may involve text indications (such as “più forte” or “piano subito”). These can be produced using a \markup block.

\markup { \italic più \dynamic f } \layout { \ragged-right = ##f }

\relative c'' {  
  c2\f c-\plusF  
}

Glissando contemporaneo

Un glissando contemporaneo senza una nota finale può essere creato usando una nota nascosta e un tempo di cadenza.

\relative c'' {  
  \time 3/4  
  \override Glissando.style = #'zigzag  
  c4 c  
  \cadenzaOn  
  c4\glissando  
  \hideNotes  
  c,,4  
}
Controlling spanner visibility after a line break

The visibility of spanners which end on the first note following a line break is controlled by the after-line-breaking callback ly:spanner::kill-zero-spanned-time.

For objects such as glissandos and hairpins, the default behaviour is to hide the spanner after a break; disabling the callback will allow the left-broken span to be shown.

Conversely, spanners which are usually visible, such as text spans, can be hidden by enabling the callback.

\paper { ragged-right = ##t }

\relative c'' {
\override Hairpin.to-barline = ##f
\override Glissando.breakable = ##t
% show hairpin
\override Hairpin.after-line-breaking = ##t
% hide text span
\override TextSpanner.after-line-breaking =
  #ly:spanner::kill-zero-spanned-time
e2<\startTextSpan
% show glissando
\override Glissando.after-line-breaking = ##t
f2\glissando
\break
f,1!\stopTextSpan
}

Controllo dell’ordine verticale degli script

L’ordine verticale degli script è determinato dalla proprietà 'script-priority. Più il numero è piccolo, più sarà posto vicino alla nota. In questo esempio, il simbolo di diesis (oggetto TextScript) ha prima la priorità più bassa, dunque è posto più in basso nel primo esempio. Nel secondo, il trillo (oggetto Script) ha la priorità più bassa, quindi si trova all’interno. Quando due oggetti hanno la stessa priorità, l’ordine in cui sono inseriti determina quale viene prima.

\relative c''' {
Creare un gruppetto ritardato

Creare un gruppetto ritardato, dove la nota più bassa del gruppetto usa l’alterazione, richiede vari \override. La proprietà outside-staff-priority deve essere impostata su #f, perché altrimenti questa avrebbe la precedenza sulla proprietà avoid-slur. Cambiando la frazione 2/3 si aggiusta la posizione orizzontale.

\relative c' {
    \after 2*2/3 \turn c2( d4) r |
    \after 4 \turn c4.( d8)
    \after 4
    {\once \set suggestAccidentals = ##t
     \once \override AccidentalSuggestion.outside-staff-priority = ##f
     \once \override AccidentalSuggestion.avoid-slur = #'inside
     \once \override AccidentalSuggestion.font-size = -3
     \once \override AccidentalSuggestion.script-priority = -1
     \once \hideNotes
     cis8\turn \noBeam
     }
    d4.( e8)
}

Creare degli arpeggi che attraversano note appartenenti a voci diverse

Si può disegnare un arpeggio che attraversa delle note in voci diverse dello stesso rigo se si aggiunge l’incisore Span_arpeggio_engraver nel contesto Staff:

\new Staff \with { 
    \consists "Span_arpeggio_engraver"
}

\relative c' {
    \set Staff.connectArpeggios = ##t
    <<
        { <e' g>4\arpeggio <d f> <d f>2 }
Creare degli arpeggi che attraversano il rigo del pianoforte

In un rigo per pianoforte (PianoStaff), è possibile far sì che un arpeggio attraversi i righi impostando la proprietà PianoStaff.connectArpeggios.

\new PianoStaff \relative c' <<
 \set PianoStaff.connectArpeggios = ##t
 \new Staff {
  \clef bass
  \repeat unfold 4 {
    \arpeggio <c e>2 \arpeggio <d f>2
  }
}

Creare degli arpeggi che attraversano i righi in altri contesti

Si possono creare arpeggi che attraversano i righi in contesti diversi da GrandStaff, PianoStaff e StaffGroup se l’incisore Span_arpeggio_ engraver è incluso nel contesto Score.

\score {
 \new ChoirStaff {
  \set Score.connectArpeggios = ##t
  <<
    \new Voice \relative c' {
      \arpeggio <c e>2
      \arpeggio <d f>2
      \arpeggio <c e>1
    }
    \new Voice \relative c {
      \arpeggio <c e>2
      \arpeggio <d f>2
      \arpeggio <c e>1
    }
  }
}
Creating double-digit fingerings

Creating fingerings larger than 5 is possible.

```
\relative c' {
  c1-10
  c1-50
  c1-36
  c1-29
}
```

Creating "real" parenthesized dynamics

Although the easiest way to add parentheses to a dynamic mark is to use a \markup block, this method has a downside: the created objects will behave like text markups, and not like dynamics.

However, it is possible to create a similar object using the equivalent Scheme code (as described in the Notation Reference), combined with the make-dynamic-script function. This way, the markup will be regarded as a dynamic, and therefore will remain compatible with commands such as \dynamicUp or \dynamicDown.

```
paren = #(define-event-function (dyn) (ly:event?))
  (make-dynamic-script
    #:markup #:concat {
      \normal-text \italic \fontsize #2 {
        \pad-x #0.2 #(ly:music-property dyn 'text)
      }
    }
```

```
Creating slurs across voices

In some situations, it may be necessary to create slurs between notes from different voices. The solution is to add invisible notes to one of the voices, using \hideNotes.

This example is measure 235 of the Ciaconna from Bach’s 2nd Partita for solo violin, BWV 1004.

Creating text spanners

The \startTextSpan and \stopTextSpan commands allow the creation of text spanners as easily as pedal indications or octavations. Override some properties of the TextSpanner object to modify its output.
Estensore testuale della dinamica personalizzato

Funzioni postix per estensori testuali personalizzati del crescendo. Gli estensori devono iniziare sulla prima nota della misura; e bisogna usare «\mycresc», altrimenti l’inizio dell’estensore viene assegnato alla nota successiva.

\% Two functions for (de)crescendo spanners where you can explicitly
\% give the spanner text.
\mycresc =
\#{(define-music-function) (mymarkup) (markup?)
    (make-music 'CrescendoEvent
        'span-direction START
        'span-type 'text
        'span-text mymarkup))
mydecresc =
#{define-music-function (mymarkup) (markup?)
  (make-music 'DecrescendoEvent
    'span-direction START
    'span-type 'text
    'span-text mymarkup))

\relative c' {
  c4-\mycresc "custom cresc" c4 c4 c4 |
  c4 c4 c4 c4 |
  c4-\mydecresc "custom decresc" c4 c4 c4 |
  c4 c4\! c4 c4
}

cresc. - - - cresc. poco a poco ~ dim. decresc.

Glissandi can skip grobs
NoteColumn grobs can be skipped over by glissandi.
\relative c' {
  a2 \glissando
  \once \override NoteColumn.glissando-skip = ##t
  f''4 d,
}
Hairpins with different line styles

Hairpins can take any style from line-interface - dashed-line, dotted-line, line, trill or zigzag.

\relative c' {  
  c2< c!  
  \override Hairpin.style = #'dashed-line  
  c2< c!  
  \override Hairpin.style = #'dotted-line  
  c2< c!  
  \override Hairpin.style = #'line  
  c2< c!  
  \override Hairpin.style = #'trill  
  c2< c!  
  \override Hairpin.style = #'zigzag  
  c2< c!  
  \revert Hairpin.style  
  c2< c!  
}

Nascondere la linea di estensione per le dinamiche testuali

I cambi di dinamica in stile testuale (come cresc. e dim.) appaiono con una linea tratteggiata che mostra la loro estensione. Questa linea può essere soppressa nel modo seguente:

\relative c'' {  
  \override DynamicTextSpanner.style = #'none  
  \crescTextCresc  
  c1< | d | b | c!  
}

Horizontally aligning custom dynamics (e.g. "sempre pp", "piu f", "subito p")

Some dynamic expressions involve additional text, like “sempre pp”. Since dynamics are usually centered under the note, the \pp would be displayed way after the note it applies to.

To correctly align the “sempre pp” horizontally so that it is aligned as if it were only the \pp, there are several approaches:
• Simply use `\once\override DynamicText.X-offset = #-9.2` before the note with the dynamics to manually shift it to the correct position. Drawback: This has to be done manually each time you use that dynamic markup...

• Add some padding (`:hspace 7.1`) into the definition of your custom dynamic mark so that after LilyPond center-aligns it, it is already correctly aligned. Drawback: The padding really takes up that space and does not allow any other markup or dynamics to be shown in that position.

• Shift the dynamic script `\once\override ... X-offset = ...`. Drawback: `\once\override` is needed for every invocation!

• Set the dimensions of the additional text to 0 (using `#:with-dimensions '(0 . 0) '(0 . 0)`). Drawback: For LilyPond, “sempre” has no extent now. This means it might put other stuff there, causing collisions (which are not detected by LilyPond’s collision detection algorithm!). There also seems to be some spacing, so it is not exactly the same alignment as without the additional text.

• Add an explicit shift directly inside the scheme function for the dynamic script.

• Set an explicit alignment inside the dynamic script. By default, this won’t have any effect, only if one sets X-offset! Drawback: One needs to set `DynamicText.X-offset`, which will apply to all dynamic texts! Also, it is aligned at the right edge of the additional text, not at the center of `\pp`.

\paper {
  ragged-right = ##f
  indent = 2.5\cm
}

% Solution 1: Using a simple markup with a particular halign value
% Drawback: It’s a markup, not a dynamic command, so `\dynamicDown` etc. will have no effect
semppMarkup = `\markup { \halign #1.4 \italic "sempre" \dynamic "pp" }

% Solution 2: Using a dynamic script & shifting with
% `\once \override ...X-offset = ..`
% Drawback: `\once \override` needed for every invocation
semppK =
#(make-dynamic-script
  (markup #:line
    #:normal-text
    #:italic "sempre"
    #:dynamic "pp"))

% Solution 3: Padding the dynamic script so the center-alignment
% puts it at the correct position
% Drawback: the padding really reserves the space, nothing else can be there
semppT =
#(make-dynamic-script
  (markup #:line
    #:normal-text
    #:italic "sempre"
    #:dynamic "pp"
    #:hspace 7.1))

% Solution 4: Dynamic, setting the dimensions of the additional text to 0
Expressive marks

% Drawback: To lilypond "sempre" has no extent, so it might put
% other stuff there => collisions
% Drawback: Also, there seems to be some spacing, so it's not exactly the
% same alignment as without the additional text

semppM =
#(make-dynamic-script
  (markup #:line
    (#:with-dimensions '0 0)'0 0)
    #:right-align
    #:normal-text
    #:italic "sempre"
    #:dynamic "pp")))

% Drawback: To lilypond "sempre" has no extent, so it might put
% other stuff there => collisions
% Drawback: Also, there seems to be some spacing, so it's not exactly the
% same alignment as without the additional text

% Solution 5: Dynamic with explicit shifting inside the scheme function

semppG =
#(make-dynamic-script
  (markup #:hspace 0
    #:translate '(−18.85 . 0)
    #:line (#:normal-text
      #:italic "sempre"
      #:dynamic "pp")))

% Solution 5: Dynamic with explicit shifting inside the scheme function

% Solution 6: Dynamic with explicit alignment. This has only effect
% if one sets X-offset!
% Drawback: One needs to set DynamicText.X-offset!
% Drawback: Aligned at the right edge of the additional text, not at the center of pp

semppMII =
#(make-dynamic-script
  (markup #:line
    #:right-align
    #:normal-text
    #:italic "sempre"
    #:dynamic "pp")))

\new StaffGroup <<
\new Staff = "s" \with { instrumentName = \markup \column { Normal } } <<
  \relative c'' {
    \key es \major
    c4\pp c\p c c | c\ff c c\pp c
  }
>>
\new Staff = "sMarkup" \with {
  instrumentName = \markup \column { Normal markup } }
<<
  \relative c'' {
    \key es \major
    c4\semppMarkup c\p c c | c\ff c c\semppMarkup c
  }
>>
\new Staff = "sK" \with {

instrumentName = \markup \column { Explicit shifting }
}
<< \relative c'' {
  \key es \major
  \once \override DynamicText.X-offset = #-9.2
  c4\semppK c\p c c
  c4\ff c
  \once \override DynamicText.X-offset = #-9.2
  c4\semppK c
}
>> \new Staff = "sT" \with {
  instrumentName = \markup \column { Right padding }
}
<< \relative c'' {
  \key es \major
  c4\semppT c\p c c | c\ff c c\semppT c
}
>> \new Staff = "sM" \with {
  instrumentName = \markup \column { Set dimension "to zero" }
}
<< \relative c'' {
  \key es \major
  c4\semppM c\p c c | c\ff c c\semppM c
}
>> \new Staff = "sG" \with {
  instrumentName = \markup \column { Shift inside dynamics}
}
<< \relative c'' {
  \key es \major
  c4\semppG c\p c c | c\ff c c\semppG c
}
>> \new Staff = "sMII" \with {
  instrumentName = \markup \column { Alignment inside dynamics }
}
<< \relative c'' {
  \key es \major
  % Setting to ##f (false) gives the same result
  \override DynamicText.X-offset = #0
  c4\semppMII c\p c c | c\ff c c\semppMII c
}
>>
>>
\layout { \override Staff.InstrumentName.self-alignment-X = #LEFT }

Inserire una cesura

I segni di cesura possono essere creati sovrascrivendo la proprietà \text dell’oggetto BreathingSign. È disponibile anche un segno di cesura curvo.

\relative c'' { 
  \override BreathingSign.text = \markup { 
    \musicglyph "scripts.caesura.straight"
  }
  c8 e4. \breathe g8. e16 c4
  
  \override BreathingSign.text = \markup { 
    \musicglyph "scripts.caesura.curved"
  }
  g8 e’4. \breathe g8. e16 c4
}

Laissez vibrer ties

Laissez vibrer ties have a fixed size. Their formatting can be tuned using \tie-configuration.

\relative c' { 
  \laissezVibrer r <c e g>4\laissezVibrer r <c f g>\laissezVibrer r
Line arrows

Arrows can be applied to text-spanners and line-spanners (such as the Glissando).

```latex
\relative c'' { \override TextSpanner.bound-padding = #1.0
\override TextSpanner.style = #'line
\override TextSpanner.bound-details.right.arrow = ##t
\override TextSpanner.bound-details.left.text = #'"fof"
\override TextSpanner.bound-details.right.text = #'"gag"
\override TextSpanner.bound-details.right.padding = #0.6

\override TextSpanner.bound-details.right.stencil-align-dir-y = #CENTER
\override TextSpanner.bound-details.left.stencil-align-dir-y = #CENTER

\override Glissando.bound-details.right.arrow = ##t
\override Glissando.arrow-length = #0.5
\override Glissando.arrow-width = #0.25

a8\startTextSpan gis a4 b\glissando b, g'4 c\stopTextSpan c2 }
```

Legature di portamento con complesse strutture di tratteggio

Le legature di portamento possono avere schemi di tratteggio complessi definendo la proprietà `dash-definition`. `dash-definition` è una lista di `dash-elements`. Un `dash-element` è una lista di parametri che definiscono il comportamento del tratteggio per un segmento della legatura.

La legatura di portamento è definita come il parametro `t` della curva di bezier che va da 0 sul margine sinistro della legatura fino a 1 su quello destro. `dash-element` è una lista di (inizio-t fine-t frazione-trattino punto-trattino). La regione della legatura di portamento che va da inizio-t a fine-t avrà una frazione frazione-trattino di ogni punto-trattino nero.
punto-trattino viene definito in spazi rigo. frazione-trattino è impostato su 1 per una legatura di portamento continua.

\relative c' {
  \once \override 
  Slur.dash-definition = #'((0 0.3 0.1 0.75) 
    (0.3 0.6 1 1) 
    (0.65 1.0 0.4 0.75)) 

c4( d e f) 
\once \override 
  Slur.dash-definition = #'((0 0.25 1 1) 
    (0.3 0.7 0.4 0.75) 
    (0.75 1.0 1 1)) 

c4( d e f) 
}

\timesig.C44/clefs.G/noteheads.s2/noteheads.s2/noteheads.s2/noteheads.s2
\timesig.C44/clefs.G/noteheads.s0/scripts.stopped/timesig.C44
\timesig.C44/clefs.G/noteheads.s0/scripts.trill/timesig.C44

Modificare i valori predefiniti per le abbreviazioni delle articolazioni


\relative c'' { c1-+ }
dashPlus = \trill

\relative c'' { c1-+ }

Moving slur positions vertically

The vertical position of a slur can be adjusted using the positions property of Slur. The property has 2 parameters, the first referring to the left end of the slur and the second to the right. The values of the parameters are not used by LilyPond to make an exact movement of the slur - instead it selects what placement of the slur looks best, taking into account the parameter values. Positive values move the slur up, and are appropriate for notes with stems down. Negative values move downward slurs further down.

\relative c' {
  \stemDown
Spostare le estremità delle forcelle

Le estremità delle forcelle possono essere spostate in modo relativo alla loro posizione pre-definita (offset) impostando la proprietà `shorten-pair` dell’oggetto `Hairpin`. Valori positivi spostano le estremità a destra, valori negativi le spostano a sinistra. Diversamente dalla proprietà `minimum-length`, questa proprietà modifica solo l’aspetto della forcella; non cambia la spaziatura orizzontale (inclusa la posizione delle dinamiche confinanti). Questo metodo è quindi utile per ritoccare una forcella entro lo spazio ad essa allocato.

```latex
{ 
  c'1-\p< 
  c'2- c'!\ 
  \once \override Hairpin.shorten-pair = #'(2 . 2) 
  c'1-\p< 
  c'2- c'!\ 
  \once \override Hairpin.shorten-pair = #'(-2 . -2) 
  c'1-\p< 
  c'2- c'!\ 
  c'1-\p-\tweak shorten-pair #'(2 . 0)\p< 
  c'2- c'\fff 
}
```
Positioning arpeggios

If you need to extend or shorten an arpeggio, you can modify the upper and lower start positions independently.

\relative c' {
  \once \override Arpeggio.positions = #'(-5 . 0)
  \once \override Arpeggio.positions = #'(0 . 5)
  \once \override Arpeggio.positions = #'(-5 . 5)
}

Posizionare il testo a margine dentro le legature di portamento

I testi a margine devono avere la proprietà outside-staff-priority impostata su false per poter apparire dentro le legature di portamento.

\relative c'' {
  \override TextScript.avoid-slur = #'inside
  \override TextScript.outside-staff-priority = ##f
  c2(\markup { \hspace{-10} \natural } d4.) c8
}

Stampare le forcelle in vari stili

Il segno di dinamica della forcella può avere diversi stili

\relative c''' {
  \override Hairpin.stencil = #flared-hairpin
  a4\< a a a\f
  a4\p< a a a\ff
  a4\sfz< a a a!
  \override Hairpin.stencil = #constante-hairpin
  a4\< a a a\f
  a4\p< a a a\ff
  a4\sfz< a a a!
Forcelle con notazione al niente

Le forcelle di dinamica possono essere rappresentate con una punta tonda (noteazione “al niente”) impostando la proprietà circled-tip dell’oggetto Hairpin su #t.

\relative c' '{
  \override Hairpin.circled-tip = ##t
  c2< c\!
  c4> c< c2\!
}
Impostare il comportamento delle forcelle sulle stanghette

Se la nota che termina una forcella si trova sul primo battito di una battuta, la forcella si ferma prima della stanghetta che precede la nota. Si può controllare questo comportamento modificando la proprietà ‘to-barline’.

\relative c’' {  
  e4\< e2.  
  e1\!  
  \override Hairpin.to-barline = ##f  
  e4\< e2.  
  e1\!  
}

Impostare la lunghezza minima delle forcelle

Se le forcelle sono troppo corte, possono essere allungate modificando la proprietà minimum-length dell’oggetto Hairpin.

<<  
  {  
    \after 4 \< \after 2 \> \after 2. \! f'1  
    \override Hairpin.minimum-length = #8  
    \after 4 \< \after 2 \> \after 2. \! f'1  
  }  
  {  
    \repeat unfold 8 c'4  
  }  
>>
Showing the same articulation above and below a note or chord

By default, LilyPond does not allow the same articulation (e.g., an accent, a fermata, a flageolet, etc.) to be displayed above and below a note. For example, \c4_\fermata\fermata only shows a fermata below. The fermata above gets simply ignored.

However, one can stick scripts (just like fingerings) inside a chord, which means it is possible to have as many articulations as desired. This approach has the advantage that it ignores the stem and positions the articulation relative to the note head. This can be seen in the case of the flageolets in the snippet. To mimic the behaviour of scripts outside a chord, 'add-stem-support would be required.

The solution is thus to write the note as a chord and add the articulations inside of <...>, using the direction modifiers ^ and _ as appropriate.

\relative c' {  
<>"Wrong"  
c2_\fermata\fermata % The second fermata is ignored!  
<e d'>2\flageolet_\flageolet

\stopStaff s1 \startStaff

<>"Works if written inside a chord"  
<e_\flageolet d'\flageolet>2  
<e_\flageolet d'\flageolet>2  
<e_\flageolet\flageolet>2  
<e_\fermata\fermata>2
}

Wrong  Works if written inside a chord
\[\begin{array}{c}
\begin{array}{c}
\text{Wrong} \\
\text{Works if written inside a chord}
\end{array}
\end{array}\]

Snap-pizzicato or Bartok pizzicato

A snap-pizzicato (also known as “Bartok pizzicato”) is a “strong pizzicato where the string is plucked vertically by snapping and rebounds off the fingerboard of the instrument” (Wikipedia). It is denoted by a circle with a vertical line going from the center upwards outside the circle.

\relative c' {  
c4\snappizzicato  
< c' e g4\snappizzicato  
< c' e g4\snappizzicato  
< c, e g4_\snappizzicato
}

\[\begin{array}{c}
\begin{array}{c}
\text{Wrong} \\
\text{Works if written inside a chord}
\end{array}
\end{array}\]

Snap-pizzicato or Bartok pizzicato

A snap-pizzicato (also known as “Bartok pizzicato”) is a “strong pizzicato where the string is plucked vertically by snapping and rebounds off the fingerboard of the instrument” (Wikipedia). It is denoted by a circle with a vertical line going from the center upwards outside the circle.
Usare un segno di spunta come simbolo di respiro

La musica vocale e per fiati usa frequentemente il segno di spunta come segno di respiro.Questo indica un respiro che sottrae un po' di tempo alla nota precedente invece di prendere una piccola pausa, indicata dal segno di respiro rappresentato dalla virgola. Il segno può essere spostato un po' su per allontanarlo dal rigo.

\relative c' \{ 
\breathe
\override BreathingSign.Y-offset = #2.6
\override BreathingSign.text = \markup \musicglyph "scripts.tickmark"
\breathe
\}


Usare arpeggioBracket per rendere i divisi più visibili

Si può usare arpeggioBracket per indicare la divisione delle voci quando non ci sono gambi che forniscono questa informazione. Questo caso è frequente nella musica corale.

\score "english.ly"

\relative c' \{ 
\key a \major \time 2/2
<< \new Voice = "upper"
<< 
\voiceOne \arpeggioBracket
a2( b2
<b d>1\arpeggio)
<cs e>\arpeggio ~
<cs e>4
}
\addlyrics \lyricmode { A -- men. }
>> \new Voice = "lower"
\voiceTwo a1 ~
a a ~
a4 \bar "."
}
Uso delle doppie legature di portamento per gli accordi legati

Alcuni compositori scrivono due legature di portamento per indicare gli accordi legati. Si può ottenere questo risultato impostando doubleSlurs.

\relative c' {
  \set doubleSlurs = ##t
  <c e>4 ( <d f> <c e> <d f>)
}

Uso della proprietà whiteout

Qualsiasi oggetto grafico può essere posizionato sopra uno sfondo bianco per mascherare parti degli oggetti che si trovano sotto. Ciò può essere utile per migliorare l’aspetto delle collisioni in situazioni complesse in cui il riposizionamento degli oggetti è troppo difficile. Bisogna impostare esplicitamente la proprietà layer (livello) per controllare quali oggetti debbano essere mascherati dallo sfondo bianco.

In questo esempio la collisione della legatura di valore con l’indicazione di tempo viene migliorata mascherando la parte della legatura che incrocia l’indicazione di tempo impostando la proprietà whiteout di TimeSignature. Per farlo si sposta TimeSignature su un livello superiore a Tie, che viene lasciato al livello predefinito 1; e StaffSymbol viene spostato su un livello superiore a TimeSignature in modo che non venga mascherato.

{\override Score.StaffSymbol.layer = #4
  \override Staff.TimeSignature.layer = #3
  b'2 b'~
  \once \override Staff.TimeSignature.whiteout = ##t
  \time 3/4
  b' r4
}

Vertical line as a baroque articulation mark

This short vertical line placed above the note is commonly used in baroque music. Its meaning can vary, but generally indicates notes that should be played with more “weight”. The following example demonstrates how to achieve such a notation.

upline =
Vertically aligning dynamics across multiple notes

Dynamics that occur at, begin on, or end on the same note will be vertically aligned. To ensure that dynamics are aligned when they do not occur on the same note, increase the staff-padding property of the DynamicLineSpanner object.

```
\relative c' {
  \override DynamicLineSpanner.staff-padding = #4
  c2 \p \ mf
  g2 <\ b4 \> c! 
}
```

```
\relative c' {
  \tweak stencil
    #(lambda (grob)
        (grob-interpret-markup grob #{ \markup \draw-line #'(0 . 1) #}))
  \stopped

  a'4^-\upline a (c d')_-\upline
}
```

```
\relative c' {
  a'4^-\upline a (c d')_-\upline
}
```

```
\relative c' {
  \override DynamicLineSpanner.staff-padding = #4
  c2 \p \ mf
  g2 <\ b4 \> c!
}
```

```
\relative c' {
  \tweak stencil
    #(lambda (grob)
        (grob-interpret-markup grob #{ \markup \draw-line #'(0 . 1) #}))
  \stopped

  a'4^-\upline a (c d')_-\upline
}
```

```
\relative c' {
  a'4^-\upline a (c d')_-\upline
}
```
Repeats

Sezione “Repeats” in Guida alla Notazione

Aggiungere le parentesi delle volte a altri righi

L’incisore Volta_engraver risiede nel contesto Score, quindi le parentesi delle ripetizioni appaiono di norma soltanto sul rigo superiore. Questo comportamento può essere modificato aggiungendo l’incisore Volta_engraver al contesto Staff in cui si desidera far apparire le parentesi; si veda anche il frammento “Volta multirigo”.

<<
\new Staff \{ \repeat volta 2 \{ c'1 \} \alternative \{ c' \} \}
\new Staff \{ \repeat volta 2 \{ c'1 \} \alternative \{ c' \} \}
\new Staff \with \{ \consists "Volta_engraver" \} \{ c'2 g' e' a' \}
\new Staff \{ \repeat volta 2 \{ c'1 \} \alternative \{ c' \} \}
>>

Centered measure numbers

Scores of large ensemble works often have bar numbers placed beneath the system, centered horizontally on the measure’s extent. This snippet shows how the Measure_counter_engraver may be used to simulate this notational practice. Here, the engraver has been added to a Dynamics context.

This snippet presents a legacy method: starting from LilyPond 2.23.3, \set Score.centerBarNumbers = ##t is enough.

\layout {
\context {
\ Dynamics
\consists #Measure_counter_engraver
\override MeasureCounter.direction = #DOWN
\override MeasureCounter.font-encoding = #'latin1
\override MeasureCounter.font-shape = #'italic
% to control the distance of the Dynamics context from the staff:
\override VerticalAxisGroup.nonstaff-relatedstaff-spacing.padding = #2
}
\context {
\ Score
}
\remove "Bar_number_engraver"

pattern = \repeat unfold 7 { c'4 d' e' f' }

\new StaffGroup <<
  \new Staff {
    \pattern
  }
  \new Staff {
    \pattern
  }
  \new Dynamics {
    \startMeasureCount s1*7
    \stopMeasureCount
  }
>>

Changing the default bar lines

Default bar lines can be changed when re-defined in a score context.

% http://lsr.di.unimi.it/LSR/Item?id=964
%=> http://lilypond.1069038.n5.nabble.com/Changing-the-default-end-repeat-bracket-tc169357.html

\layout {
  \context {
    \Score
      \default measureBarType = "!"
      \default startRepeatBarType = "[]:
      \default endRepeatBarType = "[]]
      \default doubleRepeatBarType = "[]][:
    \startMeasureCount s1*7
    \stopMeasureCount
  }

  % example:
  {   c'1
      \repeat volta 2 { \repeat unfold 2 c' }
}
Tremoli attraverso i rigi

Dato che \repeat tremolo si aspetta esattamente due argomenti musicali per i tremoli di accordi, la nota o l’accordo che cambiano rigo in un tremolo che attraversa i rigi devono essere posti tra parentesi graffe insieme al comando \change Staff.

\new PianoStaff <<
\new Staff = "up" \relative c'' {
  \key a \major
  \time 3/8
  s4.
}
\new Staff = "down" \relative c'' {
  \key a \major
  \time 3/8
  \voiceOne
  \repeat tremolo 6 {
    <a e'>32
    { 
      \change Staff = "up"
      \voiceTwo
      <cis a' dis>32
    }
  }
}>>
Engraving tremolos with floating beams

If a tremolo’s total duration is less than a quarter-note, or exactly a half-note, or between a half-note and a whole-note, it is normally typeset with all beams touching the stems. Certain engraving styles typeset some of these beams as centered floating beams that do not touch the stems. The number of floating beams in this type of tremolo is controlled with the 'gap-count property of the Beam object, and the size of the gaps between beams and stems is set with the 'gap property.

\relative c' \{ 
\repeat tremolo 8 { a32 f }  
\override Beam.gap-count = #1 
\repeat tremolo 8 { a32 f }  
\override Beam.gap-count = #2 
\repeat tremolo 8 { a32 f }  
\override Beam.gap-count = #3 
\repeat tremolo 8 { a32 f }  
\override Beam.gap-count = #3 
\override Beam.gap = #1.33 
\repeat tremolo 8 { a32 f }  
\override Beam.gap = #1 
\repeat tremolo 8 { a32 f }  
\override Beam.gap = #0.67 
\repeat tremolo 8 { a32 f }  
\override Beam.gap = #0.33 
\repeat tremolo 8 { a32 f }  
\}

Ripetizioni con segni di percentuale isolati

Si possono stampare anche segni di percentuale isolati.

\makePercent = 
#(define-music-function (note) (ly:music?) 
 "Make a percent repeat the same length as NOTE."  
(make-music 'PercentEvent  
'length (ly:music-length note)))

\relative c' \{ 
\makePercent s1 
\}

Measure counter

This snippet provides a workaround for emitting measure counters using transparent percent repeats.
Numbering groups of measures

This snippet demonstrates the use of the Measure_counter_engraver to number groups of successive measures. Any stretch of measures may be numbered, whether consisting of repetitions or not.

The engraver must be added to the appropriate context. Here, a Staff context is used; another possibility is a Dynamics context.

The counter is begun with \startMeasureCount and ended with \stopMeasureCount. Numbering will start by default with 1, but this behavior may be modified by overriding the count-from property.

When a measure extends across a line break, the number will appear twice, the second time in parentheses.

\layout {
  \context {
    \Staff
      \startMeasureCount
      \repeat unfold 7 {
        c'4 d' e' f'
      }
      \stopMeasureCount
      \bar "||"
      g'4 f' e' d'
      \override Staff.MeasureCounter.count-from = #2
      \startMeasureCount
      \repeat unfold 5 {
        g'4 f' e' d'
      }
  }
}
Visibilità del conto della ripetizione con segno percentuale

I contatori della ripetizione con segno percentuale possono essere mostrati a intervalli regolari impostando la proprietà di contesto repeatCountVisibility.

Contatore della ripetizione con segno percentuale

Le ripetizioni di misura che hanno più di due ripetizioni possono avere un contatore se si cambia la proprietà opportuna, come mostra questo esempio:
Positioning segno and coda (with line break)

If you want to place an exiting segno sign and add text like “D.S. al Coda” next to it where usually the staff lines are you can use this snippet. The coda will resume in a new line. There is a variation documented in this snippet, where the coda will remain on the same line.

```latex
\relative c'' {
  c4 c c c c c c c c c c c
  \repeat segno 2 {
    c4 c c c c c c c c
  }\alternative {
    \volta 1 {
      c4 c c c c c c c c c c c
      % If you don't use \break at Coda, use \noBreak here
      % and after \bar "" below.
      \noBreak
      \section % double bar line
      \cadenzaOn % pause bar count
      \stopStaff % remove staff lines
      % Increasing the unfold counter will expand the staff-free space
      \repeat unfold 6 {
        s1
        \bar ""
      }
    }
    % Place JumpScript where the staff would normally be.
    \once \override Score.JumpScript.outside-staff-priority = ##f
    \once \override Score.JumpScript.Y-offset = 0
    \startStaff % resume bar count
    \cadenzaOff % show staff lines again
  }\sectionLabel "Coda"
  % Show Coda on a new line
  \break
  \repeat unfold 8 { c4 c c c }
  \fine
} % D.S. $% al \phi$
e poi la Coda
Impostare la doppia ripetizione predefinita per le volte

Esistono tre diversi stili di doppie ripetizioni per le volte, che si possono impostare con \doubleRepeatBarType.

```latex
\relative c'' { 
  \repeat volta 2 { c1 }
  \set Score.doubleRepeatBarType = "\":...:"
  \repeat volta 2 { c1 }
  \set Score.doubleRepeatBarType = "\":|.|:"
  \repeat volta 2 { c1 }
  \set Score.doubleRepeatBarType = "\":|.|:"
  \repeat volta 2 { c1 }
}
```

Accorciare le parentesi delle volte

Per impostazione predefinita, le parentesi delle volte si estendono per tutta l’alternativa, ma si possono accorciare impostando voltaSpannerDuration. Nell’esempio seguente, la parentesi dura una misura, che ha una durata di 3/4.

```latex
\relative c'' { 
  \time 3/4
  c4 c c
  \set Score.voltaSpannerDuration = #(ly:make-moment 3/4)
  \repeat volta 5 { d4 d d }
  \alternative { 
    { e4 e e 
      f4 f f 
    } 
    { g4 g g }
  }
}
```

Volta sotto gli accordi

Aggiungendo l’incisore Volta_engraver al rigo, è possibile inserire le volte sotto gli accordi.

```latex
\score { 
  <<
```
Volta multi staff

By adding the Volta_engraver to the relevant staff, volte can be put over staves other than the topmost one in a score.

```
\chords {
  c1
  c1
}
\new Staff \with {
  \consists "Volta_engraver"
}
{
  \repeat volta 2 { c'1 }
  \alternative { c' }
}
>>
\layout {
  \context {
    \Score
    \remove "Volta_engraver"
  }
}
}

Volta multi staff

By adding the Volta_engraver to the relevant staff, volte can be put over staves other than the topmost one in a score.

```
voltaMusic = \relative c'' {
  \repeat volta 2 {
    c1
  }
  \alternative {
    d1
e1
  }
}
```

```
<<
\new StaffGroup <<
  \new Staff \ voltaMusic
  \new Staff \ voltaMusic
>>
\new StaffGroup <<
  \new Staff \with { \consists "Volta_engraver" }
  \ voltaMusic
  \new Staff \ voltaMusic
>>
>>
Volta text markup using repeatCommands

Though volta are best specified using \repeat volta, the context property repeatCommands must be used in cases where the volta text needs more advanced formatting with \markup.

Since repeatCommands takes a list, the simplest method of including markup is to use an identifier for the text and embed it in the command list using the Scheme syntax #(list (list 'volta textIdentifier)). Start- and end-repeat commands can be added as separate list elements:

\[\texttt{voltaAdLib} = \\texttt{\textbackslash markup} \{ 1. 2. 3... \texttt{\textbackslash text \textbackslash italic} \{ \texttt{ad lib.} \} \}\]

\[\texttt{relative \textbackslash c''} \{\]
\[\texttt{c1}\]
\[\texttt{\set \texttt{Score.repeatCommands} = #(list (list 'volta \texttt{voltaAdLib} 'start-repeat))}\]
\[\texttt{c4 b d e}\]
\[\texttt{\set \texttt{Score.repeatCommands} = #'(volta #f) (volta "4.") end-repeat)}\]
\[\texttt{f1}\]
\[\texttt{\set \texttt{Score.repeatCommands} = #'(volta #f)}\]
\[\}\]

\[\texttt{\set \texttt{Score.repeatCommands} = #'(volta #f)}\]}
Simultaneous notes

Sezione “Simultaneous notes” in Guida alla Notazione

Voci ulteriori per evitare le collisioni

In alcuni casi di musica polifonica complessa sono necessarie delle voci ulteriori per evitare le collisioni tra note. Se servono più di quattro voci parallele, si possono aggiungere altre voci definendo una variabile con la funzione Scheme function context-spec-music.

```scheme
voiceFive = #(context-spec-music (make-voice-props-set 4) 'Voice)
```

```
relative c'' {
  time 3/4
  key d \minor
  partial 2
  **new** Voice { 
    voiceOne
    a4. a8
    e'4 e4. e8
    f4 d4. c8
  }
  **new** Voice { 
    voiceTwo
    d,2
    d4 cis2
    d4 bes2
  }
  **new** Voice { 
    voiceThree
    f'2
    bes4 a2
    a4 s2
  }
  **new** Voice { 
    voiceFive
    s2
    g4 g2
    f4 f2
  }
}
```

Changing a single note’s size in a chord

Individual note heads in a chord can be modified with the \tweak command inside a chord, by altering the font-size property.
Inside the chord (within the brackets < >), before the note to be altered, place the \tweak command, followed by font-size and define the proper size like #-2 (a tiny note head).

\relative c' {
    <\tweak font-size #+2 c e g c
    \tweak font-size #-2 e>1
    \markup { A tiny e }_\markup { A big c }
}

\clefs.G /timesig.C44

Modificare le indicazioni testuali di partCombine

Quando si usa la funzionalità di combinazione automatica delle parti, si può modificare il testo delle sezioni soliste e dell’unisono:

\new Staff <<
\set Staff.soloText = "girl"
\set Staff.soloIIText = "boy"
\set Staff.aDueText = "together"
\partCombine
\relative c'' {
    g4 g r r
    a2 g
}
\relative c'' {
    r4 r a( b)
    a2 g
}
>>

Clusters

Clusters are a device to denote that a complete range of notes is to be played.

\fragment = \relative c' {
    c4 f <e d'>4
    <g a>8 <e a> a4 c2 <d b>4
    e2 c
}

<<
\new Staff \fragment
\new Staff \makeClusters \fragment
>>
Simultaneous notes

Lo strumento di unione delle parti (il comando \partCombine) permette di combinare varie parti sullo stesso rigo. Indicazioni testuali come “solo” e “a2” sono aggiunte automaticamente; per toglierle basta impostare la proprietà printPartCombineTexts su f. Per le partiture vocali (inni), non c’è bisogno di aggiungere i testi “solo/a2”, quindi dovrebbero essere disattivati. Tuttavia potrebbe convenire non usarlo se c’è una qualche parte solista, perché non verrebbe indicata. In tali casi è preferibile usare la notazione polifonica normale.

Questo frammento illustra i tre modi con cui due parti possono essere stampate su uno stesso rigo: normale polifonia, \partCombine senza testo e \partCombine con testo.

%% Combining pedal notes with clef changes

\musicUp = \relative c'' { \time 4/4 a4 c4. (g8) a4 | g4 e' g, (a8 b) | c b a2. }
\musicDown = \relative c'' { g4 e4. (d8) c4 | r2 g'4 (f8 e) | d2 \stemDown a }

\score {
<< \new Staff \with { instrumentName = "Standard polyphony" } << \musicUp \musicDown >>
\new Staff \with {
    instrumentName = "PartCombine without text"
    printPartCombineTexts = ##f
}
\partCombine \musicUp \musicDown
\new Staff \with { instrumentName = "PartCombine with text" }
\partCombine \musicUp \musicDown >>
\layout {
    indent = 6.0\cm
    \context {

Displaying complex chords

Here is a way to display a chord where the same note is played twice with different accidentals.

```latex
\fixA = \{
    \override Stem.length = #11
\}
\fixB = \{
    \override NoteHead.X-offset = #1.7
    \override Stem.length = #7
    \override Stem.rotation = #'(45 0 0)
    \override Stem.extra-offset = #'(-0.1 . -0.2)
    \override Flag.style = #'no-flag
    \override Accidental.extra-offset = #'(4 . -.1)
\}
```

```latex
\relative c' {
    << \fixA <b d!>8 > \ { \voiceThree \fixB dis } >> s
}
```

Forzare lo spostamento orizzontale delle note

Quando il motore tipografico non riesce a risolvere una situazione, si può usare la sintassi che sovrascrive le decisioni tipografiche. L’unità di misura usata è lo spazio del rigo.

```latex
\relative c' \<
    \{ <d g>2 <d g>
\}
```
Making an object invisible with the 'transparent property

Setting the transparent property will cause an object to be printed in “invisible ink”: the object is not printed, but all its other behavior is retained. The object still takes up space, it takes part in collisions, and slurs, ties and beams can be attached to it.

This snippet demonstrates how to connect different voices using ties. Normally, ties only connect two notes in the same voice. By introducing a tie in a different voice, and blanking the first up-stem in that voice, the tie appears to cross voices.

```
relative { 
  time 2/4 
  \relative { 
    { 
      \once \hide Stem 
      \once \override Stem.length = #8 
      b'8 ~ 8\noBeam 
      \once \hide Stem 
      \once \override Stem.length = #8 
      g8 ~ 8\noBeam 
    } 
    \relative { 
      b8 g g e 
    } 
    \relative { 
      b'8 g g e 
    } 
  } 
} 
```

Spostare le note puntate in polifonia

Quando una nota puntata della voce più alta viene spostata per evitare una collisione con una nota di un’altra voce, il comportamento predefinito è spostare la nota più alta a destra. Tale comportamento può essere modificato tramite la proprietà prefer-dotted-right di NoteCollision.

```
new Staff \relative c' << 
{ 
```
### Supressing warnings for clashing note columns

If notes from two voices with stems in the same direction are placed at the same position, and both voices have no shift or the same shift specified, the error message ‘warning: ignoring too many clashing note columns’ will appear when compiling the LilyPond file. This message can be suppressed by setting the ‘ignore-collision property of the NoteColumn object to #t. Please note that this does not just suppress warnings but stops LilyPond trying to resolve collisions at all and so may have unintended results unless used with care.

```lilypond
\override NoteColumn.ignore-collision = ##t
```

### Two \partCombine pairs on one staff

The `\partCombine` function takes two music expressions each containing a part, and distributes them among four Voices named “two”, “one”, “solo”, and “chords” depending on when and how the parts are merged into a common voice. The voices output from `\partCombine` can have their layout properties adjusted in the usual way. Here we define extensions of `\partCombine` to make it easier to put four voices on a staff.

- **soprano** = { d'4 | cis' b e' d'8 cis' | cis'2 b }
- **alto** = { fis4 | e8 fis gis ais b4 b | b ais fis2 }
- **tenor** = { a8 b | cis' dis' e'4 b8 cis' d'4 | gis cis' dis'2 }
- **bass** = { fis8 gis | a4 gis g fis | eis fis b,2 }

```lilypond
\new Staff <<
\key b\minor
\clef alto
\partial 4
```
\transpose b b'
\partCombineUp \soprano \alto
\partCombineDown \tenor \bass

\layout {
  \context {
    \Staff
    \accepts "VoiceBox"
  }
  \context {
    \name "VoiceBox"
    \type "Engraver_group"
    \defaultchild "Voice"
    \accepts "Voice"
    \accepts "NullVoice"
  }
}

customPartCombineUp = 
#(define-music-function (partOne partTwo)
  (ly:music? ly:music?)
  "Take the music in @var{partOne} and @var{partTwo} and return
  a @code{VoiceBox} named @q{Up} containing @code{Voice}s
  that contain @var{partOne} and @var{partTwo} merged into one
  voice where feasible. This variant sets the default voicing
  in the output to use upward stems."
  #{
    \new VoiceBox = "Up" <<
    \context Voice = "one" { \voiceOne }
    \context Voice = "two" { \voiceThree }
    \context Voice = "shared" { \voiceOne }
    \context Voice = "solo" { \voiceOne }
    \context NullVoice = "null" {}
    \partCombine #partOne #partTwo
    >>
  })
}

customPartCombineDown = #
  (define-music-function (partOne partTwo)
   (ly:music? ly:music?)
   "Take the music in @var{partOne} and @var{partTwo} and return
   a @code{VoiceBox} named @q{Down} containing @code{Voice}s
   that contain @var{partOne} and @var{partTwo} merged into one
   voice where feasible. This variant sets the default voicing
   in the output to use downward stems."
   #{
     \new VoiceBox = "Down" <<
     \set VoiceBox.soloText =="#Solo III"
     \set VoiceBox.soloIIText =="#Solo IV"
     \context Voice = "one" { \voiceFour }
     \context Voice = "two" { \voiceTwo }
   })
\context Voice ="shared" { \voiceFour }
\context Voice ="solo" { \voiceFour }
\context NullVoice = "null" {}
\partCombine #partOne #partTwo

soprano = { d'4 | cis' b e' d'8 cis' | cis'2 b }
alto = { fis4 | e8 fis gis ais b4 b | b ais fis2 }
tenor = { a8 b | cis' dis' e'4 b8 cis' d'4 | gis cis' dis'2 }
bass = { fis8 gis | a4 gis g fis | eis fis b,2 }

\new Staff <<
  \key b\minor
  \clef alto
  \partial 4
  \transpose b b'
  \customPartCombineUp soprano \alto
  \customPartCombineDown tenor \bass
>>
Staff notation

Sezione “Staff notation” in *Guida alla Notazione*

Un ambitus per voce

L’ambitus può essere specificato per voce. In tal caso occorre spostarlo manualmente per evitare collisioni.

```latex
\new Staff <<
\new Voice \with {
   \consists "Ambitus_engraver"
} \relative c'' {
   \override Ambitus.X-offset = #2.0
   \voiceOne
   c4 a d e
   f1
}
\new Voice \with {
   \consists "Ambitus_engraver"
} \relative c' {
   \voiceTwo
   es4 f g as
   b1
}
>>
```

Adding an extra staff

An extra staff can be added (possibly temporarily) after the start of a piece.

```latex
\score {
   \new Staff \relative c'' {
      c1 | c | c | c | c
   }
\new StaffGroup \relative c'' {
   \new Staff {
      c1 | c
      <
      c1 | d
   }
   \new Staff {
      \once \omit Staff.TimeSignature
      c1 | b
   }
   >>
   c1
```
Adding an extra staff at a line break

When adding a new staff at a line break, some extra space is unfortunately added at the end of the line before the break (to fit in a key signature change, which will never be printed anyway). The workaround is to add a setting of `Staff.explicitKeySignatureVisibility` as is shown in the example.

\score {
  \new StaffGroup \relative c'' {
    \new Staff
    \key f \major
    c1 c"Unwanted extra space" \break
    << { c1 | c }
      \new Staff {
      \key f \major
      \once \omit Staff.TimeSignature
      c1 | c
    }
    >>
    c1 | c"Fixed here" \break
    << { c1 | c }
    \new Staff {
    \once \set Staff.explicitKeySignatureVisibility = #end-of-line-invisible
      \key f \major
      \once \omit Staff.TimeSignature
      c1 | c
    }
    >>
  }
}
Adding indicators to staves which get split after a break

This snippet defines the \splitStaffBarLine, convUpStaffBarLine and convDownStaffBarLine commands. These add arrows at a bar line, to denote that several voices sharing a staff will each continue on a staff of their own in the next system, or that voices split in this way recombine.

#(define-markup-command (arrow-at-angle layout props angle-deg length fill) (number? number? boolean?)
  (let* (  
    (PI-OVER-180 (/ (atan 1 1) 34))  
    (degrees->radians (lambda (degrees) (* degrees PI-OVER-180))))  
    (angle-rad (degrees->radians angle-deg))  
    (target-x (* length (cos angle-rad)))  
    (target-y (* length (sin angle-rad))))
  (interpret-markup layout props
    markup
    #:translate (cons (/ target-x 2) (/ target-y 2))  
    #:rotate angle-deg  
    #:translate (cons (/ length -2) 0)  
    #:concat (#:draw-line (cons length 0)  
      #:arrow-head X RIGHT fill))))))

\combine
\arrow-at-angle #45 #(*sqrt 8) ##t  
\arrow-at-angle #45 #(*sqrt 8) ##t
}

\once \override Staff.BarLine.stencil =  
#(lambda (grob)  
  (ly:stencil-combine-at-edge  
    (ly:bar-line::print grob)  
    X RIGHT  
    (grob-interpret-markup grob splitStaffBarLineMarkup)
convDownStaffBarLine = {
  \once \override Staff.BarLine.stencil =
  \(\lambda\) (grob)
    (ly:stencil-combine-at-edge
     (ly:bar-line::print grob)
     X RIGHT
     (grob-interpret-markup grob \#{
       \markup\with-dimensions \'(0 . 0) \'(0 . 0) \{\translate \'(0 . -.13)\arrow-at-angle -45 \(\sqrt{8}\) \t
       \}#})
     0))
  \break
}

convUpStaffBarLine = {
  \once \override Staff.BarLine.stencil =
  \(\lambda\) (grob)
    (ly:stencil-combine-at-edge
     (ly:bar-line::print grob)
     X RIGHT
     (grob-interpret-markup grob \#{
       \markup\with-dimensions \'(0 . 0) \'(0 . 0) \{\translate \'(0 . .14)\arrow-at-angle 45 \(\sqrt{8}\) \t
       \}#})
     0))
  \break
}

\paper {
  \ragged-right = \t\t
  \short-indent = 10\mm
}

separateSopranos = {
  \set Staff.instrumentName = "AI AII"
  \set Staff.shortInstrumentName = "AI AII"
  \splitStaffBarLine
  \change Staff = "up"
}

convSopranos = {
  \convDownStaffBarLine
  \change Staff = "shared"
  \set Staff.instrumentName = "S A"
  \set Staff.shortInstrumentName = "S A"
}

si = {
\voiceOne
\repeat unfold 4 f''2
\separateSopranos
\repeat unfold 4 g''2
\convSopranos
\repeat unfold 4 c''2
}
sII = {
s1*2
\voiceTwo
\change Staff = "up"
\repeat unfold 4 d''2
}
aI = {
\voiceTwo
\repeat unfold 4 a'2
\voiceOne
\repeat unfold 4 b'2
\convUpStaffBarLine
\voiceTwo
\repeat unfold 4 g'2
}
aII = {
 s1*2
 \voiceTwo
 \repeat unfold 4 g'2
}
ten = {
 \voiceOne
 \repeat unfold 4 c'2
 \repeat unfold 4 d'2
 \repeat unfold 4 c'2
}
bas = {
 \voiceTwo
 \repeat unfold 4 f2
 \repeat unfold 4 g2
 \repeat unfold 4 c2


\score {
<<
 \new ChoirStaff <<
 \new Staff = up \with {
 instrumentName = "SI SII"
 shortInstrumentName = "SI SII"
 } { s1*4
 }

 \new Staff = shared \with {
 instrumentName = "S A"
}
\new Voice = sopI \sI
\new Voice = sopII \sII
\new Voice = altI \aI
\new Voice = altII \aII
\new Lyrics \with {
  alignBelowContext = up
}
\lyricsto sopII \{ e f g h \}
\new Lyrics \lyricsto altI \{ a b c d e f g h i j k l \}
\new Staff = men \with {
  instrumentName = "T B"
  shortInstrumentName = "T B"
}
\clef F
\new Voice = ten \ten
\new Voice = bas \bas
\lyricsto bas \{ a b c d e f g h i j k l \}
\Staff \RemoveEmptyStaves
  \override VerticalAxisGroup.remove-first = ##t
}
Aggiungere citazioni orchestrali a una partitura vocale

L’esempio seguente mostra un approccio per simplificare l’aggiunta di citazioni orchestrali a una riduzione per pianoforte di una partitura vocale. La funzione musicale \cueWhile prende quattro argomenti: la musica da cui prendere la citazione, come è definita da \addQuote, il nome da inserire prima delle notine, poi o \#UP o \#DOWN per specificare o \voiceOne col nome sopra il rigo o \voiceTwo col nome sotto il rigo, e infine la musica per pianoforte che deve apparire in parallelo alle notine. Il nome dello strumento citato è posto a sinistra delle notine. Molti passaggi possono essere citati, ma non possono sovrapporsi l’un l’altro nel tempo.

\cueWhile =
#{(define-music-function
  (instrument name dir music)
  (string? string? ly:dir? ly:music?)
  #{
    \cueDuring $instrument #dir { 
      \once \override TextScript.self-alignment-X = #RIGHT
      \once \override TextScript.direction = $dir
      <>-\markup { \tiny #name }
      $music
      }
    }
  })

flute = \relative c' { 
  \transposition c'
  s4 s4 e g
}
\addQuote "flute" { \flute }
clarinet = \relative c' { 
  \transposition bes
  fis4 d d c
}
\addQuote "clarinet" { \clarinet }

singer = \relative c'' { c4. g8 g4 bes4 }
words = \lyricmode { here's the lyrics }

pianoRH = \relative c' { 
  \transposition c
  \cueWhile "clarinet" "Clar." #DOWN { c4. g8 }
  \cueWhile "flute" "Flute" #UP { g4 bes4 }
}
pianoLH = \relative c { c4 <c' e> e, <g c> }

\score { 
  \new Staff { 
    \new Voice = "singer" { 
      \singer
    }
  }
  \new Lyrics { 
    \lyricsto "singer"
    \words
  }
  \new PianoStaff <<
    \new Staff { 
      \new Voice { 
        \pianoRH
      }
    }
    \new Staff { 
      \clef "bass"
      \pianoLH
    }
  >>
  >>
}
Aggiungere i segni di tempo per i glissandi lunghi

I battiti saltati nei glissandi molto lunghi vengono talvolta segnalati con delle indicazioni di tempo, che consistono solitamente in dei gambi privi di teste di nota. Questi gambi possono essere usati anche per contenere segni di espressione intermedi.

Se i gambi non si allineano bene al glissando, può essere necessario riposizionarli leggermente.

```latex
\glissandoSkipOn = {
  \override NoteColumn.glissando-skip = ##t
  \hide NoteHead
  \override NoteHead.no-ledgers = ##t
}\glissandoSkipOff = {
  \revert NoteColumn.glissando-skip
  \undo \hide NoteHead
  \revert NoteHead.no-ledgers
}\relative c' {  
  r8 f8\glissando  
  \glissandoSkipOn  
  f4 g a a8\noBeam  
  \glissandoSkipOff  
  a8  

  r8 f8\glissando  
  \glissandoSkipOn  
  g4 a8  
  \glissandoSkipOff  
  a8  |

  r4 f\glissando \<  
  \glissandoSkipOn  
  a4\f \>  
  \glissandoSkipOff  
b8\! r  |
}
```
Numeri di battuta alternativi

Si possono impostare due metodi alternativi di numerazione della battuta, utili specialmente per le ripetizioni.

\relative c'\{
  \set Score.alternativeNumberingStyle = #'numbers
  \repeat volta 3 \{ \c4 \d \e \f | \}
  \alternative \{
    \{ \c4 \d \e \f | \c2 \d \break \}
    \{ \f4 \g \a \b | \f4 \g \a \b \| \f2 \a | \break \}
    \{ \c4 \d \e \f | \c2 \d \}
  \}
  \c1 \break

  \set Score.alternativeNumberingStyle = #'numbers-with-letters
  \repeat volta 3 \{ \c, \d \e \f | \}
  \alternative \{
    \{ \c4 \d \e \f | \c2 \d \break \}
    \{ \f4 \g \a \b | \f4 \g \a \b \| \f2 \a | \break \}
    \{ \c4 \d \e \f | \c2 \d \}
  \}
  \c1
\}
Ambitus dopo armatura di chiave

Per impostazione predefinita, gli ambitus sono posizionati a sinistra della chiave. La funzione \ambitusAfter permette di cambiare questo posizionamento. La sintassi è \ambitusAfter grob-interface (vedi Graphical Object Interfaces (http://lilypond.org/doc/v2.22/Documentation/internals/graphical-object-interfaces) per un elenco dei possibili valori per grob-interface.)

Un caso d’uso comune è il posizionamento dell’ambitus tra l’armatura di chiave e l’indicazione di tempo.

\new Staff \with {
  \consists Ambitus_engraver
} \relative {
  \ambitusAfter key-signature
  \key d \major
  es'8 g bes cis d2
}

Centered measure numbers

Scores of large ensemble works often have bar numbers placed beneath the system, centered horizontally on the measure’s extent. This snippet shows how the Measure_counter_engraver may be used to simulate this notational practice. Here, the engraver has been added to a Dynamics context.

This snippet presents a legacy method: starting from LilyPond 2.23.3, \set Score.centerBarNumbers = ##t is enough.

\layout {
  \context {
    \Dynamics
    \consists #Measure_counter_engraver
    \override MeasureCounter.direction = #DOWN
    \override MeasureCounter.font-encoding = #'latin1
    \override MeasureCounter.font-shape = #'italic
    % to control the distance of the Dynamics context from the staff:
    \override VerticalAxisGroup.nonstaff-relatedstaff-spacing.padding = #2
  }
  \context {
    \Score
    \remove "Bar_number_engraver"
  }
}

pattern = \repeat unfold 7 { c'4 d' e' f' }
Changing the default bar lines

Default bar lines can be changed when re-defined in a score context.

\% http://lsr.di.unimi.it/LSR/Item?id=964
\%\%=> http://lilypond.1069038.n5.nabble.com/Changing-the-default-end-repeat-bracket-tc169357.html

\layout {
  \context {
    \Score
    \% Changing the defaults from engraver-init.ly
    measureBarType = #"!"
    startRepeatBarType = #"[|
    endRepeatBarType = #"|]
    doubleRepeatBarType = #"|][|
  }
}
\%\% example:
{
c'1
\repeat volta 2 { \repeat unfold 2 c' }
\repeat volta 2 { \repeat unfold 2 c' }
\alternative {
  { c' }
  \% v2.18 workaround
  \once\override \Score.VoltaBracket.shorten-pair = #'(1 . -1)
}
Changing the number of lines in a staff

The number of lines in a staff may changed by overriding the StaffSymbol property line-count.

```plaintext
upper = \relative c' \{ 
c4 d e f
}\}

lower = \relative c \{
\clef bass
\score { 
\context PianoStaff <<
   \new Staff {
      \upper
   } 
\new Staff { 
   \override Staff.StaffSymbol.line-count = #4
   \lower
   } >> 
}\}
```

Changing the staff size

Though the simplest way to resize staves is to use `#(set-global-staff-size xx)`, an individual staff’s size can be changed by scaling the properties 'staff-space and fontSize.'

```plaintext
<< 
\new Staff { 
   \relative c' { 
      \dynamicDown 
      c8\ff c c c c c c c
   }
```

![Staff notation 140](image)
Creating blank staves

To create blank staves, generate empty measures then remove the Bar_number_engraver from the Score context, and the Time_signature_engraver, Clef_engraver and Bar_engraver from the Staff context.

#(set-global-staff-size 20)

\score {
  \new Staff \with {
    fontSize = #-3
    \override StaffSymbol.staff-space = #(magstep -3)
  }
  \clef bass
  c8 c c c c\f c c c
}

% uncomment these lines for "letter" size
%
\paper {
  #(set-paper-size "letter")
  ragged-last-bottom = ##f
  line-width = 7.5\in
  left-margin = 0.5\in
  bottom-margin = 0.25\in
}
\topmargin = 0.25\in
}
\%

\%
uncomment these lines for "A4" size
%
\paper {
  #(set-paper-size "a4")
  ragged-last-bottom = ##f
  line-width = 180
  left-margin = 15
  bottom-margin = 10
  top-margin = 10
}
\%

Creating custom key signatures

LilyPond supports custom key signatures. In this example, print for D minor with an extended range of printed flats.

```lilypond
\new Staff \with { 
  \override StaffSymbol.line-count = #8 
  \override KeySignature.flat-positions = #'((-7 . 6)) 
  \override KeyCancellation.flat-positions = #'((-7 . 6)) 
  \% presumably sharps are also printed in both octaves 
  \override KeySignature.sharp-positions = #'((-6 . 7)) 
  \override KeyCancellation.sharp-positions = #'((-6 . 7))

  \override Clef.stencil = # ( 
    (lambda (grob) (grob-interpret-markup grob 
    #{( 
      \musicglyph "clefs.C"
      \translate #'(-3 . -2)
      \musicglyph "clefs.F"
    )})) 
    clefPosition = #3 
    middleCPosition = #3 
    middleCClefPosition = #3 
  )}

  { 
    key d\minor 
    f bes, f bes, 
  }

  \noteheads.s2/clefs.C/clefs.z/accidentals.flat/timesig.C44
```

Creating double-digit fingerings

Creating fingerings larger than 5 is possible.

```lilypond
\relative c' { 
  c1-10 
  c1-50 
  c1-36 
  c1-29 
}

\noteheads.s0/fattened.one/fattened.zero 
\noteheads.s0 /fattened.three/fattened.six 
\noteheads.s0/timesig.C44/clefs.G 
```

Cross staff stems

This snippet shows the use of the Span.stem. engraver and \crossStaff to connect stems across staves automatically.

The stem length need not be specified, as the variable distance between noteheads and staves is calculated automatically.
Mostrare la parentesi anche se c’è un solo rigo nel sistema

Se c’è un solo rigo in uno dei tipi di rigo ChoirStaff o StaffGroup, la parentesi e la stanghetta iniziale non appaiono. Si può modificare questo comportamento predefinito sovrascrivendo collapse-height e impostando un valore inferiore al numero di linee del rigo.

Nei contesti PianoStaff e GrandStaff, dove i sistemi iniziano con una parentesi graffa invece di una parentesi quadra, occorre impostare un’altra proprietà, come si vede nel secondo sistema dell’esempio.

```
\layout {
  \context {
    PianoStaff
    \consists "Span_stem_engraver"
  }

  \new PianoStaff <<
      \new Staff {
        \override StaffGroup.SystemStartBracket.collapse-height = #4
        \override Score.SystemStartBar.collapse-height = #4
        <b d'>4 r d'16> e'8. g8 r!\
        e'8 f' g'4 e'2
      }
      \new Staff {
        clef bass
        voiceOne
        \autoBeamOff
        \crossStaff { <e g>4 e, g16 a8. c8} d
        \autoBeamOn
        g8 f g4 c2
      }
    >>
  }
}
```
Extending a TrillSpanner

For TrillSpanner, the minimum-length property becomes effective only if the set-spacing-rods procedure is called explicitly.

To do this, the springs-and-rods property should be set to ly:spanner::set-spacing-rods.

\relative c' { 
\key c\minor
\time 2/4
\once\override TrillSpanner.minimum-length = \##15
\once\override TrillSpanner.springs-and-rods = \#ly:spanner::set-spacing-rods
\afterGrace es4
\startTrillSpan { d16[(\stopTrillSpan es)]}
\c{\c' g es c g' es d}
\hideNotes\c8
}

Estendere i glissandi sulle volte delle ripetizioni

Un glissando che si estende in vari blocchi \alternative può essere simulato aggiungendo all’inizio di ogni blocco \alternative una nota di abbellimento nascosta da cui inizia un glissando. La nota di abbellimento deve avere la stessa altezza della nota da cui parte il glissando iniziale. In questo frammento si usa una funzione musicale che prende come argomento l’altezza della nota di abbellimento.

Attenzione: nella musica polifonica la nota di abbellimento deve avere una nota di abbellimento corrispondente in tutte le altre voci.

\repeatGliss = #(!define-music-function (grace)
(ly:pitch?))
\#{
  \% the next two lines ensure the glissando is long enough
  \% to be visible
  \once \override Glissando.springs-and-rods = #ly:spanner::set-spacing-rods
  \once \override Glissando.minimum-length = #3.5
  \once \hideNotes
  \grace $\text{\textgreek{\texteuro}}$ \text{\textgreek{\textdollar}} \text{\textgreek{\textdollar}} \text{\textgreek{\textdollar}} \text{\textgreek{\textdollar}}
#

\score {
  \relative c'' {
    \repeat volta 3 { c4 d e f\glissando }
    \alternative {
      { g2 d }
      { \repeatGliss f g2 e }
      { \repeatGliss f e2 d }
    }
  }
}

music = \relative c' {
  \voiceOne
  \repeat volta 2 {
    g a b c\glissando
  }
  \alternative {
    { d1 }
    { \repeatGliss c \once \omit StringNumber e1\2 }
  }
}

\score {
  \new StaffGroup <<
    \new Staff <<
      \new Voice { \clef "$\text{G}_8$" \music }
    >>
  \new TabStaff <<
    \new TabVoice { \clef "moderntab" \music }
  >>
}
Flat Ties

The function takes the default Tie.stencil as an argument, calculating the result relying on the extents of this default.

Further tweaking is possible by overriding Tie.details.height-limit or with \shape. It’s also possible to change the custom-definition on the fly.

%(http://lsr.di.unimi.it/LSR/Item?id=1031

#(define ((flared-tie coords) grob)
  (define (pair-to-list pair)
    (list (car pair) (cdr pair)))
  (define (normalize-coords goods x y dir)
    (map
      (lambda (coord)
        ;(coord-scale coord (cons x (* y dir)))
        (cons (* x (car coord)) (* y dir (cdr coord))))
    goods))
  (define (my-c-p-s points thick)
    (make-connected-path-stencil
      points
      thick
      1.0
      1.0
      #f
      #f))

;; outer let to trigger suicide
(let ((sten (ly:tie::print grob)))
  (if (grob::is-live? grob)
    (let* ((layout (ly:grob-layout grob))
      (line-thickness (ly:output-def-lookup layout 'line-thickness))
      (thickness (ly:grob-property grob 'thickness 0.1))
      (used-thick (* line-thickness thickness))
      (dir (ly:grob-property grob 'direction))
      (xex (ly:stencil-extent sten X))
      (yex (ly:stencil-extent sten Y))
      (lenx (interval-length xex))
      (leny (interval-length yex))
      (xtrans (car xex))
      (ytrans (if (> dir 0) (car yex) (cdr yex))))
(uplist
   (map pair-to-list
         (normalize-coords coords lenx (* leny 2) dir))))

(ly:stencil-translate
   (my-c-p-s uplist used-thick)
   (cons xtrans ytrans))'
)

#(define flare-tie
   (flared-tie '((0 . 0) (0.1 . 0.2) (0.9 . 0.2) (1.0 . 0.0))))

\layout {
   \context {
      \Voice
      \override Tie.stencil = #flare-tie
   }
}

\paper {ragged-right = ##f}

\relative c' {
   a4-a
   \override Tie.height-limit = 4
   a'4-a
   a'4-a
   <a, , c e a c e a c e> q

   \break
   a4-a
   \once \override Tie.details.height-limit = 14
   a4-a

   \break
   a4-a
   \once \override Tie.details.height-limit = 0.5
   a4-a

   \break
   a4-a
   \shape #'((0 . 0) (0 . 0.4) (0 . 0.4) (0 . 0)) Tie
   a4-a

   \break
   a4-a
   \once \override Tie.stencil =
      #(flared-tie '((0 . 0) (0.1 . 0.4) (0.9 . 0.4) (1.0 . 0.0)))
   a4-a
\texttt{\texttt{Staff notation}}

\begin{verbatim}
a4-a
\once \override Tie.stencil =
   #(flared-tie '((0 . 0)(0.06 . 0.1) (0.94 . 0.1) (1.0 . 0.0)))
a4-a
\}
\end{verbatim}

\begin{figure}
\centering
\includegraphics[width=\textwidth]{example.png}
\caption{Example music staff notation with flared ties.}
\end{figure}

\textbf{Forcing measure width to adapt to MetronomeMark’s width}

By default, metronome marks do not influence horizontal spacing. This can be solved through a simple override, as shown in the second half of the example.

\begin{verbatim}
example = {
   \tempo "Allegro"
   R1*6
   \tempo "Rall."
   R1*2
   \tempo "A tempo"
   R1*8
}

{\
   \compressMMRests {
      \example
      R1
      R1
      \override Score.MetronomeMark.extra-spacing-width = #'(-3 . 0)
      \example
   }\
}
\end{verbatim}
Glissandi can skip grobs

NoteColumn grobs can be skipped over by glissandi.

```
\relative c' {
  a2 \glissando
  \once \override NoteColumn.glissando-skip = ##t
  f''4 d,
}
```

Incipit

Quando si trascrive musica mensurale, un incipit all’inizio del brano è utile per indicare il tempo e l’armatura di chiave originali. I musicisti oggi sono abituati alle stanghette, ma queste non erano note all’epoca della musica mensurale. Come compromesso, spesso le stanghette vengono poste tra i righi, uno stile di formattazione chiamato mensurstriche.

%% With 2.23. this throws:
%% programming error: Loose column does not have right side to attach to.
%% Likely "Hidden BarLine during note yields programming error"
%% https://gitlab.com/lilypond/lilypond/-/issues/4084
%% --Harm

```lilypond
% A short excerpt from the Jubilate Deo by Orlande de Lassus

global = {
  \set Score.skipBars = ##t
  \key g \major
  \time 4/4

  \% the actual music
  \skip 1*8

  \% let finis bar go through all staves
  \override Staff.BarLine.transparent = ##f

  \% finis bar
```
\bar "|." }

discantusIncipit = {
  \clef "neomensural-c1"
  \key f \major
  \time 2/2
  c''1.
}

discantusNotes = {
  \transpose c' c'' {
    \clef "treble"
    d'2. d'4 |
    b e' d'2 |
    c'4 e'4. ( d'8 c' b |
    a4) b a2 |
    b4. ( c'8 d'4) c'4 |
    \once \hide NoteHead
    c'1 |
    b\breve |
  }
}

discantusLyrics = \lyricmode {
  Ju -- bi -- la -- te De -- o,
  om -- nis ter -- ra, -- om-
  "..."
  -us.
}

altusIncipit = {
  \clef "neomensural-c3"
  \key f \major
  \time 2/2
  r1 f'1.
}

altusNotes = {
  \transpose c' c'' {
    \clef "treble"
    r2 g2. e4 fis g |
    a2 g4 e |
    fis g4. ( fis16 e fis4) |
    g1 |
    \once \hide NoteHead
    g1 |
    g\breve |
  }
}

altusLyrics = \lyricmode {

Ju -- bi -- la -- te
De -- o, om -- nis ter -- ra,
"..."
-us.

\[\text{tenorIncipit} = \{
  \text{\textbackslash clef} "neomensural-c4"
  \text{\textbackslash key} f \text{\textbackslash major}
  \text{\textbackslash time} 2/2
  \text{r\longa}
  \text{r\breve}
  \text{r1 c'1.}
  \}\]

\[\text{tenorNotes} = \{
  \text{\textbackslash transpose} c' c' \{ 
    \text{\textbackslash clef} "treble_8"
    \text{R1 |}
    \text{R1 |}
    \text{R1 |}
    \% two measures
    \text{r2 d'2. d'4 b e' |}
    \text{\once \textbackslash hide NoteHead}
    \text{e'1 |}
    \text{d'\breve |}
  \}\}
  \}\]

\[\text{tenorLyrics} = \text{\textbackslash lyricmode} \{
  \text{Ju -- bi -- la -- te}
  "..."
  -us.\}

\[\text{bassusIncipit} = \{
  \text{\textbackslash clef} "mensural-f"
  \text{\textbackslash key} f \text{\textbackslash major}
  \text{\textbackslash time} 2/2
  \text{r\maxima}
  \text{f1.}
  \}\]

\[\text{bassusNotes} = \{
  \text{\textbackslash transpose} c' c' \{ 
    \text{\textbackslash clef} "bass"
    \text{R1 |}
    \text{R1 |}
    \text{R1 |}
    \text{R1 |}
    \text{g2. e4 |}
    \text{\once \textbackslash hide NoteHead}
  \}\}
\score {

new StaffGroup = choirStaff <<
new Voice = "discantusNotes" <<
set Staff.instrumentName = "Discantus"
incipit discantusIncipit
global
discantusNotes
>>
new Lyrics \lyricsto discantusNotes { \discantusLyrics }
new Voice = "altusNotes" <<
set Staff.instrumentName = "Altus"
global
incipit altusIncipit
altusNotes
>>
new Lyrics \lyricsto altusNotes { \altusLyrics }
new Voice = "tenorNotes" <<
set Staff.instrumentName = "Tenor"
global
incipit tenorIncipit
tenorNotes
>>
new Lyrics \lyricsto tenorNotes { \tenorLyrics }
new Voice = "bassusNotes" <<
set Staff.instrumentName = "Bassus"
global
incipit bassusIncipit
bassusNotes
>>
new Lyrics \lyricsto bassusNotes { \bassusLyrics }
>
>
layout {
context {
Score
% no bar lines in staves or lyrics
hide BarLine
}
% the next two instructions keep the lyrics between the bar lines
context {
}

e1 |
g\breve |
}
\Lyrics
\consists "Bar_ engraver"
\consists "Separating_line_group_ engraver"
}
\context {
  \Voice
  \%% no slurs
  \hide Slur
  \%% Comment in the below "\remove" command to allow line
  \%% breaking also at those bar lines where a note overlaps
  \%% into the next measure. The command is commented out in this
  \%% short example score, but especially for large scores, you
  \%% will typically yield better line breaking and thus improve
  \%% overall spacing if you comment in the following command.
  \%\remove "Forbid_line_break_ engraver"
}
\indent = 6\,cm
\incipit-width = 4\,cm
}
Inserting score fragments above a staff, as markups

The \markup command is quite versatile. In this snippet, it contains a \score block instead of texts or marks.

```
tuning = \markup { 
    \score { 
        \new Staff \with { \remove "Time_signature_engraver" } 
        \clef bass 
        \<c, g, d g>1 
        \layout { \ragged-right = ##t \indent = 0\cm } 
    } 
}
```

```
\header { 
    title = "Solo Cello Suites" 
    subtitle = "Suite IV" 
    subsubtitle = \markup { Originalstimmung: \raise #0.5 \tuning } 
}
```

```
\relative c' { 
    \time 4/8 
    \tuplet 3/2 \{ c8 d e \} \tuplet 3/2 \{ c d e \} 
    \tuplet 3/2 \{ c8 d e \} \tuplet 3/2 \{ c d e \} 
    g8 a g a 
    g8 a g a 
}
```

Solo Cello Suites
Suite IV
Originalstimmung: \begin{figure}[h]
\begin{center}
\begin{tabular}{c}
\begin{music}
\staves[4] \times\times \times
\end{music}
\end{tabular}
\end{center}
\end{figure}

Let TabStaff print the topmost string at bottom

In tablatures usually the first string is printed topmost. If you want to have it at the bottom change the stringOneTopmost-context-property. For a context-wide setting this could be done in layout as well.

```
\layout { 
    \context { 
        \Score 
        \stringOneTopmost = ##f 
    } 
}
```
Letter tablature formatting

Tablature can be formatted using letters instead of numbers.

\[ \text{music} = \text{relative } c \{ \]
\[ \quad \text{c4 d e f} \]
\[ \quad \text{g4 a b c} \]
\[ \quad \text{d4 e f g} \]
\[ \} \]

\[ << \]
\[ \text{new Staff} \{ \text{\clef G_8} \text{ \relative c} \} \]
\[ \text{new TabStaff} \{ \]
\[ \quad \text{\set Score.stringOneTopmost = ##f} \]
\[ \quad \text{\set TabStaff.tablatureFormat = #fret-letter-tablature-format} \]
\[ \} \]
\[ >> \]
Lasciare che i glissandi vadano a capo

Per permettere a un glissando di andare a capo se capita su un’interruzione di riga, si impostano le proprietà `breakable` e `after-line-breaking` su `##t`:

```latex
\makeatletter
\glissandoSkipOn = {
  \override NoteColumn.glissando-skip = ##t
  \hide NoteHead
  \override NoteHead.no-ledgers = ##t
}\makeatother

```

Rendere alcune linee del rigo più spesse delle altre

In ambito didattico può essere utile rendere più spesso una linea del rigo (per esempio, la linea centrale, o per sottolineare la linea della chiave di Sol). Per farlo si possono aggiungere altre linee e posizionarle molto vicino alla linea che deve essere evidenziata, usando la proprietà `line-positions` dell’oggetto `StaffSymbol`.

```latex
{\relative c'' {
  \override Glissando.breakable = ##t
  \override Glissando.after-line-breaking = ##t
  f1\glissando |
  \break
  a4 r2. |
  f1\glissando \
  \once \glissandoSkipOn
  \break
  a2 a4 r4 |}
}
\override Staff.StaffSymbol.line-positions =
#'(4 -2 -0.2 0 0.2 2 4)
d'4 e' f' g'

Measure counter
This snippet provides a workaround for emitting measure counters using transparent percent repeats.
<<
\context Voice = "foo" {
  \clef bass
  c4 r g r
c4 r g r
c4 r g r
} \context Voice = "foo" {
  \set countPercentRepeats = ##t
  \hide PercentRepeat
  \override PercentRepeatCounter.staff-padding = #1
  \repeat percent 4 { s1 }
}
>>

Formattazione mensurale (stanghette tra i righi)
La formattazione mensurale, in cui le stanghette non appaiono sui righi ma nello spazio tra i righi, si può ottenere usando StaffGroup al posto di ChoirStaff. La stanghetta sui righi viene nascosta con \hide.
\layout {
  \context {
    \Staff
      measureBarType = "-span"
  }
}

music = \fixed c'' {
  c1
d2 section e2
  f1 \fine
}

\new StaffGroup <<
Modificare l’inclinazione dell’estensore dell’ottava
È possibile cambiare l’inclinazione dell’estensore dell’ottava.

```latex
\relative c' {
  \override Staff.OttavaBracket.stencil = #ly:line-spanner::print
  \override Staff.OttavaBracket.bound-details = #"((left . ((Y . 0)
    (attach-dir . ,LEFT)
    (padding . 0)
    (stencil-align-dir-y . ,CENTER))
  (right . ((Y . 5.0); Change the number here
    (padding . 0)
    (attach-dir . ,RIGHT)
    (text . ,(make-draw-dashed-line-markup
      (cons 0 -1.2))))))
  \override Staff.OttavaBracket.left-bound-info = #ly:horizontal-line-spanner::calc-left-bound-info-and-text
  \override Staff.OttavaBracket.right-bound-info = #ly:horizontal-line-spanner::calc-right-bound-info
  \ottava #1
c1
c''''1
}
```

Annidare i righi
Si può usare la proprietà `systemStartDelimiterHierarchy` per creare gruppi di righi annidati più complessi. Il comando `\set StaffGroup.systemStartDelimiterHierarchy` prende come argomento una lista alfabetica dell’insieme di righi prodotti. Prima di ogni rigo si può assegnare un delimitatore di inizio del sistema. Deve essere racchiuso tra parentesi e collega tutti i righi compresi tra le parentesi. Gli elementi nella lista possono essere omessi, ma la prima parentesi quadra collega sempre tutti i righi. Le possibilità sono `SystemStartBar`, `SystemStartBracket`, `SystemStartBrace` e `SystemStartSquare`.

```latex
\new StaffGroup
```
Armature di chiave non tradizionali

Il comando \key comunemente usato imposta la proprietà keyAlterations del contesto Staff. Per creare armature di chiave non standard, tale proprietà va impostata esplicitamente.

Il formato di questo comando è una lista:
\set Staff.keyAlterations = #`(((ottava . grado) . alterazione) ((ottava . grado) . alterazione) ...)

dove, per ogni elemento della lista, ottava indica l’ottava (0 è l’ottava dal Do centrale al Si precedente), grado indica la nota all’interno dell’ottava (0 significa Do e 6 significa Si) e alterazione può essere ,SHARP ,FLAT ,DOUBLE-SHARP etc.

Altrimenti, usando il formato breve per ogni elemento della lista, (grado . alterazione), ciò indica che la stessa alterazione deve essere presente in tutte le ottave. Per le scale microtonalidove un “diesis” non è 100 centesimi, alterazione si riferisce alla proporzione di un duecentesimo di tono intero.

\include "arabic.ly"
\relative do' {
  \set Staff.keyAlterations = #`((0 ,SEMI-FLAT)
  (1 ,SEMI-FLAT)
  (2 ,FLAT)
  (5 ,FLAT)
  (6 ,SEMI-FLAT))
\set Staff.extraNatural = ##f
\re reb \dwn reb resd
dod dob dosd \dwn dob |
dobsb dodsd do do |
}

Numbering groups of measures

This snippet demonstrates the use of the Measure_counter_engraver to number groups of successive measures. Any stretch of measures may be numbered, whether consisting of repetitions or not.

The engraver must be added to the appropriate context. Here, a Staff context is used; another possibility is a Dynamics context.

The counter is begun with \startMeasureCount and ended with \stopMeasureCount. Numbering will start by default with 1, but this behavior may be modified by overriding the count-from property.

When a measure extends across a line break, the number will appear twice, the second time in parentheses.

\layout {
\context {
\Staff
\consists #Measure_counter_engraver
}
}

\new \Staff {
\startMeasureCount
\repeat unfold 7 {
  c'4 d' e' f'
}
\stopMeasureCount
\bar "||"
g'4 f' e' d'
\override \Staff.MeasureCounter.count-from = #2
\startMeasureCount
\repeat unfold 5 {
  g'4 f' e' d'
}
g'4 f'
\bar ""
\break
e'4 d'
\repeat unfold 7 {
  g'4 f' e' d'
}
\stopMeasureCount
}
Modello per orchestra, coro e pianoforte

Questo modello mostra come usare i contesti annidati StaffGroup e GrandStaff per creare sottogruppi degli strumenti dello stesso tipo. Mostra anche come usare \transpose in modo che le variabili mantengano la musica per gli strumenti traspositori nell’intonazione reale.

#(set-global-staff-size 17)
\paper{
  indent = 3.0cm % add space for instrumentName
  short-indent = 1.5cm % add less space for shortInstrumentName
}

fluteMusic = \relative c' { \key g \major g'1 b }

% Pitches as written on a manuscript for Clarinet in A
% are transposed to concert pitch.

clarinetMusic = \transpose c' a
  \relative c'' { \key bes \major bes1 d }

trumpetMusic = \relative c { \key g \major g''1 b }

% Key signature is often omitted for horns

hornMusic = \transpose c' f
  \relative c { d'1 fis }

percussionMusic = \relative c { \key g \major g1 b }

sopranoMusic = \relative c' { \key g \major g'1 b }

sopranoLyrics = \lyricmode { Lyr -- ics }

altoIMusic = \relative c' { \key g \major g'1 b }

altoIIIMusic = \relative c' { \key g \major g'1 b }

altoILyrics = \sopranoLyrics

altoIIILyrics = \lyricmode { Ah -- ah }
tenorMusic = \relative c' { \clef "treble_8" \key g \major g1 b }

tenorLyrics = \sopranoLyrics

pianoRHMusic = \relative c { \key g \major g'1 b }
pianoLHMusic = \relative c { \clef bass \key g \major g1 b }

violinIMusic = \relative c' { \key g \major g'1 b }

violinIIImusic = \relative c' { \key g \major g'1 b }

violaMusic = \relative c { \clef alto \key g \major g'1 b }
celloMusic = \relative c { \clef bass \key g \major g1 b }
bassMusic = \relative c { \clef "bass_8" \key g \major g,1 b }

\score {
<<
\new StaffGroup = "StaffGroup_woodwinds" <<
\new Staff = "Staff_flute" \with { instrumentName = "Flute" }
\fluteMusic

\new Staff = "Staff_clarinet" \with {
    instrumentName = \markup { \concat { "Clarinet in B" \flat } } }

% Declare that written Middle C in the music
% to follow sounds a concert B flat, for
% output using sounded pitches such as MIDI.
%\transposition bes

% Print music for a B-flat clarinet
\transpose bes c' \clarinetMusic
>>

\new StaffGroup = "StaffGroup_brass" <<
\new Staff = "Staff_hornI" \with { instrumentName = "Horn in F" }
% \transposition f
\transpose f c' \hornMusic

\new Staff = "Staff_trumpet" \with { instrumentName = "Trumpet in C" }
\trumpetMusic

>>

\new RhythmicStaff = "RhythmicStaff_percussion"
\with { instrumentName = "Percussion" }
<<
\percussionMusic
>>
\new PianoStaff \with { instrumentName = "Piano" }
<<
\new Staff { \pianoRHMusic }
\new Staff { \pianoLHMusic }
>>
\new ChoirStaff = "ChoirStaff_choir" <<
\new Staff = "Staff_soprano" \with { instrumentName = "Soprano" }
\new Voice = "soprano"
\sopranoMusic
\new Lyrics \lyricsto "soprano" { \sopranoLyrics }
\new GrandStaff = "GrandStaff_altos"
\with { \accepts Lyrics } <<
\new Staff = "Staff_altoI" \with { instrumentName = "Alto I" }
\new Voice = "altoI"
\altoIMusic
\new Lyrics \lyricsto "altoI" { \altoILyrics }
\new Staff = "Staff_altoII" \with { instrumentName = "Alto II" }
\new Voice = "altoII"
\altoIIMusic
\new Lyrics \lyricsto "altoII" { \altoIILyrics }
>>
\new Staff = "Staff_tenor" \with { instrumentName = "Tenor" }
\new Voice = "tenor"
\tenorMusic
\new Lyrics \lyricsto "tenor" { \tenorLyrics }
>>
\new StaffGroup = "StaffGroup_strings" <<
\new GrandStaff = "GrandStaff_violins" <<
\new Staff = "Staff_violinI" \with { instrumentName = "Violin I" }
\violinIMusic
\new Staff = "Staff_violinII" \with { instrumentName = "Violin II" }
\violinIIMusic
>>
\new Staff = "Staff_viola" \with { instrumentName = "Viola" }
\violaMusic
\new Staff = "Staff_cello" \with { instrumentName = "Cello" }
\celloMusic
\new Staff = "Staff_bass" \with { instrumentName = "Double Bass" }
\bassMusic
>>
\layout { }
Print ChordNames with same root and different bass as slash and bass-note

To print subsequent ChordNames only differing in its bass note as slash and bass note use the here defined engraver. The behaviour may be controlled in detail by the chordChanges context property.

```scheme
(define Bass_changes_equal_root_engraver
  (lambda (ctx)
    "For sequential @code{ChordNames} with same root, but different bass, the root markup is dropped: D D/C D/B -> D /C /B
    The behaviour may be controlled by setting the @code{chordChanges} context-property."
    (let ((chord-pitches '())
           (last-chord-pitches '())
           (bass-pitch #f))
      (make- engraver
        ((initialize this-engraver)...
```

(Please note: The full code snippet is not provided due to limitations in the natural text representation.)
(let ((chord-note-namer (ly:context-property ctx 'chordNoteNamer)))
  ;; Set 'chordNoteNamer, respect user setting if already done
  (ly:context-set-property! ctx 'chordNoteNamer
    (if (procedure? chord-note-namer)
        chord-note-namer
        note-name->markup)))

(listeners
  ((note-event this-engraver event)
    (let* ((pitch (ly:event-property event 'pitch))
           (pitch-name (ly:pitch-notename pitch))
           (pitch-alt (ly:pitch-alteration pitch))
           (bass (ly:event-property event 'bass #f))
           (inversion (ly:event-property event 'inversion #f)))
      ;; Collect notes of the chord
      ;; - to compare inverted chords we need to collect the bass note
      ;;   as usual member of the chord, whereas an added bass must be
      ;;   treated separate from the usual chord-notes
      ;; - notes are stored as pairs containing their
      ;;   pitch-name (an integer), i.e. disregarding their octave and
      ;;   their alteration
      (cond (bass (set! bass-pitch pitch))
            (inversion
             (set! bass-pitch pitch)
             (set! chord-pitches
              (cons (cons pitch-name pitch-alt) chord-pitches)))
            (else
             (set! chord-pitches
              (cons (cons pitch-name pitch-alt) chord-pitches))))))

(acknowledgers
  ((chord-name-interface this-engraver grob source-engraver)
    (let ((chord-changes (ly:context-property ctx 'chordChanges #f)))
      ;; If subsequent chords are equal apart from their bass,
      ;; reset the 'text-property.
      ;; Equality is done by comparing the sorted lists of this chord's
      ;; elements and the previous chord. Sorting is needed because
      ;; inverted chords may have a different order of pitches.
      ;; 'chord-changes' needs to be true
      (if (and bass-pitch
               (equal?
                (sort chord-pitches car)<)
               (sort last-chord-pitches car)<)
          (ly:grob-set-property! grob 'text
            (make-line-markup
             (list
              (ly:context-property ctx 'slashChordSeparator)
              (ly:context-property ctx 'chordNoteNamer)
              bass-pitch
              (ly:context-property ctx 'chordNameLowercaseMinor)))))
      (set! last-chord-pitches chord-pitches)
      (set! chord-pitches '())
      (set! bass-pitch #f))))
((finalize this- engraver)
(set! last- chord- pitches '()))))

myChords = \chordmode {
  \germanChords

  \set chordChanges = \#t
d2:m d:m/cis
d:m/c

  \set chordChanges = \#f
d:m/b
e1:7

  \set chordChanges = \#t
e
  \break
  \once \set chordChanges = \#f
e1/f
e2/gis e/+gis e:m/f d:m d:m/cis d:m/c

  \set chordChanges = \#f
d:m/b
}

<<
  new ChordNames
  \with { \consists Bass_changes_equal_root_engraver }

  new Staff myChords
>>

\Staff

\n\nPutting lyrics inside the staff

Lyrics can be moved vertically to place them inside the staff. The lyrics are moved with
\override LyricText.extra- offset = \#'(0 . dy) and there are similar commands to move
the extenders and hyphens. The offset needed is established with trial and error.

<<
  new Staff <<

  new Voice = "voc" \relative c' { \stemDown a bes c8 b c4 }

>>
\new Lyrics \with { 
  \override LyricText.extra-offset = #'(0 . 8.6) 
  \override LyricExtender.extra-offset = #'(0 . 8.6) 
  \override LyricHyphen.extra-offset = #'(0 . 8.6) 
} \lyricsto "voc" { La la -- la ___ la } 

Quoting another voice

The quotedEventTypes property determines the music event types which should be quoted. The default value is (note-event rest-event tie-event beam-event tuplet-span-event), which means that only the notes, rests, ties, beams and tuplets of the quoted voice will appear in the \quoteDuring expression.

In the following example, a 16th rest is not quoted since rest-event is not in quotedEventTypes.

For a list of event types, consult the “Music classes” section of the Internals Reference.

quoteMe = \relative c' { 
  fis4 r16 a8.-- b4\ff c 
} 
\addQuote quoteMe \quoteMe 

original = \relative c' { 
  c8 d s2 
  \once \override NoteColumn.ignore-collision = ##t 
  es8 gis8 
} 

<< 
\new Staff \with { instrumentName = "quoteMe" } 
\quoteMe 

\new Staff \with { instrumentName = "orig" } 
original 

\new Staff \with { 
  instrumentName = "orig+quote" 
  quotedEventTypes = #'(note-event articulation-event) 
} 
\relative c'' 
<< 
\new Voice { 
  s4 
  \set fontSize = #-4 
  \override Stem.length-fraction = #(magstep -4)
Quoting another voice with transposition

Quotations take into account the transposition of both source and target. In this example, all instruments play sounding middle C; the target is an instrument in F. The target part may be transposed using \transpose. In this case, all the pitches (including the quoted ones) are transposed.

\addQuote clarinet {
  \transposition bes
  \repeat unfold 8 { d'16 d' d'8 }
}

\addQuote sax {
  \transposition es'
  \repeat unfold 16 { a8 }
}

quoteTest = {
  % french horn
  \transposition f
  g'4
  << \quoteDuring "clarinet" { \skip 4 } s4~"clar." >>
  << \quoteDuring "sax" { \skip 4 } s4~"sax." >>
  g'4
}

{
  \new Staff \with {
    instrumentName = \markup { \column { Horn "in F" } }
  }
  quoteTest
  \transpose c' d' << \quoteTest s4~"up a tone" >>
}

Horn in F clar. sax. clar. sax. up a tone
Removing brace on first line of piano score

This snippet removes the first brace from a PianoStaff or a GrandStaff. It may be useful when cutting and pasting the engraved image into existing music. It uses `\alterBroken`.

```latex
someMusic = {
  \once \override Staff.Clef.stencil = ##f
  \once \override Staff.TimeSignature.stencil = ##f
  \repeat unfold 3 c1 \break
  \repeat unfold 5 c1 \break
  \repeat unfold 5 c1
}

\score {
  \new PianoStaff
  \<<
    \new Staff = "right" \relative c' \someMusic
    \new Staff = "left" \relative c' { \clef F \someMusic }
  \>>
  \layout {
    \indent = 75
    \context {
      \PianoStaff
      \alterBroken transparent #'(\t) SystemStartBrace
    }
  }
}
```

Eliminare la prima linea vuota

Il primo rigo vuoto si può toglie re dalla partitura impostando la proprietà `remove-first` di `VerticalAxisGroup`. Questa impostazione agisce a livello globale se posta nel blocco `\layout`,
a livello locale se posta nel rigo specifico che deve essere tolto. Nel secondo caso, si deve specificare il contesto (Staff si applica solo al rigo corrente) prima della proprietà.

Il rigo inferiore del secondo gruppo di righi non viene rimosso, perché l'impostazione ha effetto solo sul rigo in cui si trova.

```latex
\layout {
  \context {
    \Staff \RemoveEmptyStaves
    % To use the setting globally, uncomment the following line:
    % \override VerticalAxisGroup.remove-first = ##t
  }
}
\new StaffGroup <<
  \new Staff {relative c' { e4 f g a \break c1 }
  \new Staff {
    % To use the setting globally, comment this line,
    % uncomment the line in the \layout block above
    \override Staff.VerticalAxisGroup.remove-first = ##t
    R1 \break R
  }
}>>
\new StaffGroup <<
  \new Staff {relative c' { e4 f g a \break c1 }
  \new Staff { R1 \break R }
}>>
```

![Music notation image]
Setting system separators

System separators can be inserted between systems. Any markup can be used, but \slashSeparator has been provided as a sensible default.

\paper {
  system-separator-markup = \slashSeparator
  line-width = 120
}

notes = \relative c' {
  c1 | c \break
  c1 | c \break
  c1 | c
}

\book {
\score {
  \new GrandStaff <<
  \new Staff \notes
  \new Staff \notes
  >>
}\score
}\book
Staff notation
Tick bar lines

'Tick' bar lines are often used in music where the bar line is used only for coordination and is not meant to imply any rhythmic stress.

```latex
\relative c' { 
\set Score.measureBarType = #"\''"
\c4 d e f
g4 f e d
\c4 d e f
g4 f e d
\bar "|." 
}
```

![Tick bar lines example](image)

Time signature in parentheses

The time signature can be enclosed within parentheses.

```latex
\relative c'' { 
\override Staff.TimeSignature.stencil = #(lambda (grob)
  (bracketify-stencil (ly:time-signature::print grob) Y 0.1 0.2 0.1))
\time 2/4
a4 b8 c 
}
```

![Time signature in parentheses example](image)

Time signature in parentheses - method 3

Another way to put the time signature in parenthesis

```latex
\relative c'' { 
\override Staff.TimeSignature.stencil = #(lambda (grob)
  (parenthesize-stencil (ly:time-signature::print grob) 0.1 0.4 0.4 0.1 ))
\time 2/4
a4 b8 c 
}
```

![Time signature in parentheses method 3 example](image)

Modifiche manuali della proprietà della chiave

Cambiando il glifo della chiave, la sua posizione o l’ottavazione non cambia la posizione delle note successive nel rigo. Per far sì che le armature di chiave si trovino sulle linee del rigo corrette, bisogna specificare anche middleCPosition, con valori positivi o negativi che spostano il Do centrale rispettivamente su o giù in senso relativo alla linea centrale del rigo.
Per esempio, \clef "treble\_8" equivale a impostare clefGlyph, clefPosition (che regola la posizione verticale della chiave), middleCPosition e clefTransposition. Viene stampata una chiave quando cambia una di queste proprietà, eccetto middleCPosition.

Gli esempi seguenti mostrano le possibilità date dall’impostazione manuale di tali proprietà. Sulla prima linea le modifiche manuali preservano il posizionamento relativo standard di chiavi e note, mentre sulla seconda linea non lo fanno.

{%
% The default treble clef
\key f \major
\clef c'1
%
% The standard bass clef
\set Staff.clefGlyph = #"clefs.F"
\set Staff.clefPosition = #2
\set Staff.middleCPosition = #6
\set Staff.middleCClefPosition = #6
\key g \major
\clef c'1
%
% The baritone clef
\set Staff.clefGlyph = #"clefs.C"
\set Staff.clefPosition = #4
\set Staff.middleCPosition = #4
\set Staff.middleCClefPosition = #4
\key f \major
\clef c'1
%
% The standard choral tenor clef
\set Staff.clefGlyph = #"clefs.G"
\set Staff.clefPosition = #-2
\set Staff.clefTransposition = #-7
\set Staff.middleCPosition = #1
\set Staff.middleCClefPosition = #1
\key f \major
\clef c'1
%
% A non-standard clef
\set Staff.clefPosition = #0
\set Staff.clefTransposition = #0
\set Staff.middleCPosition = #-4
\set Staff.middleCClefPosition = #-4
\key g \major
\clef c'1 \break
%
% The following clef changes do not preserve
% the normal relationship between notes, key signatures
% and clefs:
%
\set Staff.clefGlyph = #"clefs.F"
\set Staff.clefPosition = #2
\clef c'1
\set Staff.clefGlyph = #"clefs.G"
\clef c'1
\set Staff.clefGlyph = #"clefs.C"
\clef c'1
%}
\set Staff\.clefTransposition = \#7
\set Staff\.clefTransposition = \#0
\set Staff\.clefPosition = \#0
\set Staff\.melisma\ = \#0
\c'1

% Return to the normal clef:
\set Staff\.middleCPosition = \#0
\c'1

Two \partCombine pairs on one staff

The \partCombine function takes two music expressions each containing a part, and distributes them among four Voices named “two”, “one”, “solo”, and “chords” depending on when and how the parts are merged into a common voice. The voices output from \partCombine can have their layout properties adjusted in the usual way. Here we define extensions of \partCombine to make it easier to put four voices on a staff.

soprano = \{ d'4 | cis' b e' d'8 cis' | cis'2 b \}
alto = \{ fis4 | e8 fis gis ais b4 b | b ais fis2 \}
tenor = \{ a8 b | cis' dis' e'4 b8 cis' d'4 | gis cis' dis'2 \}
bass = \{ fis8 gis | a4 gis g fis | eis fis b,2 \}

\new Staff <<
\key b\minor
\clef alto
\partial 4
\transpose b b'
\partCombineUp soprano \alto
\partCombineDown tenor \bass
>>

\layout {
\context {
\Staff
\accepts "VoiceBox"
}
\context {
\name "VoiceBox"
\type "Engraver\_group"
\defaultchild "Voice"
\accepts "Voice"
\accepts "NullVoice"
}
customPartCombineUp = 
#(define-music-function (partOne partTwo)
 (ly:music? ly:music?)
"Take the music in @var{partOne} and @var{partTwo} and return a @code{VoiceBox} named @q{Up} containing @code{Voice}s that contain @var{partOne} and @var{partTwo} merged into one voice where feasible. This variant sets the default voicing in the output to use upward stems."
#{
  \new VoiceBox = "Up" <<
  \context Voice = "one" { \voiceOne }
  \context Voice = "two" { \voiceThree }
  \context Voice = "shared" { \voiceOne }
  \context Voice = "solo" { \voiceOne }
  \context NullVoice = "null" {}
  \partCombine #partOne #partTwo
  >>
}#)

customPartCombineDown = #
(define-music-function (partOne partTwo)
 (ly:music? ly:music?)
"Take the music in @var{partOne} and @var{partTwo} and return a @code{VoiceBox} named @q{Down} containing @code{Voice}s that contain @var{partOne} and @var{partTwo} merged into one voice where feasible. This variant sets the default voicing in the output to use downward stems."
#{
  \new VoiceBox = "Down" <<
  \set VoiceBox.soloText =="#Solo III"
  \set VoiceBox.soloIIText =="#Solo IV"
  \context Voice ="one" { \voiceFour }
  \context Voice ="two" { \voiceTwo }
  \context Voice ="shared" { \voiceFour }
  \context Voice ="solo" { \voiceFour }
  \context NullVoice = "null" {}
  \partCombine #partOne #partTwo
  >>
}#)

soprano = { d'4 | cis' b e' d'8 cis' | cis'2 b }
alto = { fis4 | e8 fis gis ais b4 b | b ais fis2 }
tenor = { a8 b | cis' dis' e'4 b8 cis' d'4 | gis cis' dis'2 }
bass = { fis8 gis | a4 gis g fis | eis fis b,2 }

\new Staff <<
\key b\minor
\clef alto
\partial 4
Usare una parentesi quadra all’inizio di un gruppo di righi

Si può usare il segno SystemStartSquare (uno dei segni che delimitano l’inizio del sistema) impostandolo esplicitamente in un contesto StaffGroup o ChoirStaff.

```
\score {
  \new StaffGroup { <<
    \set StaffGroup.systemStartDelimiter = #'SystemStartSquare
    \new Staff { c'4 d' e' f' }
    \new Staff { c'4 d' e' f' }
  } >> }
```

Using autochange with more than one voice

Using autochange with more than one voice.

```
\score
{
  \new PianoStaff
  \new Staff = "up" { <<
    \set Timing.beamExceptions = #'()
    \set Timing.beatStructure = '#(4)
    \new Voice {
      \relative c' {
        g8 a b c d e f g
        g,8 a b c d e f g
      }
    } }
    \new Voice { }
}
```
Using marklines in a Frenched score

Using MarkLine contexts (such as in LSR1010 (https://lsr.di.unimi.it/LSR/Item?id=1010)) in a Frenched score can be problematic if all the staves between two MarkLines are removed in one system. The \texttt{Keep\_alive\_together\_engraver} can be used within each \texttt{StaffGroup} to keep the MarkLine alive only as long as the other staves in the group stay alive.

\begin{verbatim}
bars = {
    \tempo "Allegro" 4=120
    s1*2
    \repeat unfold 5 \{ \mark \default s1*2 \}
    \bar "||"
    \tempo "Adagio" 4=40
    s1*2
    \repeat unfold 8 \{ \mark \default s1*2 \}
    \bar "."
}
winds = \repeat unfold 120 \{ c''4 \}
trumpet = \{ \repeat unfold 8 g'2 R1*16 \repeat unfold 4 g'2 R1*8 \}
trombone = \{ \repeat unfold 4 c'1 R1*8 d'1 R1*17 \}
strings = \repeat unfold 240 \{ c''8 \}

#(set-global-staff-size 16)
\paper {
    systems-per-page = 5
    ragged-last-bottom = ##f
}
\layout {
\end{verbatim}
indent = 15\mm
short-indent = 5\mm
\context {  
  \name MarkLine  
  \type Engraver_group  
  \consists Output_property_engraver  
  \consists Axis_group_engraver  
  \consists Mark_engraver  
  \consists Metronome_mark_engraver  
  \consists Staff_collecting_engraver  
  \override VerticalAxisGroup.remove-empty = ##t  
  \override VerticalAxisGroup.remove-layer = #'any  
  \override VerticalAxisGroup.staff-affinity = #DOWN  
  \override VerticalAxisGroup.nonstaff-relatedstaff-spacing.padding = 1  
  \keepAliveInterfaces = #'()  
}
\context {  
  \Staff  
  \override VerticalAxisGroup.remove-empty = ##t  
  \override VerticalAxisGroup.remove-layer = ##f  
}
\context {  
  \StaffGroup  
  \accepts MarkLine  
  \consists Keep_alive_together_engraver  
}
\context {  
  \Score  
  \remove Mark_engraver  
  \remove Metronome_mark_engraver  
  \remove Staff_collecting_engraver  
}
}
\score {  
  \new StaffGroup = "winds" \with {  
    instrumentName = "Winds"  
    shortInstrumentName = "Winds"  
  }  
  \new MarkLine \bars  
  \new Staff \winds  
}  
\new StaffGroup = "brass" <<  
  \new MarkLine \bars  
  \new Staff = "trumpet" \with {  
    instrumentName = "Trumpet"  
    shortInstrumentName = "Tpt"  
  } \trumpet  
  \new Staff = "trombone" \with {  
    instrumentName = "Trombone"  
    shortInstrumentName = "Tbn"  
  } \trombone  
}
\new StaffGroup = "strings" \with {
  instrumentName = "Strings"
  shortInstrumentName = "Strings"
}
<<
\new MarkLine \bars
\new Staff = "strings" { \strings }
>>

\new \trombone

Allegro ($\d = 120$)

A B

Winds

Trumpet

Trombone

Strings

C D

Adagio ($\d = 40$)

F

E

Tbn

E

Strings

G H

Winds

Strings
Vertical aligned StaffGroups without connecting SystemStartBar

This snippet shows how to achieve vertically aligned StaffGroups with a SystemStartBar for each StaffGroup, but without connecting them.

```
#(set-global-staff-size 18)

\paper {
  indent = 0
  ragged-right = ##f
  print-all-headers = ##t
}

\layout {
  \context {
    \StaffGroup
    \consists Text_mark_engraver
    \consists Staff_collecting_engraver
    systemStartDelimiterHierarchy =
      #'(SystemStartBrace (SystemStartBracket a b))
  }

  \context {
    \Score
    \remove Text_mark_engraver
    \remove Staff_collecting_engraver
    \override SystemStartBrace.style = #'bar-line
    \omit SystemStartBar
    \override SystemStartBrace.padding = #-0.1
    \override SystemStartBrace.thickness = #1.6
    \override StaffGrouper.staffgroup-staff-spacing.basic-distance = #15
  }
}
%% EXAMPLE

txt = \lyricmode {
    Wer4 nur den lie -- ben Gott läßt wal2 -- ten4
    und4 hof -- fet auf ihn al -- le Zeit2.
}

% First StaffGroup "exercise"

eI = \relative c' {
    \textMark \markup {
        \text \Teacher:
            This is a simple setting of the choral. Please improve it.
    }
    \key a \minor
    \time 4/4
    \voiceOne

    \partial 4
e4
    a b c b
    a b gis2
    e4\fermata g! g f
    e a a gis
    a2.\fermata

    \bar "|.|"
}

eII = \relative c' {
    \key a \minor
    \time 4/4
    \voiceTwo
    \partial 4
c4
    e e e gis
    a f e2
    b4 b d d
    c c d d
    c2.

    \bar "|.|"
}

eIII =
\relative c' {
    \key a \minor
    \time 4/4
    \clef bass
    \voiceOne
\partial 4
a_4
c b a b
c d b_2
gis_4 g g b
c a f e
e_2.
}
eIV =
\relative c' {
key a \minor
\time 4/4
\clef bass
\voiceTwo
\partition 4
a,4
a', gis a e
a, d e_2
e,4 \text{\ fermata} e' b g
c f d e
a,2. \text{\ fermata}
\bar ".:|."}
}

exercise =
\new StaffGroup = "exercise"
<<
\new Staff
<<
\new Voice \eI
\new Voice \eII
>>
\new Lyrics \txt
\new Staff
<<
\new Voice \eIII
\new Voice \eIV
>>
>>

/* Second StaffGroup "simple Bach"

sbI =
\relative c' {
\textMark \markup { \textbf{Pupil:} Here's my version! }
\key a \minor
\time 4/4
\voiceOne
\partial 4
e4
a b c b
a b gis2
e4\fermata g! g f
e a a gis
a2.\fermata
\bar "::.|."
}

sbII =
\relative c' {
  \key a \minor
  \time 4/4
  \voiceTwo
  \partial 4
c8 d
e4 e e8 f g4
  f f e2
  b4 b8 c d4 d
e8 d c4 b8 c d4
c2.
  \bar "::.|."
}

sbIII =
\relative c' {
  \key a \minor
  \time 4/4
  \clef bass
  \voiceOne

  \partial 4
  a8 b
c4 b a b8 c
d4 d8 c b2
gis4 g g8 a b4
  b a8 g f4 e
e2.
}

sbIV =
\relative c' {
  \key a \minor
  \time 4/4
  \clef bass
  \voiceTwo

  \partial 4
  a,4
\new StaffGroup = "simple Bach"

\new Staff
<<
  \new Voice \sbI
  \new Voice \sbII
>>

\new Lyrics \txt

\new Staff
<<
  \new Voice \sbIII
  \new Voice \sbIV
>>

% Third StaffGroup "chromatic Bach"

\new Voice \cbI
\relative c' {
  \textMark \markup {
    \bold "Teacher:"
    \column {
      "Well, you simply copied and transposed a version of J.S.Bach."
      "Do you know this one?"
    }
  }
  \key a \minor
  \time 4/4
  \voiceOne

  \partial 4
  e4
  a b c b
  a b gis4. fis8
  e4\fermata g! g f
  e a a8 b gis4
  a2.\fermata
  \bar ":|."
cbII = 
\relative c' { 
  \key a \minor 
  \time 4/4 
  \partialTwo 
  \c8 d 
  e4 e e8 fis gis4 
  a8 g! f!4 e2 
  b4 e e d 
  d8[ cis] d dis e fis e4 
  e2. 
  \bar ":|." 
}

cbIII = 
\relative c' { 
  \key a \minor 
  \time 4/4 
  \clef bass 
  \partialOne 
  \a8 b 
  c[ b] a gis8 a4 d, 
  e8[ e'] d c b4. a8 
  gis4 b c d8 c 
  b[ a] a b c b b c16 d 
  c2. 
}

cbIV = 
\relative c' { 
  \key a \minor 
  \time 4/4 
  \clef bass 
  \partialTwo 
  \c4 e a, b 
  c d e2 
  e4\fermata e a b8 c 
  gis[ g] fis f e dis e4 
  a,2.\fermata 
  \bar ".:|." 
}

chromaticBach = 
\new StaffGroup = "chromatic Bach"
<<
Exercise: Improve the given choral
Teacher: This is a simple setting of the choral. Please improve it.

Pupil: Here's my version!

Teacher: Well, you simply copied and transposed a version of J.S. Bach. Do you know this one?
Volta sotto gli accordi

Aggiungendo l’incisore Volta_engraver al rigo, è possibile inserire le volte sotto gli accordi.

```latex
\score {
  <<
    \chords {
      c1
    }
    \new Staff \with {
      \consists "Volta_engraver"
    }
    {\repeat volta 2 { c'1 }
    \alternative { c' } }
  >>
  \context {
    \Score
    \remove "Volta_engraver"
  }
}
```
Volta multi staff

By adding the `Volta_engraver` to the relevant staff, volte can be put over staves other than the topmost one in a score.

```latex
\texttt{voltaMusic} = \texttt{\relative c'} \{ \\
\texttt{\repeat volta 2 \{ \\
C1 \\
\}} \texttt{\alternative \{ \\
D1 \\
E1 \\
\}} \\
\} \\
<< \texttt{\new StaffGroup} << \\
\texttt{\new Staff \ voltaMusic} \\
\texttt{\new Staff \ voltaMusic} \\
\} \texttt{\new StaffGroup} << \\
\texttt{\new Staff \ with \{ \ consists \ "Volta_engraver" \}} \\
\texttt{\ voltaMusic} \\
\texttt{\ new Staff \ voltaMusic} \\
\} \\
\} \\
```

```latex
\begin{music}
\begin{notation}
\texttt{\relative c'} \\
\texttt{\repeat volta 2 \{ \\
C1 \\
\}} \texttt{\alternative \{ \\
D1 \\
E1 \\
\}} \\
\end{notation}
\end{music}
```
Editorial annotations

Sezione “Editorial annotations” in Guida alla Notazione

Adding fingerings to a score

Fingering instructions can be entered using a simple syntax.
\relative c' { 
  c4-1 d-2 f-4 e-3 
}

Adding links to objects

To add a link to a grob stencil you can use add-link as defined here. It works both with \override and \tweak.

Drawback: point-and-click is disturbed for the linked grobs.

Limitation: Works for PDF only.

The linked objects are colored with a separate command. Note that the links are not displayed and are not clickable from inside the LSR.

#(define (add-link url-strg)
  (lambda (grob)
    (let* ((stil (ly:grob-property grob 'stencil)))
      (if (ly:stencil? stil)
        (let* ((x-ext (ly:stencil-extent stil X))
               (y-ext (ly:stencil-extent stil Y))
               (url-expr `(url-link ,url-strg ,x-ext ,y-ext))
               (new-stil
                (ly:stencil-add
                 (ly:make-stencil url-expr x-ext y-ext)
                 stil)))
        (ly:grob-set-property! grob 'stencil new-stil)))))

%%% test

%%% For easier maintenance of this snippet the URL is formatted to use the %% actually used LilyPond version.
%%% Of course a literal URL would work as well.

#(define major.minor-version
  (string-join (take (string-split (lilypond-version) \\.) 2) "."))

urlI =
#(format #f
  "http://lilypond.org/doc/v-\a/Documentation/notation/writing-pitches"
  major.minor-version)

urlII =
\relative c' {
    \key cis \minor

    \once \override Staff.Clef.color = #green
    \once \override Staff.Clef.after-line-breaking = #(add-link urlI)

    \once \override Staff.TimeSignature.color = #green
    \once \override Staff.TimeSignature.after-line-breaking = #(add-link urlII)

    \once \override NoteHead.color = #green
    \once \override NoteHead.after-line-breaking = #(add-link urlIII)

    cis'1
    \once \override Beam.color = #green
    \once \override Beam.after-line-breaking = #(add-link urlIV)
    cis8 dis e fis gis2
    <gis,
    \tweak Accidental.color #green
    \tweak Accidental.after-line-breaking #(add-link urlVI)
    \tweak color #green
    \tweak after-line-breaking #(add-link urlV)
    \tweak style #'harmonic
    bis
    dis
fis
>1
<cis, cis' e>

Adding markups in a tablature

By default markups does not show in a tablature.

To make them appear, simply use the command \revert TabStaff.TextScript.stencil

%% http://lsr.di.unimi.it/LSR/Item?id=919
%% by P.P.Schneider on June 2014

high = { r4 r8 <g c'> q r8 r4 }

low = { c4 r4 c8 r8 g,8 b, }

pulse = { s8"1" s"\&" s"2" s"\&" s"3" s"\&" s"4" s"\&" }

\score {
  \new TabStaff {
    \repeat unfold 2 << \high \ \low \ \pulse >>
  }
  \layout {
    \context {
      \TabStaff
      \clef moderntab
      \revert TextScript.stencil
      \override TextScript.font-series = #'bold
      \override TextScript.font-size = #-2
      \override TextScript.color = #red
    }
    \context {
      \Score
      proportionalNotationDuration = #(ly:make-moment 1/8)
    }
  }
}

Far sì che la diteggiatura appaia dentro il rigo

Per impostazione predefinita, le diteggiature orientate verticalmente sono poste fuori dal rigo; questo comportamento tuttavia può essere disabilitato. Occorre fare attenzione alle situazioni
in cui le diteggiature e i gambi sono rivolti nella stessa direzione: normalmente le diteggiature evitano soltanto i gambi con travature. Questa impostazione predefinita può essere cambiata in modo da evitare tutti i gambi oppure nessuno. L’esempio seguente mostra queste due opzioni, così come tornare al comportamento predefinito.

\relative c' {
  \override Fingering.staff-padding = #'()
  <c-1 e-2 g-3 b-5>2
  \override Fingering.add-stem-support = ##f
  a[-1 b]-2 g-0 r
  \override Fingering.add-stem-support = ##t
  a[-1 b]-2 g-0 r
  \override Fingering.add-stem-support = #only-if-beamed
  a[-1 b]-2 g-0 r
}

Numeri di battuta alternativi

Si possono impostare due metodi alternativi di numerazione della battuta, utili specialmente per le ripetizioni.

\relative c' {  
  \set Score.alternativeNumberingStyle = #'numbers
  \repeat volta 3 { c4 d e f | } 
  \alternative { 
  \set Score.alternativeNumberingStyle = #'numbers
  \repeat volta 3 { c,4 d e f | } 
  \alternative { 
  \set Score.alternativeNumberingStyle = #'numbers-with-letters
  \repeat volta 3 { c,4 d e f | } 
  \alternative {
  } 
  c1 
  \endrepeat
}
Parentesi analitiche sopra il rigo

Delle semplici parentesi analitiche orizzontali vengono aggiunte, per impostazione predefinita, sotto il rigo. L’esempio seguente mostra un modo per posizionarle sopra il rigo.

\layout{
  \context{
    \Voice
    \consists "Horizontal_bracket_engraver"
  }
}

\relative c''{
  \once \override HorizontalBracket.direction = #UP
  c2\startGroup
d2\stopGroup
}

Parentesi analitiche con etichette

Si può aggiungere del testo alle parentesi analitiche tramite la proprietà text del grob HorizontalBracketText. L’aggiunta di vari frammenti di testo alle parentesi che iniziano nello stesso momento musicale richiede l’uso del comando \tweak. Dopo un’interruzione di linea il testo viene messo tra parentesi.

\paper{ tagline = ##f}
Applicazione degli stili delle teste di nota in base al grado della scala

La proprietà `shapeNoteStyles` può essere usata per definire vari stili di teste di nota per ogni grado della scala (definita dall’armatura di chiave o dalla proprietà `tonic`). Questa proprietà richiede un insieme di simboli, che può essere puramente arbitrario (sono permessi espressioni geometriche come `triangle`, `cross` e `xcircle`) o basato sull’antica tradizione tipografica americana (sono consentiti anche alcuni nomi di nota latini).

Detto questo, per imitare gli antichi canzonieri americani, ci sono vari stili predefiniti disponibili attraverso dei comodi comandi come `\aikenHeads` o `\sacredHarpHeads`.

Questo esempio mostra modi diversi di ottenere teste di nota di varie forme e illustra la possibilità di trasporre una melodia senza perdere la corrispondenza tra le funzioni armoniche e gli stili delle teste.

```plaintext
\layout { 
  \context { 
    \Voice
    \consists "Horizontal_bracket_engraver"
    \override HorizontalBracket.direction = #UP
  }
  
  \once \override HorizontalBracketText.text = "a"
  c'\startGroup d'\stopGroup
  \once \override HorizontalBracketText.text = "a'"
  e'\startGroup d'\stopGroup |
  c'\tweak HorizontalBracketText.text
    \markup \bold \huge "b" \startGroup
    \tweak HorizontalBracketText.text "a" \startGroup
  d'\stopGroup
  e'\tweak HorizontalBracketText.text "a'" \startGroup
  d'\stopGroup\stopGroup |
  c'\tweak HorizontalBracketText.text foo \startGroup
    d' e' f' | \break
  g'\startGroup a' b' c'\stopGroup
}
```

\[\begin{array}{c}
  \text{foo} \\
  (\text{foo})
\end{array}\]

\[\begin{array}{c}
  \text{\textbf{b}} \\
  a \quad a' \quad a \quad a' \quad \text{foo}
\end{array}\]
Blanking staff lines using the \whiteout command

The \whiteout command underlays a markup with a white box. Since staff lines are in a lower layer than most other grobs, this white box will not overlap any other grob.

Changing a single note’s size in a chord

Individual note heads in a chord can be modified with the \tweak command inside a chord, by altering the font-size property.
Inside the chord (within the brackets < >), before the note to be altered, place the \tweak command, followed by font-size and define the proper size like #,-2 (a tiny note head).

```latex
\relative c' {
\tweak font-size #+2 c e g c
\tweak font-size #-2 e>1
\markup { A tiny e }
\markup { A big c }
}
```

Changing the appearance of a slur from solid to dotted or dashed

The appearance of slurs may be changed from solid to dotted or dashed.

```latex
\relative c' {
\c4( d e c)
\slurDotted c4( d e c)
\slurSolid c4( d e c)
\slurDashed c4( d e c)
\slurSolid c4( d e c)
}
```

Coloring notes depending on their pitch

It is possible to color note heads depending on their pitch and/or their names: the function used in this example even makes it possible to distinguish enharmonics.

```latex
%(define color-mapping
(list
  (cons (ly:make-pitch 0 0 NATURAL) (x11-color 'red))
  (cons (ly:make-pitch 0 0 SHARP) (x11-color 'green))
  (cons (ly:make-pitch 0 1 FLAT) (x11-color 'green))
  (cons (ly:make-pitch 0 2 NATURAL) (x11-color 'red))
  (cons (ly:make-pitch 0 2 SHARP) (x11-color 'green))
  (cons (ly:make-pitch 0 3 FLAT) (x11-color 'red))
  (cons (ly:make-pitch 0 3 NATURAL) (x11-color 'green))
  (cons (ly:make-pitch 0 4 SHARP) (x11-color 'red))
  (cons (ly:make-pitch 0 5 NATURAL) (x11-color 'green))
  (cons (ly:make-pitch 0 5 FLAT) (x11-color 'red)))
```
%Compare pitch and alteration (not octave).

#(define (pitch-equals? p1 p2)
    (and
        (= (ly:pitch-alteration p1) (ly:pitch-alteration p2))
        (= (ly:pitch-notename p1) (ly:pitch-notename p2))))

#(define (pitch-to-color pitch)
    (let ((color (assoc pitch color-mapping pitch-equals?)))
        (if color
            (cdr color))))

#(define (color-notehead grob)
    (pitch-to-color
        (ly:event-property (event-cause grob) 'pitch)))

\score {
    \new Staff \relative c' {
        \override NoteHead.color = #color-notehead
        \set clef = \clef treble
        c8 b d dis ees f g aes
    }
}

Controllare il posizionamento delle diteggiature di un accordo

Il posizionamento dei numeri della diteggiatura può essere regolato in modo preciso. Perché l’orientamento funzioni, occorre usare il costrutto per gli accordi <> anche per le note singole. Si può impostare in modo simile l’orientamento dei numeri di corda e delle diteggiature della mano destra.

\relative c' {
    \set fingeringOrientations = #'(left)
    <c-1 e-3 a-5>4
    \set fingeringOrientations = #'(down)
    <c-1 e-3 a-5>4
    \set fingeringOrientations = #'(down right up)
    <c-1 e-3 a-5>4
    \set fingeringOrientations = #'(up)
    <c-1 e-3 a-5>4
    \set fingeringOrientations = #'(left)
    <c-1>2
    \set fingeringOrientations = #'(down)
    <e-3>2
\set stringNumberOrientations = #'(up left down)
<\f\r\3 \a\2 \c\1>1
\set strokeFingerOrientations = #'(down right up)
<<\rightHandFinger #1 e\rightHandFinger #2 c'\rightHandFinger #4 >

Creare un gruppetto ritardato

Creare un gruppetto ritardato, dove la nota più bassa del gruppetto usa l’alterazione, richiede vari \override. La proprietà outside-staff-priority deve essere impostata su #f, perché altri
menti questa avrebbe la precedenza sulla proprietà avoid-slur. Cambiando la frazione 2/3 si aggiusta la posizione orizzontale.
\relative c’ { 
  \after 2*2/3 \turn c2( d4) r |
  \after 4 \turn c4.( d8)
  \after 4
  {
    \once \set suggestAccidentals = ##t
    \once \override AccidentalSuggestion.outside-staff-priority = ##f
    \once \override AccidentalSuggestion.avoid-slur = #'inside
    \once \override AccidentalSuggestion.font-size = -3
    \once \override AccidentalSuggestion.script-priority = -1
    \once \hideNotes
cis8\turn \noBeam
  }
d4.( e8)
}

Creating blank staves

To create blank staves, generate empty measures then remove the Bar_number_engraver from
the Score context, and the Time_signature_engraver, Clef_engraver and Bar_engraver from the Staff context.
#(set-global-staff-size 20)

\score {
{ 
  \repeat unfold 12 { s1 \break } 
}
\layout {

indent = 0\in
\context {  
  \Staff
  \remove "Time_signature_engraver"
  \remove "Clef_engraver"
  \remove "Bar_engraver"
}
\context {  
  \Score
  \remove "Bar_number_engraver"
}
%

% uncomment these lines for "letter" size
%
\paper {  
  #(set-paper-size "letter")
  ragged-last-bottom = ##f
  line-width = 7.5\in
  left-margin = 0.5\in
  bottom-margin = 0.25\in
  top-margin = 0.25\in
}
%

% uncomment these lines for "A4" size
%
\paper {  
  #(set-paper-size "a4")
  ragged-last-bottom = ##f
  line-width = 180
  left-margin = 15
  bottom-margin = 10
  top-margin = 10
}
%
Creating double-digit fingerings

Creating fingerings larger than 5 is possible.

```latex
\relative c' {
    c1-10
    c1-50
    c1-36
    c1-29
}
```

Direzione predefinita dei gambi sulla linea centrale del rigo

La direzione predefinita dei gambi sulla linea centrale del rigo si imposta con la proprietà `neutral-direction` dell’oggetto Stem.

```latex
\relative c'' {
    a4 b c b
    \override Stem.neutral-direction = #up
    a4 b c b
    \override Stem.neutral-direction = #down
    a4 b c b
}
```
Different font size settings for instrumentName and shortInstrumentName

Choose different font sizes for instrumentName and shortInstrumentName as a context override.

```
InstrumentNameFontSize =
#(define-music-function (font-size-pair)(pair?)
"Sets the @code{font-size} of @code{InstrumentName}. The font-size for the initial @code{instrumentName} is taken from the first value in @var{font-size-pair}. @code{shortInstrumentName} will get the second value of @var{font-size-pair}.
"

;; This code could be changed/extended to set different values for each occurance of `shortInstrumentName'

#{
  \override InstrumentName.after-line-breaking =
  #(lambda (grob)
    (let* ((orig (ly:grob-original grob))
        (siblings (if (ly:grob? orig)
                      (ly:spanner-broken-into orig)
                      '())))
      (if (pair? siblings)
        (begin
          (ly:grob-set-property! (car siblings) 'font-size (car font-size-pair))
          (for-each (lambda (g)
                      (ly:grob-set-property! g 'font-size (cdr font-size-pair)))
                    (cdr siblings))))))

  \layout {
    \context {
      \Staff
        \InstrumentNameFontSize #'(6 . -3)
    }
    }

  \new StaffGroup <<
    \new Staff
      \with {
        instrumentName = "Flute"
        shortInstrumentName = "Fl."
      }
    { clef clef=perc c''1 \break c'' \break c'' }
  \new Staff
      \with {
        instrumentName = "Violin"
        shortInstrumentName = "Vl."
      }
```
Drawing boxes around grobs

The print-function can be overridden to draw a box around an arbitrary grob.

```tex
\relative c' \{ 
  \override TextScript.stencil =
    #(make-stencil-boxer 0.1 0.3 ly:text-interface::print) 
  c'4"foo"

  \override Stem.stencil =
    #(make-stencil-boxer 0.05 0.25 ly:stem::print)
  \override Score.RehearsalMark.stencil =
    #(make-stencil-boxer 0.15 0.3 ly:text-interface::print)

  b8

  \revert Stem.stencil
  \revert Flag.stencil
  c4. c4
  \mark \default
  c1
\}
```
Drawing circles around note heads

Here is how to circle a note.

\begin{verbatim}
circle = \once \override NoteHead.stencil = #(\lambda (grob)
(let* ((note (ly:note-head::print grob))
  (combo-stencil (ly:stencil-add note
    (circle-stencil note 0.1 0.8))))
  (ly:make-stencil (ly:stencil-expr combo-stencil)
    (ly:stencil-extent note X)
    (ly:stencil-extent note Y))))

{ \circle c'' }
\end{verbatim}

Drawing circles around various objects

The \circle markup command draws circles around various objects, for example fingering indications. For other objects, specific tweaks may be required: this example demonstrates two strategies for rehearsal marks and measure numbers.

\begin{verbatim}
relative c' {
c1 \set Score.rehearsalMarkFormatter =
  #(\lambda (mark context)
    (make-circle-markup (format-mark-numbers mark context)))
\mark \default

c2 d^\markup {
  \override #'(thickness . 3) {
    \circle \finger 2
  }
}
\override Score.BarNumber.break-visibility = #all-visible
\override Score.BarNumber.stencil =
  #(make-stencil-circler 0.1 0.25 ly:text-interface::print)
}\end{verbatim}

Embedding native PostScript in a \markup block

PostScript code can be directly inserted inside a \markup block.

\begin{verbatim}
\relative c'' {
\end{verbatim}
Modificare l’aspetto delle linee della griglia

L’aspetto delle linee della griglia può essere modificato sovrascrivendo alcune delle loro proprietà.

```
\score {
  \new ChoirStaff <<
  \new Staff {
    \relative c'' {
      \stemUp
      c'4. d8 e8 f g4
    }
  }
  \new Staff {
    \relative c {
      \override \Score.GrainLine.extra-offset = #'(0.0 . 1.0)
      \stemDown
      \clef bass
      \once \override \Score.GrainLine.thickness = #5.0
      c4
      \once \override \Score.GrainLine.thickness = #1.0
      g'4
      \once \override \Score.GrainLine.thickness = #3.0
      f'4
      \once \override \Score.GrainLine.thickness = #5.0
      e4
    }
  }
} >>
\layout {
  \context {
    \Staff
    \consists "Grid_point_engraver"
    \gridInterval = #(ly:make-moment 1/4)
  }
  \context {
    \Score
    \consists "Grid_line_span_engraver"
    \this moves them to the right half a staff space
  }
```
Grid lines: emphasizing rhythms and notes synchronization

Regular vertical lines can be drawn between staves to show note synchronization; however, in case of monophonic music, you may want to make the second stave invisible, and make the lines shorter like in this snippet.

```
\score {
  \new ChoirStaff {
    \relative c' <<
    \new Staff {
      \time 12/8
      \stemUp
      c4. d8 e8 f g4 f8 e8. d16 c8
    }
    \new Staff {
      % hides staff and notes so that only the grid lines are visible
      \hideNotes
      \hide Staff.BarLine
      \override Staff.StaffSymbol.line-count = #0
      \hide Staff.TimeSignature
      \hide Staff.Clef

      % dummy notes to force regular note spacing
      \once \override Score.GridLine.thickness = #4.0
      c8 c c
      \once \override Score.GridLine.thickness = #3.0
      c8 c c
      \once \override Score.GridLine.thickness = #4.0
      c8 c c
      \once \override Score.GridLine.thickness = #3.0
      c8 c c
    }
  }
}
\layout {
  \context {
    \Score
    \consists "Grid_line_span_engraver"
    % center grid lines horizontally below note heads
```
Martellato e strappato

Il martellato (hammer on) e lo strappato (pull off) si possono ottenere con le legature di portamento.

\new TabStaff { 
  \relative c' { 
    d4( e\2) 
    a( g) 
  } 
}

Martellato e strappato usando accordi

Quando il martellato o lo strappato si applicano a delle note in un accordo, viene disegnato un solo arco. Ma è possibile avere un “doppio arco” impostando la proprietà doubleSlurs su #t.

\new TabStaff { 
  \relative c' { 
    % chord hammer-on and pull-off 
    \set doubleSlurs = #t 
    <g' b>8( <a c> <g b>) 
  } 
}
Martellato e strappato usando le voci
L’arco del martellato o dello strappato è rivolto in alto nella prima e terza voce, mentre è rivolto in basso nella seconda e quarta voce.

```
\new TabStaff {
  \relative c' {
    << { \voiceOne g2( a ) }
    \ \ { \voiceTwo a,( b) }
    >> \oneVoice
  }
}
```

Rendere alcune linee del rigo più spesse delle altre
In ambito didattico può essere utile rendere più spesso una linea del rigo (per esempio, la linea centrale, o per sottolineare la linea della chiave di Sol). Per farlo si possono aggiungere altre linee e posizionarle molto vicino alla linea che deve essere evidenziata, usando la proprietà line-positions dell’oggetto StaffSymbol.

```
{ \override Staff.StaffSymbol.line-positions =
  #'(-4 -2 -.2 0 0.2 2 4)
  d'4 e' f' g'
}
```

Marking notes of spoken parts with a cross on the stem (Sprechstimme)
This example shows how to put crosses on stems. Mark the beginning of a spoken section with the \speakOn keyword, and end it with the \speakOff keyword.

```
\speakOn = {
  \override Stem.stencil =
  #(lambda (grob)
    (let* ((x-parent (ly:grob-parent grob x))
      (is-rest? (ly:grob? (ly:grob-object x-parent 'rest))))
    (if is-rest?
      empty-stencil
      (ly:stencil-combine-at-edge (ly:stem::print grob))
    ))
```

\( Y \)
\( (- \ (1y:\text{grob-property} \ \text{grob} \ 'direction)) \)
\( (\text{grob-interpret-markup} \ \text{grob} \ 
\text{(markup} \ #:\text{center-align} #:\text{fontsize} \ -4 \ 
\ #:\text{musicglyph} \ "\text{noteheads.s2cross}) \)) \)
\( -2.3))))) \}

\{ \text{\speakOff} = \}
\text{\speakOn} \text{\revert Stem stencil} \\
\text{\speakOn} \text{\revert Flag stencil} \\
\}

\text{\score \{}
\text{\new Staff \{}
\text{\relative c'' \{}
\text{a4 b a c} \\
\text{\speakOn} \\
\text{g4 f r g} \\
\text{b4 r d e} \\
\text{\speakOn} \\
\text{c4 a g f} \\
\}
\}
\}

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{example.pdf}
\end{figure}

**Measure counter**

This snippet provides a workaround for emitting measure counters using transparent percent repeats.

\<<
\text{\context Voice = "foo" \{}
\text{\clef bass} \\
\text{c4 r g r} \\
\text{c4 r g r} \\
\text{c4 r g r} \\
\text{c4 r g r} \\
\}
\text{\context Voice = "foo" \{}
\text{\set countPercentRepeats = ##t} \\
\text{\hide PercentRepeat} \\
\text{\override PercentRepeatCounter.staff-padding = #1} \\
\text{\repeat percent 4 \{ s1 \}} \\
\}
\>>
Measure spanner

Measure spanners are an alternate way to print annotated brackets. As opposed to horizontal brackets, they extend between two bar lines rather than two notes. The text is displayed in the center of the bracket.

```
\paper { tagline = ##f }

\layout {
  \context {
    \Staff
    \consists Measure_spanner_engraver
  }
}

<<
  \new Staff \relative c'' {
    \key d \minor
    R1*2
    \tweak text "Answer"
    \startMeasureSpanner
    \tuplet 3/2 8 {
      a16[ b c] d[ c b] c[ d e] f[ e d]
    }
    e8 a gis g
    fis f e d- d c b e
    \stopMeasureSpanner
  }
  \new Staff \relative c' {
    \key d \minor
    \tweak text "Subject"
    \tweak direction #DOWN
    \startMeasureSpanner
    \tuplet 3/2 8 {
      d16[ e f] g[ f e] f[ g a] bes[ a g]
    }
    a8 d cis c
    b bes a g- g f e a
    \stopMeasureSpanner
    \tweak text "Counter-subject"
    \tweak direction #DOWN
    \startMeasureSpanner
    f8 e a r r16 b, c d e fis g e
    a gis a b c fis, b a gis e a4 g8
    \stopMeasureSpanner
  }
>>
```
Numbering groups of measures

This snippet demonstrates the use of the Measure_counter_engraver to number groups of successive measures. Any stretch of measures may be numbered, whether consisting of repetitions or not.

The engraver must be added to the appropriate context. Here, a Staff context is used; another possibility is a Dynamics context.

The counter is begun with \startMeasureCount and ended with \stopMeasureCount. Numbering will start by default with 1, but this behavior may be modified by overriding the count-from property.

When a measure extends across a line break, the number will appear twice, the second time in parentheses.

\layout {
  \context {
    \Staff
    \consists \Measure_counter_engraver
  }
}

\new Staff {
  \startMeasureCount
  \repeat unfold 7 {
    c'4 d' e' f'
  }
  \stopMeasureCount
  \bar "||"
  g'4 f' e' d'
  \override Staff.MeasureCounter.count-from = #2
  \startMeasureCount
  \repeat unfold 5 {
    g'4 f' e' d'
  }
  g'4 f'

Positioning fingering indications precisely

Generally the options available for positioning the fingering of chords work well by default, but if one of the indications needs to be positioned more precisely the following tweak may be used. This is particularly useful for correcting the positioning when intervals of a second are involved.

\relative c' {
\set fingeringOrientations = #'(left)
\<c-1 d-2 a'-5>4\n\set tweak extra-offset #'(0 . 0.2)-2 a'-5>4
\set fingeringOrientations = #'(down)
\<c-1 d-2 a'-5>4\n\<c-1 d-2 a'-5>4\n\set tweak extra-offset #'(-1.2 . -1.8)-2 a'-5>4
\set fingeringOrientations = #'(down right up)
\<c-1 d-2 a'-5>4\n\set tweak extra-offset #'(-0.3 . 0)-2 a'-5>4
\set tweak extra-offset #'(-1 . 1.2)-2 a'-5>4
\set tweak extra-offset #'(-0.3 . 0)-2 a'-5>4
\set tweak extra-offset #'(-1 . 1.2)-2 a'-5>4
\set tweak extra-offset #'(-1.2 . 1.5)-2 a'-5>4
\set tweak extra-offset #'(-0.3 . 0)-2 a'-5>4
\set tweak extra-offset #'(-1 . 1.2)-2 a'-5>4
\set tweak extra-offset #'(-1.2 . 1.5)-2 a'-5>4
\set tweak extra-offset #'(-0.3 . 0)-2 a'-5>4
\set tweak extra-offset #'(-1 . 1.2)-2 a'-5>4
\set tweak extra-offset #'(-1.2 . 1.5)-2 a'-5>4
\set tweak extra-offset #'(-0.3 . 0)-2 a'-5>4
\set tweak extra-offset #'(-1 . 1.2)-2 a'-5>4
\set tweak extra-offset #'(-1.2 . 1.5)-2 a'-5>4
}
Posizionare il testo a margine dentro le legature di portamento

I testi a margine devono avere la proprietà outside-staff-priority impostata su false per poter apparire dentro le legature di portamento.

\relative c' { 
  \override TextScript.avoid-slu = #'inside 
  \override TextScript.outside-staff-priority = ##f 
  c2("\markup { \halign #-10 \natural } d4") c8 }

Printing text from right to left

It is possible to print text from right to left in a markup object, as demonstrated here.

{ 
  b1"\markup { \line \ingirumimusnotec } 
} 

f'\markup { \override #(text-direction . -1) 
  \line \ingirumimusnotec } 
}

String number extender lines

Make an extender line for string number indications, showing that a series of notes is supposed to be played all on the same string.

\stringNumberSpanner = 
  #(define-music-function (StringNumber) (string?) 
  #{ 
    \override TextSpanner.style = #'solid 
    \override TextSpanner.font-size = #-5 
    \override TextSpanner.bound-details.left.stencil-align-dir-y = #CENTER 
    \override TextSpanner.bound-details.left.text = 
    \markup { \circle \number $StringNumber } 
  })

\relative c { 
  \clef "treble_8" 
  \stringNumberSpanner "5" 
  \textSpannerDown 
  a8\startTextSpan
Using PostScript to generate special note head shapes

When a note head with a special shape cannot easily be generated with graphic markup, PostScript code can be used to generate the shape. This example shows how a parallelogram-shaped note head is generated.

\begin{verbatim}
parallelogram =
  #(ly:make-stencil (list 'embedded-ps
    "gsave
      currentpoint translate
      newpath
      0 0.25 moveto
      1.3125 0.75 lineto
      1.3125 -0.25 lineto
      0 -0.75 lineto
      closepath
      fill
      grestore"
    )
    (cons 0 1.3125)
    (cons -.75 .75))

myNoteHeads = \override NoteHead.stencil = \parallelogram
normalNoteHeads = \revert NoteHead.stencil

\relative c'' {
  \myNoteHeads
  g4 d'
  \normalNoteHeads
  <f, \tweak stencil \parallelogram b e>4 d
}
\end{verbatim}

Uso della proprietà whiteout

Qualsiasi oggetto grafico può essere posizionato sopra uno sfondo bianco per mascherare parti degli oggetti che si trovano sotto. Ciò può essere utile per migliorare l’aspetto delle collisioni in situazioni complesse in cui il riposizionamento degli oggetti è troppo difficile. Bisogna impostare esplicitamente la proprietà layer (livello) per controllare quali oggetti debbano essere mascherati dallo sfondo bianco.
In questo esempio la collisione della legatura di valore con l’indicazione di tempo viene migliorata mascherando la parte della legatura che incrocia l’indicazione di tempo impostando la proprietà `whiteout` di `TimeSignature`. Per farlo si sposta `TimeSignature` su un livello superiore a `Tie`, che viene lasciato al livello predefinito 1; e `StaffSymbol` viene spostato su un livello superiore a `TimeSignature` in modo che non venga mascherato.

```plaintext
\override Score.StaffSymbol.layer = #4
\override Staff.TimeSignature.layer = #3
b'2 b'~
\once \override Staff.TimeSignature.whiteout = ##t
\time 3/4
b' r4
```

\begin{figure}
\centering
\includegraphics[width=0.5\textwidth]{example.png}
\end{figure}
**Text**

Sezione “Text” in *Guida alla Notazione*

**Adding a QR code**

This snippet lets you draw a QR-code, for example to provide a link to the composer’s, or the music editor’s website. Actually encoding the URL into a QR-code is not done here (this just draws the QR-code from a grid of "black" or "white" values), but see the code for a short Python snippet you can use to avoid having to fill for each small square if it’s black or white.

%% (snippet author: Jean Abou Samra <jean@abou-samra.fr>)

\paper { tagline = ##f }

\begin{lyxcode}
\begin{verbatim}
\ifdef (index-map f . lsts)
"Applies \texttt{f} to corresponding elements of \texttt{lists}, just as \texttt{map}, providing an additional counter starting at zero. \texttt{f} needs to have the counter in its arguments like \texttt{\{index-map \{lambda \{i \arg\} \texttt{<body>\} \lists\}\}}"

(let loop ((lsts lsts)
                  (acc '())
                  (i 0))
   (if (not null? lsts)
       (reverse! acc)
       (loop (map cdr lsts)
             (cons (apply f i (map car lsts))
                   acc)
             (1+ i))))

\demifunction (define-markup-command \{qr-code layout props data\}) \{string?\}
\ifdef (define-markup-command qr-code layout props data) \{string?\}
\ifdef (define-markup-command qr-code layout props data) \{string?\}
\ifdef (define-markup-command qr-code layout props data) \{string?\}
\ifdef (define-markup-command qr-code layout props data) \{string?\}
\ifdef (define-markup-command qr-code layout props data) \{string?\}

(let* (lines (reverse (remove string-null? (map string-trim-both (string-split data #\newline))))))

(n (length lines))
(square-width (/ width n))
(box (make-filled-box-stencil `(0 . ,square-width)
                               `(0 . ,square-width))))

;; Build the final qr-code-stencil from line-stencils list
(apply ly:stencil-add
       ;; Get a list of line-stencils
       (index-map
        (lambda (i line)
           ;; Build a line-stencil from square-stencils list
           (apply ly:stencil-add
                  ;; Get a list of (already translated) square-stencils
                  ;; per line
                  (index-map
                   (apply ly:stencil-add
                          ;; Get a list of (already translated) square-stencils
                          ;; per line
                          (index-map
                           (apply ly:stencil-add
                                  ;; Get a list of (already translated) square-stencils
                                  ;; per line
                                  (index-map
                               `(0 . ,square-width)
                               `(0 . ,square-width)))))))))
\end{verbatim}
\end{lyxcode}
(lambda (j char)
  (ly:stencil-translate
   (stencil-with-color
    box
    (case char
      ((#\0) white)
      ((#\1) black)
      (else
       (ly:warning
        "unrecognized character ~a, should be 0 or 1" char)
       red)))
    (cons (* j square-width)
      (* i square-width)))
   (string->list line)))))

%{ A string representation of the QR code. 0 means white, 1 means black. You can generate this automatically using Python and the pyqrcode module ("pip install pyqrcode"). Use this line of code in a Python prompt:

```python
>>> import pyqrcode; print(pyqrcode.create("https://lilypond.org").text(quiet_zone=0))
```
%

lilypondDotOrg = "11111100111000111110101111111 100000100100000101111010000001 101110100110001000011000101101101 101110100110000011100011111111111 000000011100011111110100000000 001100110110001100011001000000 1010100111100010000011111111111 0000000101100100101110111111111 01010001011000011100101010011110 0101000110010001000110111000001 01110110100110000111010111010110 01010001110001000001100011011101 01010001110001000011001110111010 110010010100100100001111111111 00000001011001001011101111111110 01010001110001000011001110111010 11111110111011010110110111110"
Adding markups in a tablature

By default markups do not show in a tablature.

To make them appear, simply use the command \revert TabStaff.TextScript.stencil

%% http://lsr.di.unimi.it/LSR/Item?id=919
% by P.P.Schneider on June 2014

\begin{verbatim}
\revert TabStaff.TextScript.stencil

\score { 
  \new TabStaff { 
    \repeat unfold 2 \<< \high \ll \low \ll \pulse \>> 
  } 
  \layout { 
    \context { 
      \TabStaff 
    } 
  } 

\end{verbatim}
Adding the current date to a score

With a little Scheme code, the current date can easily be added to a score.

\% first, define a variable to hold the formatted date:
date = #(strftime "%d-%m-%Y" (localtime (current-time)))

\% use it in the title block:
\header { title = "Including the date!" subtitle = \date }

\score { \relative c' { c4 c c c } }

\% and use it in a \markup block:
\markup { \date }

\n
Including the date!
18-11-2023

\n
18-11-2023
Adjusting lyrics vertical spacing

This snippet shows how to bring the lyrics line closer to the staff.

```
% Default layout:
<
\new Staff \new Voice = melody \relative c' {
  c4 d e f
g4 f e d
c1}
\new Lyrics \lyricsto melody { aa aa aa aa aa aa aa aa }

\new Staff {
  \new Voice = melody \relative c' {
    c4 d e f
g4 f e d
c1}
}

% Reducing the minimum space below the staff and above the lyrics:
\new Lyrics \with {
  \override VerticalAxisGroup.nonstaff-relatedstaff-spacing = #'((basic-distance . 1))
}
\lyricsto melody { aa aa aa aa aa aa aa aa }
```

Aligning and centering instrument names

The horizontal alignment of instrument names is tweaked by changing the `Staff.InstrumentName.self-alignment-X` property. The \layout variables `indent` and `short-indent` define the space in which the instrument names are aligned before the first and the following systems, respectively.

```
\paper { left-margin = 3\cm }

\score {
  \new StaffGroup <<

  \new Staff \with {
    \override InstrumentName.self-alignment-X = #LEFT
    instrumentName = \markup {\left-column {
      "Left aligned"
      "instrument name"
    }}
shortInstrumentName = "Left"
}
{ c''1 \break c''1 }
\new Staff \with {
  \override InstrumentName.self-alignment-X = #CENTER
  instrumentName = \markup \center-column { Centered "instrument name" }
  shortInstrumentName = "Centered"
}
{ g'1 g'1}
\new Staff \with {
  \override InstrumentName.self-alignment-X = #RIGHT
  instrumentName = \markup \right-column { "Right aligned" "instrument name" }
  shortInstrumentName = "Right"
}
{ e'1 e'1 }

\layout {
  \ragged-right = ##t
  indent = 4\cm
  short-indent = 2\cm
}
Aligning syllables with melisma

By default, lyrics syllables that start a melisma are left aligned on their note. The alignment can be altered using the \lyricMelismaAlignment property.

\score {
<<
\new Staff {
  \relative c''
  \new Voice = "vocal" {
    c d-\markup default d e
    c d-\markup "right aligned" d e
    c d-\markup "center aligned" d e
    c d-\markup "reset to default" d e
  }
}
\new Lyrics \lyricsto "vocal" {
  word word word
  \set lyricMelismaAlignment = #RIGHT
  word word word
  \set lyricMelismaAlignment = #CENTER
  word word word
  \unset lyricMelismaAlignment
  word word word
}
>>
}

Blanking staff lines using the \whiteout command

The \whiteout command underlays a markup with a white box. Since staff lines are in a lower layer than most other grobs, this white box will not overlap any other grob.

\layout {
  ragged-right = ##f
}

\relative c' {
Center text below hairpin dynamics

This example provides a function to typeset a hairpin (de)crescendo with some additional text below it, such as “molto” or “poco”. The added text will change the direction according to the direction of the hairpin. The Hairpin is aligned to DynamicText.

The example also illustrates how to modify the way an object is normally printed, using some Scheme code.

```
\paper { \tagline = \##f }

\hairpinWithCenteredText =
#(define-music-function (text) (markup?))
#
\once \override Voice.Hairpin.after-line-breaking =
#(lambda (grob)
  (let* ((stencil (ly:hairpin::print grob))
         (par-y (ly:grob-parent grob Y))
         (dir (ly:grob-property par-y 'direction))
         (staff-line-thickness
          (ly:output-def-lookup (ly:grob-layout grob) 'line-thickness))
         (new-stencil (ly:stencil-aligned-to
do-(ly:stencil-combine-at-edge
            (ly:stencil-aligned-to stencil X CENTER)
            Y dir
            (ly:stencil-aligned-to
do-(ly:grob-interpret-markup
do-(ly:make-fontsize-markup
do-(\magnification->font-size
            (+ (ly:staff-symbol-staff-space grob)
            (/ staff-line-thickness 2)))
do-(text)) X CENTER))
do-))
do-(staff-space (ly:output-def-lookup
do-(ly:grob-layout grob) 'staff-space))
do-(par-x (ly:grob-parent grob X))
do-(dyn-text (grob::has-interface par-x 'dynamic-text-interface))
do-(dyn-text-stencil-x-length
do-(if dyn-text
            (interval-length
            (ly:stencil-extent (ly:grob-property par-x 'stencil) X))
do-0))
do-(x-shift
```
Changing ottava text

Internally, \texttt{\textbackslash ottava} sets the properties \texttt{ottavation} (for example, to \texttt{8va} or \texttt{8vb}) and \texttt{middleCPosition}. To override the text of the bracket, set \texttt{ottavation} after invoking \texttt{\textbackslash ottava}.

Short text is especially useful when a brief ottava is used.

{\c'2}
Changing the default text font family

The default font families for text can be overridden with `make-pango-font-tree`.

```
%{
You may have to install additional fonts.

Red Hat Fedora

dejavu-fonts-all

Debian GNU/Linux, Ubuntu

fonts-dejavu-core
fonts-dejavu-extra

%}
```

```
\paper {
  \% change for other default global staff size.
  myStaffSize = #20
  \%
  \run
    lilypond -dshow-available-fonts
    to show all fonts available in the process log.
  \%
}
```

```
#(define fonts
  (make-pango-font-tree "DejaVu Serif"
    "DejaVu Sans"
    "DejaVu Sans Mono"
    (/ myStaffSize 20)))
}
```

```
{ g''''4
\markup {
  DejaVu Serif: \bold bold
  \italic italic
  \italic \bold { bold italic }
}
Combining dynamics with markup texts

Some dynamics may involve text indications (such as “più forte” or “piano subito”). These can be produced using a \markup block.

\markup \piuF = \markup { \italic \pi`u \ dynamic f }

\layout { \ragged-right = ##f }

\relative c' { c2\f c-\piuF }

Combining due parti sullo stesso rigo

Lo strumento di unione delle parti (il comando \partCombine) permette di combinare varie parti sullo stesso rigo. Indicazioni testuali come “solo” e “a2” sono aggiunte automaticamente; per toglierle basta impostare la proprietà printPartCombineTexts su f. Per le partiture vocali (inni), non c’è bisogno di aggiungere i testi “solo/a2”, quindi dovrebbero essere disattivati. Tuttavia potrebbe convenire non usarlo se c’è una qualche parte solista, perché non verrebbe indicata. In tali casi è preferibile usare la notazione polifonica normale.

Questo frammento illustra i tre modi con cui due parti possono essere stampate su uno stesso rigo: normale polifonia, \partCombine senza testo e \partCombine con testo.

%% Combining pedal notes with clef changes
\musicDown = \relative c'' { 
  g4 e4. (d8) c4 |
  r2 g'4( f8 e) |
  d2 \stemDown a 
}

\score { 
  << 
   \new Staff \with { instrumentName = "Standard polyphony" } 
   \new Staff \with { 
      instrumentName = "PartCombine without text" 
      printPartCombineTexts = ##f 
   }  
  \partCombine \musicUp \musicDown 
  \new Staff \with { instrumentName = "PartCombine with text" } 
  \partCombine \musicUp \musicDown 
  >> 
  \layout { 
    indent = 6.0\cm 
    \context { 
      \Score 
      \override SystemStartBar-collapse-height = #30 
    } 
  } 
}
Creating "real" parenthesized dynamics

Although the easiest way to add parentheses to a dynamic mark is to use a \markup block, this method has a downside: the created objects will behave like text markups, and not like dynamics.

However, it is possible to create a similar object using the equivalent Scheme code (as described in the Notation Reference), combined with the make-dynamic-script function. This way, the markup will be regarded as a dynamic, and therefore will remain compatible with commands such as \dynamicUp or \dynamicDown.

```plaintext
paren =  
#(define-event-function (dyn) (ly:event?))  
(make-dynamic-script  
  #{ \markup \concat  
      \normal-text \italic \fontsize #2 (  
      \pad-x #0.2 #(ly:music-property dyn 'text)  
      \normal-text \italic \fontsize #2 )  
  }  
  #}))

\relative c'' {  
c4/paren/f c c \dynamicUp c/paren/p
}

Creating text spanners

The \startTextSpan and \stopTextSpan commands allow the creation of text spanners as easily as pedal indications or octavations. Override some properties of the TextSpanner object to modify its output.

```plaintext
\paper { ragged-right = ##f }

\relative c'' {  
\override TextSpanner.bound-details.left.text = #'bla
\override TextSpanner.bound-details.right.text = #'blu
a4 \startTextSpan
b4 c
a4 \stopTextSpan

\override TextSpanner.style = #'line
\once \override TextSpanner.bound-details.left.stencil-align-dir-y = #CENTER
a4 \startTextSpan
b4 c
a4 \stopTextSpan

\override TextSpanner.style = #'dashed-line
\override TextSpanner.bound-details.left.text =  
\markup { \draw-line #'(0 . 1) }
```
Demonstrating all headers

All header fields with special meanings.

```
\header {
  \copyright = "copyright"
  \title = "title"
  \subtitle = "subtitle"
  \composer = "composer"
  \arranger = "arranger"
  \instrument = "instrument"
  \meter = "meter"
  \opus = "opus"
  \piece = "piece"
  \poet = "poet"
  \texidoc = "All header fields with special meanings."
  \copyright = "public domain"
  \enteredby = "jcn"
  \source = "urtext"
}

\layout {
  \ragged-right = ##f
}

\score {
  \relative c' { c1 | c | c | c }
}

\score {
  \relative c' { c1 | c | c | c }
}
Embedding native PostScript in a \markup block

PostScript code can be directly inserted inside a \markup block.

% PostScript is a registered trademark of Adobe Systems Inc.

\relative c' {
  a4 \markup { \postscript "3 4 moveto 5 3 rlineto stroke" }
  \markup { \postscript "[ 0 1 ] 0 setdash 3 5 moveto 5 -3 rlineto stroke " }
  b4 \markup { \postscript "3 4 moveto 0 0 1 2 8 4 20 3.5 rcurveto stroke" }
  s2
  a"1
}

Embedding native PostScript in a \markup block

PostScript code can be directly inserted inside a \markup block.

% PostScript is a registered trademark of Adobe Systems Inc.

\relative c' {
  a4 \markup { \postscript "3 4 moveto 5 3 rlineto stroke" }
  \markup { \postscript "[ 0 1 ] 0 setdash 3 5 moveto 5 -3 rlineto stroke " }
  b4 \markup { \postscript "3 4 moveto 0 0 1 2 8 4 20 3.5 rcurveto stroke" }
  s2
  a"1
}

Embedding native PostScript in a \markup block

PostScript code can be directly inserted inside a \markup block.

% PostScript is a registered trademark of Adobe Systems Inc.

\relative c' {
  a4 \markup { \postscript "3 4 moveto 5 3 rlineto stroke" }
  \markup { \postscript "[ 0 1 ] 0 setdash 3 5 moveto 5 -3 rlineto stroke " }
  b4 \markup { \postscript "3 4 moveto 0 0 1 2 8 4 20 3.5 rcurveto stroke" }
  s2
  a"1
}

Embedding native PostScript in a \markup block

PostScript code can be directly inserted inside a \markup block.

% PostScript is a registered trademark of Adobe Systems Inc.

\relative c' {
  a4 \markup { \postscript "3 4 moveto 5 3 rlineto stroke" }
  \markup { \postscript "[ 0 1 ] 0 setdash 3 5 moveto 5 -3 rlineto stroke " }
  b4 \markup { \postscript "3 4 moveto 0 0 1 2 8 4 20 3.5 rcurveto stroke" }
  s2
  a"1
}

Embedding native PostScript in a \markup block

PostScript code can be directly inserted inside a \markup block.

% PostScript is a registered trademark of Adobe Systems Inc.

\relative c' {
  a4 \markup { \postscript "3 4 moveto 5 3 rlineto stroke" }
  \markup { \postscript "[ 0 1 ] 0 setdash 3 5 moveto 5 -3 rlineto stroke " }
  b4 \markup { \postscript "3 4 moveto 0 0 1 2 8 4 20 3.5 rcurveto stroke" }
  s2
  a"1
}
Formattazione delle sillabe del testo vocale

La modalità markup può essere usata per formattare le singole sillabe del testo vocale.

```latex
mel = \texttt{\relative c'' \{ c4 c c c \}}
lyr = \texttt{\lyricmode \{ Lyrics \markup { \italic can } \markup { \with-color \texttt{#red} \texttt{contain} } \markup { \texttt{\fontsize \texttt{#8} \texttt{\bold Markup!} } } \}}

<< \texttt{\new Voice = melody \mel} \texttt{\new Lyrics \lyricsto melody \lyr} >>
```

![Musical notation with lyrics]

Lyrics can contain Markup!

How to put ties between syllables in lyrics

This can be achieved by separating those syllables by tildes.

```latex\texttt{\lyrics \{ wa~o~a \}}```

`wa\_o\_a`

Allineamento del testo vocale

L’allineamento orizzontale del testo vocale si imposta attraverso la proprietà `self-alignment-X` dell’oggetto `LyricText`. #1 è destra, #0 è centro e #1 è destra; si possono usare anche #LEFT, #CENTER e #RIGHT.

```latex
\texttt{\layout \{ \texttt{ragged-right} = ##f \}}\texttt{\relative c'' \{ \texttt{cl} \texttt{cl} \texttt{cl} \texttt{cl} \}}
```

`\texttt{\addlyrics \{ \texttt{\once \override LyricText.self-alignment-X} = \texttt{#LEFT} \quad \texttt{This is left-aligned} \}}
\texttt{\once \override LyricText.self-alignment-X} = \texttt{#CENTER} \quad \texttt{This is centered} \}
\texttt{\once \override LyricText.self-alignment-X} = \texttt{#1} \quad \texttt{This is right-aligned} \}
```

![Musical notation with alignment]

This is left-aligned This is centered This is right-aligned
Il y avait en Westphalie, dans le château de M. le baron de Thunder-ten-tronckh, un jeune garçon à qui la nature avait donné les mœurs les plus douces. Sa physionomie annonçait son âme. Il avait le jugement assez droit, avec l'esprit le plus c'est, je crois, pour cette raison qu'on le nommait Candide. Les anciens domestiques de la maison soupçonnaient qu'il était fils de la sœur de monsieur le baron et d'un bon et honnête gentilhomme du voisinage, que cette demoiselle ne voulut jamais épouser parce qu'il n'avait pu prouver que soixante et onze quartiers, et que le reste de son arbre généalogique avait été perdu par l'injure du temps.


\vspace {.3}
\paragraph { Monsieur le baron était un des plus puissants seigneurs de la Westphalie, car son château avait une porte et des fenêtres. Sa grande salle même était ornée d'une tapisserie. Tous les chiens de ses basses-cours composaient une meute dans le \concat { besoin \hspace {.3} ; } ses palefreniers étaient ses \concat { piqueurs \hspace {.3} ; } le vicaire du village était son grand-aumônier. Ils l'appelaient tous monseigneur, et ils riaient quand il faisait des contes.


Il y avait en Westphalie, dans le château de M. le baron de Thunder-ten-tronckh, un jeune garçon à qui la nature avait donné les mœurs les plus douces. Sa physionomie annonçait son âme. Il avait le jugement
assez droit, avec l'esprit le plus simple ; c'est, je crois,
pour cette raison qu'on le nommait Candide. Les
anciens domestiques de la maison soupçonnaient qu'il
était fils de la sœur de monsieur le baron et d'un bon et
honnête gentilhomme du voisinage, que cette
demoiselle ne voulut jamais épouser parce qu'il n'avait
pu prouver que soixante et onze quartiers, et que le
reste de son arbre généalogique avait été perdu par
l'injure du temps.

Monsieur le baron était un des plus puissants
seigneurs de la Westphalie, car son château avait une
porte et des fenêtres. Sa grande salle même était ornée
da tapisserie. Tous les chiens de ses basses-cours
composaient une meute dans le besoin ; ses
palefreniers étaient ses piqueurs ; le vicaire du village
était son grand-aumônier. Ils l'appelaient tous
monseigneur, et ils riaient quand il faisait des contes.
Testo a margine delle pause multiple

Il testo a margine di una pausa multipla viene centrato sopra o sotto di essa. Se il testo è lungo, la misura non si espanderà. Per espandere la pausa multipla in modo che si allinei col testo, conviene usare un accordo vuoto con del testo attaccato prima della pausa multipla.

Il testo così attaccato a una nota spaziatrice viene allineato a sinistra della posizione in cui la nota sarebbe posta nella misura, ma se la lunghezza della misura è determinata dalla lunghezza del testo, il testo verrà centrato.

\relative c' {
  \compressMMRests {
    \textLengthOn
    <>\markup { [MAJOR GENERAL] }
    R1*19
    <>\markup { italic { Cue: ... it is yours } }
    <>\markup { A }
    R1*30\markup { [MABEL] }
    \textLengthOff
    c4\markup { CHORUS } d f c
  }
}

Of the ubiquity of markup objects

Text objects are entered either as simple strings between double quotes or as \markup blocks that can accept a variety of advanced text formatting and graphical enhancements.

As such, markup blocks may be used:
• in any TextScript object (attached to notes with -, ^ or _),
• any TextMark introduced with the \textMark or \textEndMark command, or other similar objects such as MetronomeMark introduced with \tempo,
• as standalone markup blocks, entered at the top level outside of any \score block,
• in any definition inside the \header block (e.g. title, subtitle, composer) or in some variables defined inside the \paper block such as evenHeaderMarkup for page numbers.

\markup may additionally be used for lyrics, in chord names, and as dynamics. In fact, it is possible to use \markup to customize the appearance of virtually any object, as demonstrated in this example using various methods.


\paper {
  \paper-width = 8\cm \paper-height = 8\cm
}
\header {
  title = \markup "Header"
  tagline = \markup "(tagline)"
}
\markup "Top-level markup"
\text{Top-level markup}

\textbf{MetronomeMark}  
\textbf{TextMark}  
C\textsuperscript{maj7}  
\textbf{TextScript}  
\textbf{LyricText}  
\textbf{DynamicText}

\textbf{Outputting the version number}

It is possible to print the version number of LilyPond in markup.
\texttt{\textbackslash markup \{ Processed with LilyPond version \#(lilypond-version) \}}

Processed with LilyPond version 2.24.3

\textbf{Modello per pianoforte con testo al centro}

Invece di destinare un rigo a parte alla linea melodica e al suo testo, è possibile collocare il testo al centro di un doppio pentagramma per pianoforte.
\texttt{upper = \texttt{\relative c' \{}}
Stampare i numeri di battuta a intervalli regolari variabili

Tramite la funzione di contesto \{set-bar-number-visibility\} si possono modificare gli intervalli dei numeri di battuta.

\relative c' {  
  \override Score.BarNumber.break-visibility = #end-of-line-invisible  
  \context Score \applyContext #(set-bar-number-visibility 4)  
  \repeat unfold 10 c'1  
  \context Score \applyContext #(set-bar-number-visibility 2)  
  \repeat unfold 10 c  
}
Stampare le indicazioni su ogni rigo
Sebbene le indicazioni testuali siano di norma collocate solo sopra il rigo più alto, è possibile farle apparire su ogni rigo.

\score {
  <<
    \new Staff { \mark \default c''1 \textMark "molto" c''}
    \new Staff { \mark \default c'1 \textMark "molto" c'}
  >>
\layout {
  \context {
    \Score
    \remove Mark_engraver
    \remove Text_mark_engraver
    \remove Staff_collecting_engraver
  }
  \context {
    \Staff
    \consists Mark_engraver
    \consists Text_mark_engraver
    \consists Staff_collecting_engraver
  }
}
}

Printing text from right to left
It is possible to print text from right to left in a markup object, as demonstrated here.

{\b1-\markup {
  \line { ingirumimusnocte}
}}
{\f'\markup {
  \override #'(text-direction . -1)
  \line { ingirumimusnocte}
}}
Putting lyrics inside the staff

Lyrics can be moved vertically to place them inside the staff. The lyrics are moved with \override LyricText.extra-offset = #'(0 . dy) and there are similar commands to move the extenders and hyphens. The offset needed is established with trial and error.

<<
\new Staff <<
\new Voice = "voc" \relative c' { \stemDown a bes c8 b c4 }
>>
\new Lyrics \with {
    \override LyricText.extra-offset = #'(0 . 8.6)
    \override LyricExtender.extra-offset = #'(0 . 8.6)
    \override LyricHyphen.extra-offset = #'(0 . 8.6)
} \lyricsto "voc" { La la -- la __ _ la }
>>

Testo separato su due colonne

Il testo separato può essere disposto su varie colonne con i comandi di \markup:

\markup {
    \fill-line {
        \hspace #1
        \column {
            \line { O sacrum convivium }
            \line { in quo Christus sumitur, }
            \line { recolitur memoria passionis ejus, }
            \line { mens impletur gratia, }
            \line { futurae gloriae nobis pignus datur. }
            \line { Amen. }
        }
        \hspace #2
        \column \italic {
            \line { O sacred feast }
            \line { in which Christ is received, }
            \line { the memory of His Passion is renewed, }
            \line { the mind is filled with grace, }
            \line { and a pledge of future glory is given to us. }
            \line { Amen. }
        }
        \hspace #1
    }
}
O sacrum convivium
in quo Christus sumitur,
recolitur memoria passionis ejus,
mens impletur gratia,
futurae gloriae nobis pignus datur.
    Amen.

O sacred feast
in which Christ is received,
the memory of His Passion is renewed,
the mind is filled with grace,
and a pledge of future glory is given to us.
    Amen.

String number extender lines
Make an extender line for string number indications, showing that a series of notes is supposed
to be played all on the same string.

```
stringNumberSpanner =
    #(define-music-function (StringNumber) (string?)
        #{
            \override TextSpanner.style = #'solid
            \override TextSpanner.font-size = #-5
            \override TextSpanner.bound-details.left.stencil-align-dir-y = #CENTER
            \override TextSpanner.bound-details.left.text =
                markup { \circle \number $StringNumber }
        }\)

relative c {
    \clef "treble_8"
    \stringNumberSpanner "5"
    \textSpannerDown
    a8\startTextSpan
    b c d e f\stopTextSpan
    \stringNumberSpanner "4"
    g\startTextSpan
    b e s 4 a g2\stopTextSpan
}
```

Three-sided box
This example shows how to add a markup command to get a three sided box around some text
(or other markup).

```
% New command to add a three sided box, with sides north, west and south
% Based on the box-stencil command defined in scm/stencil.scm
% Note that ";;" is used to comment a line in Scheme
#(define-public (NWS-box-stencil stencil thickness padding)
    "Add a box around STENCIL, producing a new stencil."
    (let* ((x-ext (interval-widen (ly:stencil-extent stencil X) padding))
            (y-ext (interval-widen (ly:stencil-extent stencil Y) padding)))
```
(y-rule (make-filled-box-stencil (cons 0 thickness) y-ext))
(x-rule (make-filled-box-stencil
           (interval-widen x-ext thickness) (cons 0 thickness))))

;; (set! stencil (ly:stencil-combine-at-edge stencil X 1 y-rule padding))
(set! stencil (ly:stencil-combine-at-edge stencil X LEFT y-rule padding))
(set! stencil (ly:stencil-combine-at-edge stencil Y UP x-rule 0.0))
(set! stencil (ly:stencil-combine-at-edge stencil Y DOWN x-rule 0.0))
(stencil))

% The corresponding markup command, based on the \box command defined
% in scm/define-markup-commands.scm
#(define-markup-command (NWS-box layout props arg) (markup?)
   #:properties ((thickness 0.1) (font-size 0) (box-padding 0.2))
   "Draw a box round @var{arg}. Looks at @code{thickness},
   @code{box-padding} and @code{font-size} properties to determine line
   thickness and padding around the markup."
   (let ((pad (* (magstep font-size) box-padding))
          (m (interpret-markup layout props arg)))
       (NWS-box-stencil m thickness pad)))

% Test it:

\relative c' {
  c1\ markup { \NWS-box ABCD }
  c1\ markup { \NWS-box \note {4} #1.0 }
}

\includegraphics{ABCD.jpg}

UTF-8

Various scripts may be used for texts (like titles and lyrics) by entering them in UTF-8 encoding,
and using a Pango based backend. Depending on the fonts installed, this fragment will render
Bulgarian (Cyrillic), Hebrew, Japanese and Portuguese.

%{
You may have to install additional fonts.

Red Hat Fedora

   linux-libertine-fonts (Latin, Cyrillic, Hebrew)
   google-noto-serif-jp-fonts (Japanese)

Debian GNU/Linux, Ubuntu

   fonts-linuxlibertine (Latin, Cyrillic, Hebrew)
   fonts-noto-cjk (Japanese)
%}

% 'Linux Libertine' fonts also contain Cyrillic and Hebrew glyphs.
\paper {
\{define fonts
(\set-global-fonts
  #:roman "Linux Libertine O, Noto Serif CJK JP, Noto Serif JP"
))
}

bulgarian = \lyricmode {
   Жълтата дяля беше щастлива, че пухът, който цъфна, замръзна като гъд.
}

hebrew = \lyricmode {

}

japanese = \lyricmode {

}

portuguese = \lyricmode {
   à vo -- cê um a can -- ça o le -- gal
}

\relative c' {
   c2 d
   e2 f
   g2 f
   e2 d
}
\addlyrics { \bulgarian }
\addlyrics { \hebrew }
\addlyrics { \japanese }
\addlyrics { \portuguese }

% "a nice song for you"

\addlyrics { \ Relative c' { c2 d e2 f g2 f e2 d } \addlyrics { \bulgarian } \addlyrics { \hebrew } \addlyrics { \japanese } \addlyrics { \portuguese } }
Modello per gruppo vocale con testo allineato sotto e sopra i righi

Questo modello è fondamentalmente analogo al semplice modello “Complesso vocale”, con l’unica differenza che qui tutti i versi del testo sono posizionati usando alignAboveContext e alignBelowContext.

```
global = {
    \key c \major
    \time 4/4
}

sopMusic = \relative c' { 
    c4 c c8[( b)] c4
}

sopWords = \lyricmode { 
    hi hi hi hi
}

altoMusic = \relative c' { 
    e4 f d e
}

altoWords = \lyricmode { 
    ha ha ha ha
}

tenorMusic = \relative c' { 
    g4 a f g
}

tenorWords = \lyricmode { 
    hu hu hu hu
}

bassMusic = \relative c { 
    c4 c g c
}

bassWords = \lyricmode { 
    ho ho ho ho
}

\score { 
    \new ChoirStaff << 
        \new Staff = "women" << 
            \new Voice = "sopranos" { \voiceOne << \global \sopMusic >> }
```
\new Voice = "altos" { \voiceTwo << \global \altoMusic >> }

\new Lyrics \with { alignAboveContext = #'women } \lyricsto "sopranos" \sopWords
\new Lyrics \with { alignBelowContext = #'women } \lyricsto "altos" \altoWords
% we could remove the line about this with the line below, since
% we want the alto lyrics to be below the alto Voice anyway.
% \new Lyrics \lyricsto "altos" \altoWords

\new Staff = "men" <<
\clef bass
\new Voice = "tenors" { \voiceOne << \global \tenorMusic >> }
\new Voice = "basses" { \voiceTwo << \global \bassMusic >> }

\new Lyrics \with { alignAboveContext = #'men } \lyricsto "tenors" \tenorWords
\new Lyrics \with { alignBelowContext = #'men } \lyricsto "basses" \bassWords
% again, we could replace the line above this with the line below.
% \new Lyrics \lyricsto "basses" \bassWords

nu

Volta text markup using repeatCommands

Though volte are best specified using \repeat volta, the context property repeatCommands
must be used in cases where the volta text needs more advanced formatting with \markup.

Since repeatCommands takes a list, the simplest method of including markup is to use an
identifier for the text and embed it in the command list using the Scheme syntax #(list (list
'volta textIdentifier)). Start- and end-repeat commands can be added as separate list
elements:
voltaAdLib = \markup { 1. 2. 3... \text \italic { ad lib. } }

\relative c'' { c1 \set Score.repeatCommands = #(list (list 'volta voltaAdLib) 'start-repeat)
c4 b d e \set Score.repeatCommands = #'(volta #f) (volta "4.") end-repeat)
f1
Vocal music

Sezione “Vocal music” in *Guida alla Notazione*

**Un ambitus per voce**

L’ambitus può essere specificato per voce. In tal caso occorre spostarlo manualmente per evitare collisioni.

\new Staff <<
\new Voice \with {
  \consists "Ambitus_ engraver"
} \relative c' \{ 
  \override Ambitus.X-offset = #2.0
  \c4 a d e
  f1
}
\new Voice \with {
  \consists "Ambitus_ engraver"
} \relative c' \{
  \voiceTwo
  es4 f g as
  b1
}
>>

Adding indicators to staves which get split after a break

This snippet defines the \splitStaffBarLine, convUpStaffBarLine and convDownStaffBarLine commands. These add arrows at a bar line, to denote that several voices sharing a staff will each continue on a staff of their own in the next system, or that voices split in this way recombine.

#(define-markup-command (arrow-at-angle layout props angle-deg length fill)
  (number? number? boolean?)
  (let* ( 
    (PI-OVER-180 (/ (atan 1 1) 34))
    (degrees->radians (lambda (degrees) (* degrees PI-OVER-180)))
    (angle-rad (degrees->radians angle-deg))
    (target-x (* length (cos angle-rad)))
    (target-y (* length (sin angle-rad))))
  (interpret-markup layout props
    markup
    #:translate (cons (/ target-x 2) (/ target-y 2))
    #:rotate angle-deg
    #:translate (cons (/ length -2) 0)
    #:concat (#:draw-line (cons length 0)
              #:arrow-head X RIGHT fill))))
\\n\text{Vocal music}\n
\text{splitStaffBarLineMarkup} = \text{\textbackslash markup \textbackslash \textbackslash with-dimensions \#'\textbackslash (0 . 0) \#'\textbackslash (0 . 0) \{}\n\text{\textbackslash combine}\n\text{\textbackslash arrow-at-angle \#45 \#(sqrt 8) \#t}\n\text{\textbackslash arrow-at-angle \#-45 \#(sqrt 8) \#t}\n\}\n
\text{splitStaffBarLine} = \{}\n\text{\once \textbackslash override Staff.BarLine.stencil} = \n\text{\#(lambda (grob) }\n\text{ (ly:stencil-combine-at-edge}\n\text{ (ly:bar-line::print grob) }\n\text{ X RIGHT}\n\text{ (grob-interpret-markup grob splitStaffBarLineMarkup) }\n\text{ 0))}\n\text{ \break}\n\}\n
\text{convDownStaffBarLine} = \{}\n\text{\once \textbackslash override Staff.BarLine.stencil} = \n\text{\#(lambda (grob) }\n\text{ (ly:stencil-combine-at-edge}\n\text{ (ly:bar-line::print grob) }\n\text{ X RIGHT}\n\text{ (grob-interpret-markup grob \{}\n\text{ \textbackslash markup\textbackslash \textbackslash with-dimensions \#'\textbackslash (0 . 0) \#'\textbackslash (0 . 0) \{}\n\text{ \textbackslash translate \#'\textbackslash (0 . -.13)\textbackslash arrow-at-angle \#-45 \#(sqrt 8) \#t}\n\text{ \})\}\n\text{ 0))}\n\text{ \break}\n\}\n
\text{convUpStaffBarLine} = \{}\n\text{\once \textbackslash override Staff.BarLine.stencil} = \n\text{\#(lambda (grob) }\n\text{ (ly:stencil-combine-at-edge}\n\text{ (ly:bar-line::print grob) }\n\text{ X RIGHT}\n\text{ (grob-interpret-markup grob \{}\n\text{ \textbackslash markup\textbackslash \textbackslash with-dimensions \#'\textbackslash (0 . 0) \#'\textbackslash (0 . 0) \{}\n\text{ \textbackslash translate \#'\textbackslash (0 . .14)\textbackslash arrow-at-angle \#45 \#(sqrt 8) \#t}\n\text{ \})\}\n\text{ 0))}\n\text{ \break}\n\}\n
\text{\paper \{}\n\text{ \ragged-right = \#t}\n\text{ \short-indent = 10\textbackslash mm}\n\text{ \}}
separateSopranos = {
    \set Staff.instrumentName = "AI AII"
    \set Staff.shortInstrumentName = "AI AII"
    \splitStaffBarLine
    \change Staff = "up"
}

convSopranos = {
    \convDownStaffBarLine
    \change Staff = "shared"
    \set Staff.instrumentName = "S A"
    \set Staff.shortInstrumentName = "S A"
}

sI = {
    \voiceOne
    \repeat unfold 4 f''2
    \separateSopranos
    \repeat unfold 4 g''2
    \convSopranos
    \repeat unfold 4 c''2
}

sII = {
    s1*2
    \voiceTwo
    \change Staff = "up"
    \repeat unfold 4 d''2
}

aI = {
    \voiceTwo
    \repeat unfold 4 a'2
    \voiceOne
    \repeat unfold 4 b'2
    \convUpStaffBarLine
    \voiceTwo
    \repeat unfold 4 g'2
}

aII = {
    s1*2
    \voiceTwo
    \repeat unfold 4 g'2
}

ten = {
    \voiceOne
    \repeat unfold 4 c'2
    \repeat unfold 4 d'2
    \repeat unfold 4 c'2
}

bas = {
    \voiceTwo
    \repeat unfold 4 f2
    \repeat unfold 4 g2
\repeat unfold 4 \c2
}
\score {
<<
\new ChoirStaff <<
\new Staff = up \with {
  instrumentName = "SI SII"
  shortInstrumentName = "SI SII"
}\{ s1*4 \}
\new Staff = shared \with {
  instrumentName = "S A"
  shortInstrumentName = "S A"
} <<
\new Voice = sopI \sI
\new Voice = sopII \sII
\new Voice = altI \aI
\new Voice = altII \aII
>>
\new Lyrics \with {
  alignBelowContext = up
}\lyricsto sopII { e f g h }
\new Lyrics \lyricsto altI { a b c d e f g h i j k l }
\new Staff = men \with {
  instrumentName = "T B"
  shortInstrumentName = "T B"
} <<
\clef F
\new Voice = ten \ten
\new Voice = bas \bas
>>
\new Lyrics \lyricsto bas { a b c d e f g h i j k l }
>>
<<
\layout {
\context {
  \Staff \RemoveEmptyStaves
  \override VerticalAxisGroup.remove-first = ##t
}
}
>}

Aggiungere citazioni orchestrali a una partitura vocale

L’esempio seguente mostra un approccio per semplificare l’aggiunta di citazioni orchestrali a una riduzione per pianoforte di una partitura vocale. La funzione musicale `\cueWhile` prende quattro argomenti: la musica da cui prendere la citazione, come è definita da `\addQuote`, il nome da inserire prima delle notine, poi o `\#UP` o `\#DOWN` per specificare o `\voiceOne` col nome sopra il rigo o `\voiceTwo` col nome sotto il rigo, e infine la musica per pianoforte che deve apparire in parallelo alle notine. Il nome dello strumento citato è posto a sinistra delle notine. Molti passaggi possono essere citati, ma non possono sovrapporsi l’uno l’altro nel tempo.

```plaintext
\cueWhile =
#(define-music-function
  (instrument name dir music)
  (string? string? ly:dir? ly:music?)
#
  \cueDuring $instrument #dir {
    \once \override TextScript.self-alignment-X = #RIGHT
    \once \override TextScript.direction = $dir
```
\markup { \tiny #name }

$music

#}

\flute = \relative c'' { 
  \transposition c'
  s4 s4 e g
}
\addQuote "flute" { \flute }

\clarinet = \relative c' { 
  \transposition bes
  fis4 d d c
}
\addQuote "clarinet" { \clarinet }

\singer = \relative c'' { c4. g8 g4 bes4 }

\words = \lyricmode { here's the lyr -- ics }

\pianoRH = \relative c'' { 
  \transposition c'
  \cueWhile "clarinet" "Clar." #DOWN { c4. g8 }
  \cueWhile "flute" "Flute" #UP { g4 bes4 }
}

\pianoLH = \relative c { c4 <c' e> e, <g c> }

\score { <<
  \new Staff { 
    \new Voice = "singer" { 
      \singer 
    }
  }
  \new Lyrics { 
    \lyricsto "singer"
    \words 
  }
  \new PianoStaff <<
    \new Staff { 
      \new Voice { 
        \pianoRH 
      }
    }
    \new Staff { 
      \clef "bass"
      \pianoLH 
    }
  >>
  >>
}
Adjusting lyrics vertical spacing

This snippet shows how to bring the lyrics line closer to the staff.

% Default layout:
<<
\new Staff \new Voice = melody \relative c' {
  c4 d e f
g4 f e d
c1
}
\new Lyrics \lyricsto melody { aa aa aa aa aa aa aa aa }

\new Staff {
  \new Voice = melody \relative c' {
    c4 d e f
g4 f e d
c1
  }
}
% Reducing the minimum space below the staff and above the lyrics:
\new Lyrics \with {
  \override VerticalAxisGroup.nonstaff-relatedstaff-spacing = #'((basic-distance . 1))
}
\lyricsto melody { aa aa aa aa aa aa aa aa }
>>

Aligning syllables with melisma

By default, lyrics syllables that start a melisma are left aligned on their note. The alignment can be altered using the lyricMelismaAlignment property.
\score {
\new Staff {  
  \relative c'  
  \new Voice = "vocal" {  
    c d- \markup default \ d e  
    c d- \markup "right aligned" \ d e  
    c d- \markup "center aligned" \ d e  
    c d- \markup "reset to default" \ d e  
  }  
}  
\new Lyrics \lyricsto "vocal" {  
  word word word  
  \set lyricMelismaAlignment = \#RIGHT  
  word word word  
  \set lyricMelismaAlignment = \#CENTER  
  word word word  
  \unset lyricMelismaAlignment  
  word word word  
}  
}  

\layout {  
  \context {  
    \Voice  
      \consists "Ambitus_engraver"  
  }  
}  

Ambitus

Ambitus indicate pitch ranges for voices. Accidentals only show up if they are not part of the key signature. AmbitusNoteHead grobs also have ledger lines.

\layout {  
  \context {  
    \Voice  
      \consists "Ambitus_engraver"  
  }  
}  

\new Staff {  
  \relative c' {  
    \time 2/4  
    c4 f'  
  }  
}  

\new Staff {  
  \relative c' {  
    \time 2/4  
    \key d \major  
    cis4 as'  
  }  
}
Ambitus dopo armatura di chiave

Per impostazione predefinita, gli ambitus sono posizionati a sinistra della chiave. La funzione \ambitusAfter permette di cambiare questo posizionamento. La sintassi è \ambitusAfter grob-interface (vedi Graphical Object Interfaces (http://lilypond.org/doc/v2.22/Documentation/internals/graphical-object-interfaces) per un elenco dei possibili valori per grob-interface.)

Un caso d’uso comune è il posizionamento dell’ambitus tra l’armatura di chiave e l’indicazione di tempo.

\new Staff \with {
  \consists Ambitus_engraver
} \relative {
  \ambitusAfter key-signature
  \key d \major
  es’8 g bes cis d2
}

Ambitus su più voci

Se si aggiunge l’incisore Ambitus_engraver al contesto Staff viene creato un solo ambitus per il rigo, anche nel caso di righi che hanno più voci.

\new Staff \with {
  \consists "Ambitus_engraver"
}

\new Voice \relative c' { \voiceOne c4 a d e f1 }
\new Voice \relative c' { \voiceTwo es4 f g as b1 }

}}
Modello per notazione antica – trascrizione moderna di musica gregoriana

Questo esempio mostra come realizzare una trascrizione moderna di musica gregoriana. La musica gregoriana non presenta la suddivisione in misure né gambi; impiega soltanto le teste della minima e della semiminima, e dei segni appositi che indicano pause di diversa lunghezza.
\include "gregorian.ly"

chant = \relative c' {
  \set Score.timing = ##f
  f4 a2 \divisioMinima
  g4 b2 a2 f2 \divisioMaior
  g4 f ( g ) a2 \finalis
}

verba = \lyricmode {
  Lo -- rem ip -- sum do -- lor sit a -- met
}

\score {
  \new GregorianTranscriptionStaff <<
    \new GregorianTranscriptionVoice = "melody" \chant
    \new GregorianTranscriptionLyrics = "one" \lyricsto melody \verba
  >>
}

Lorem ipsum dolor sit amet

Modello per salmo anglicano

Questo modello presenta un modo per impostare un salmo anglicano. Mostra anche come le strofe possano essere aggiunte come testo separato al di sotto della musica. Le due strofe sono scritte con stili diversi per illustrare le varie possibilità.
SopranoMusic = \relative g' {
  g1 | c2 b | a1 | \bar "||"
  a1 | d2 c | c b | c1 | \bar "||"
}

AltoMusic = \relative c' {
  e1 | g2 g | f1 |
  f1 | f2 e | d d | e1 |
}

TenorMusic = \relative a {
Vocal music

\begin{music}
c1 | c2 c | c1 |
d1 | g,2 g | g g | g1 |
\end{music}

\begin{music}
BassMusic = \relative c {
  c1 | e2 e | f1 |
d1 | b2 c | g' g | c,1 |
}\end{music}

\begin{music}
\time 2/2
\end{music}

\begin{music}
dot = \markup {
  \raise #0.7 \musicglyph "dots.dot"
}\end{music}

\begin{music}
tick = \markup {
  \raise #1 \fontsize #-5 \musicglyph "scripts.rvarcomma"
}\end{music}

% Use \markup to center the chant on the page
\begin{music}
\markup {
  \fill-line {
    \score { % centered
      \new ChoirStaff <<
        \new Staff <<
          \global
          \clef "treble"
          \new Voice = "Soprano" <<
            \voiceOne
            \SopranoMusic
          \>
          \new Voice = "Alto" <<
            \voiceTwo
            \AltoMusic
          \>
        \>
        \new Staff <<
          \clef "bass"
          \global
          \new Voice = "Tenor" <<
            \voiceOne
            \TenorMusic
          \>
          \new Voice = "Bass" <<
            \voiceTwo
            \BassMusic
          \>
        \>
    }
  }
}\end{music}
let us \textbf{sing} | unto \textbf{the} | Lord : let
\line
us heartily
\concat \{ re \textbf{joice} \}
in the | strength of | our
\line
sal | vation.
\null
\line
\hspace #2.5 8. Today if ye will hear his voice *
\line
\concat \{ \textbf{hard en} \}
\tick not your \tick hearts : as in the pro-
\line\vocation * and as in the \textbf{day of tempt-} \tick
\line \text{-ation} \tick in the \tick wilderness.
O come let us sing | unto the | Lord: let
us heartily rejoice in the | strength of | our
sal | vation.

8. Today if ye will hear his voice *
harden’not your’hearts: as in the pro-
vocation * and as in the day of tempt-
-ation ‘in the wilderness.

Arranging separate lyrics on a single line

Sometimes you may want to put lyrics for different performers on a single line: where there is
rapidly alternating text, for example. This snippet shows how this can be done with \override
VerticalAxisGroup.nonstaff-nonstaff-spacing.minimum-distance = ##f.

```latex
\layout {
  \context {
    \Lyrics
      \override VerticalAxisGroup.nonstaff-nonstaff-spacing.minimum-distance = ##f
  }
}

aliceSings = \markup { \smallCaps "Alice" }
eveSings = \markup { \smallCaps "Eve" }

<<
  \new Staff <<
    \new Voice = "alice" {
      f'4^\aliceSings g’ r2 |
      s1 |
      f'4^\aliceSings g’ r2 |
      s1 | \break
      % ...

    \voiceOne
    s2 a'8^\aliceSings a’ b’4 |
    \oneVoice
    g’1
    }
  \new Voice = "eve" {
    s1 |
```
Changing stanza fonts

Fonts can be changed independently for each stanza, including the font used for printing the stanza number.

%{You may have to install additional fonts.}

Red Hat Fedora

dejavu-fonts-all

Debian GNU/Linux, Ubuntu

fonts-dejavu-core
fonts-dejavu-extra
\relative c'\ {  
  \time 3/4  
g2 e4 \n  a2 f4 \n  g2.  
}  
\addlyrics {  
  \set stanza = "1. "  
  Hi, my name is Bert.  
}  
\addlyrics {  
  \override StanzaNumber.font-name = "DejaVu Sans"  
  \set stanza = "2. "  
  \override LyricText.font-family = "typewriter"  
  Oh, ché -- ri, je t'aime  
}  

\begin{music}  
  1. Hi, my name is Bert.  
  2. Oh, ché-ri, je t'aime  
\end{music}  

\section{Notazione per canti e salmi}  

Questa forma di notazione è utilizzata per i salmi, dove i versi non sono sempre della stessa lunghezza.

\begin{music}  
  \set stemOff = \hide Staff.Stem  
  \set stemOn = \undo \stemOff  
  \new Staff \with { \remove "Time_signature_engraver" }  
  {  
    \key g \minor  
    \cadenzaOn  
    \stemOff a'\breve bes'4 g'4  
    \stemOn a'2 \section  
    \stemOff a'\breve g'4 a'4  
    \stemOn f'2 \section  
    \stemOff a'\breve\markup { \italic flexe }  
    \stemOn g'2 \fine  
  }  
\end{music}
**Forcing hyphens to be shown**

If LilyPond does not think there is space for a hyphen, it will be omitted. The behaviour can be overridden with the `minimum-distance` property of `LyricHyphen`.

\relative c'' {
   c32 c c c
   c32 c c c
   c32 c c c
   c32 c c c
}

\addlyrics {
   syl -- lab word word
   \override LyricHyphen.minimum-distance = #1.0
   syl -- lab word word
   \override LyricHyphen.minimum-distance = #2.0
   syl -- lab word word
   \revert LyricHyphen.minimum-distance
   syl -- lab word word
}

```
\relative c'' {
   c32 c c c
   c32 c c c
   c32 c c c
   c32 c c c
}
```

**Formattazione delle sillabe del testo vocale**

La modalità markup può essere usata per formattare le singole sillabe del testo vocale.

```
\relative c'' {
   c4 c c c
}
\lyricmode {
   Lyrics \markup { \italic can } \markup { \with-color #red contain }
   \markup { \fontsize #8 \bold Markup! }
}
```

```
<<
   \new Voice = melody \mel
   \new Lyrics \lyricsto melody \lyr
>>
```

```
Lyrics can contain Markup!
```

**How to put ties between syllables in lyrics**

This can be achieved by separating those syllables by tildes.

```
\lyrics {
   wa~o~a
}
```
waqwa

Modello per inno

Il codice seguente presenta un modo di impostare un inno in cui ogni verso inizia e finisce con una misura parziale. Mostra anche come aggiungere delle strofe come testo separatamente sotto la musica.

```latex
Timeline = {
  \time 4/4
  \tempo 4=96
  \partial 2
  s2 | s1 | s2 \textit{breathe} s2 | s1 | s2 \textit{caesura} \textit{break}
  s2 | s1 | s2 \textit{breathe} s2 | s1 | s2 \textit{fine}
}

SopranoMusic = \relative g' { 
 g4 g | g g g g | g g g g | g g g g | g2 
 g4 g | g g g g | g g g g | g g g g | g2 
}

AltoMusic = \relative c' { 
 d4 d | d d d d | d d d d | d d d d | d2 
 d4 d | d d d d | d d d d | d d d d | d2 
}

TenorMusic = \relative a { 
 b4 b | b b b b | b b b b | b b b b | b2 
 b4 b | b b b b | b b b b | b b b b | b2 
}

BassMusic = \relative g { 
 g4 g | g g g g | g g g g | g g g g | g2 
 g4 g | g g g g | g g g g | g g g g | g2 
}

global = { 
 \key g \major 
}

\score { % Start score 
  << 
    \new PianoStaff << % Start pianostaff 
    \new Staff << % Start Staff = RH 
    \global 
    \clef "treble" 
    \new Voice = "Soprano" << % Start Voice = "Soprano" 
    \Timeline 
    \voiceOne 
    \SopranoMusic 
    >> % End Voice = "Soprano" 
    \new Voice = "Alto" << % Start Voice = "Alto" 
    \Timeline 
  }
```
\voiceTwo
\AltoMusic
>>  \% End Voice = "Alto"
>>  \% End Staff = RH
\new Staff <<  \% Start Staff = LH
\global
\clef "bass"
\new Voice = "Tenor" <<  \% Start Voice = "Tenor"
\Timeline
\voiceOne
\TenorMusic
>>  \% End Voice = "Tenor"
\new Voice = "Bass" <<  \% Start Voice = "Bass"
\Timeline
\voiceTwo
\BassMusic
>>  \% End Voice = "Bass"
>>  \% End Staff = LH
>>  \% End pianostaff
>
}  \% End score

\markup {
\fill-line {
    ""
    {
        \column {
            \left-align {
                "This is line one of the first verse"
                "This is line two of the same"
                "And here's line three of the first verse"
                "And the last line of the same"
            }
        }
    }""

\layout {
  \context {
    \Score 
    caesuraType = '#'((bar-line . "||"))
    fineBarType = "||"
  }
}

\paper {  \% Start paper block
    indent = 0  \% don't indent first system
    line-width = 130  \% shorten line length to suit music
}  \% End paper block
This is line one of the first verse
This is line two of the same
And here’s line three of the first verse
And the last line of the same

Allineamento del testo vocale

L’allineamento orizzontale del testo vocale si imposta attraverso la proprietà self-alignment-\(X\) dell’oggetto LyricText. \(-1\) è sinistra, \(0\) è centro e \(1\) è destra; si possono usare anche \#LEFT, \#CENTER e \#RIGHT.

\[
\text{\texttt{\\\layout{ ragged-right = ##f }}}
\]
\[
\text{\texttt{\relative { c'' {}}}}
\]
\[
\text{\texttt{\addlyrics {}}}
\]
\[
\text{\texttt{\once \override LyricText.self-alignment-\(X\) = #LEFT}}
\]
\[
\text{\texttt{"This is left-aligned"}}
\]
\[
\text{\texttt{\once \override LyricText.self-alignment-\(X\) = #CENTER}}
\]
\[
\text{\texttt{"This is centered"}}
\]
\[
\text{\texttt{\once \override LyricText.self-alignment-\(X\) = #1}}
\]
\[
\text{\texttt{"This is right-aligned"}}
\]

Marking notes of spoken parts with a cross on the stem
(Sprechstimme)

This example shows how to put crosses on stems. Mark the beginning of a spoken section with the `\speakOn` keyword, and end it with the `\speakOff` keyword.

\[
\text{\texttt{\speakOn = {}}}
\]
Ottenere la spaziatura del testo della vecchia versione 2.12

Il motore di spaziatura verticale è cambiato a partire dalla versione 2.14. Ciò può far sì che il testo vocale abbia un posizionamento diverso.

È possibile impostare delle proprietà dei contesti Lyric e Staff che facciano sì che il motore di spaziatura si comporti come nella versione 2.12.

global = {
  \key d \major
  \time 3/4
}

sopMusic = \relative c' { % VERSE ONE
  fis4 fis fis | \break
fis4. e8 e4
}

altoMusic = \relative c' {
% VERSE ONE
d4 d d |
d4. b8 b4 |
}

tenorMusic = \relative c' {
a4 a a |
b4. g8 g4 |
}

bassMusic = \relative c {
d4 d d |
g,4. g8 g4 |
}

words = \lyricmode {
Great is Thy faith -- ful -- ness,
}

\score {
\new ChoirStaff <<
\new Lyrics = sopranos
\new Staff = women <<
\new Voice = "sopranos" {
\voiceOne
\global \sopMusic
}
\new Voice = "altos" {
\voiceTwo
\global \altoMusic
}
>>
\new Lyrics = "altos"
\new Lyrics = "tenors"
\new Staff = men <<
clef bass
\new Voice = "tenors" {
\voiceOne
\global \tenorMusic
}
\new Voice = "basses" {
\voiceTwo \global \bassMusic
}
>>
\new Lyrics = basses
\context Lyrics = sopranos \lyricsto sopranos \words
\context Lyrics = altos \lyricsto altos \words
\context Lyrics = tenors \lyricsto tenors \words
\texttt{context Lyrics} = basses \texttt{lyricsto} basses \texttt{words}

\texttt{layout}\{
  \texttt{context}\{
    \texttt{Lyrics}\{
      \texttt{override VerticalAxisGroup.staff-affinity} = ##f
      \texttt{override VerticalAxisGroup.staff-staff-spacing} = 
      #'((basic-distance . 0)
      (minimum-distance . 2)
      (padding . 2))
    }\}
    \texttt{context}\{
      \texttt{Staff}\{
        \texttt{override VerticalAxisGroup.staff-staff-spacing} = 
        #'((basic-distance . 0)
        (minimum-distance . 2)
        (padding . 2))
      }\}
  }\}

Great is Thy

faith - - - ful - ness,
Modello per orchestra, coro e pianoforte

Questo modello mostra come usare i contesti annidati StaffGroup e GrandStaff per creare sottogruppi degli strumenti dello stesso tipo. Mostra anche come usare \transpose in modo che le variabili mantengano la musica per gli strumenti traspositori nell’intonazione reale.

#(set-global-staff-size 17)
\paper {
  \indent = 3.0\cm  % add space for instrumentName
  \short-indent = 1.5\cm  % add less space for shortInstrumentName
}

fluteMusic = \relative c' { \key g \major g'1 b }
% Pitches as written on a manuscript for Clarinet in A
% are transposed to concert pitch.

clarinetMusic = \transpose c' a
  \relative c'' { \key bes \major bes1 d }

trumpetMusic = \relative c { \key bes \major bes1 d }
% Key signature is often omitted for horns

hornMusic = \transpose c' f
  \relative c { d'1 fis }

percussionMusic = \relative c { \key g \major g1 b }

sopranoMusic = \relative c' { \key g \major g'1 b }
sopranoLyrics = \lyricmode { Lyr -- ics }
altoIMusic = \relative c' { \key g \major g'1 b }
altoIIImusic = \relative c{ \key g \major g'1 b }
altoIIIlyrics = \lyricmode { Ah -- ah }
tenorMusic = \relative c' { \clef "treble_8" \key g \major g1 b }
tenorLyrics = \lyricmode { sopranolys }
pianoRHMUSIC = \relative c{ \key g \major g'1 b }
pianoLHMUSIC = \relative c{ \clef bass \key g \major g1 b }
violinoIMUSIC = \relative c{ \key g \major g'1 b }
violinoIIIMUSIC = \relative c{ \key g \major g'1 b }
violaMusic = \relative c { \clef alto \key g \major g'1 b }

celloMusic = \relative c { \clef bass \key g \major g1 b }

bassMusic = \relative c { \clef "bass_8" \key g \major g,1 b }

\score {
  \new StaffGroup = "StaffGroup_woodwinds" <<
  \new Staff = "Staff_flute" \with { instrumentName = "Flute" }
    \fluteMusic

  \new Staff = "Staff_clarinet" \with {
    instrumentName = \markup { \concat { "Clarinet in B" \flat } }
  }

  % Declare that written Middle C in the music
  % to follow sounds a concert B flat, for
  % output using sounded pitches such as MIDI.
  % \transposition bes

  % Print music for a B-flat clarinet
  \transpose bes c' \clarinetMusic

  \new StaffGroup = "StaffGroup_brass" <<
  \new Staff = "Staff_hornI" \with { instrumentName = "Horn in F" }
  % \transposition f
  \transpose f c' \hornMusic

  \new Staff = "Staff_trumpet" \with { instrumentName = "Trumpet in C" }
    \trumpetMusic

  \new RhythmicStaff = "RhythmicStaff_percussion"
    \with { instrumentName = "Percussion" }
  \percussionMusic

  \new PianoStaff \with { instrumentName = "Piano" }
  \pianoRHMusic
  \pianoLHMusic

  \new ChoirStaff = "ChoirStaff_choir" <<
  \new Staff = "Staff_soprano" \with { instrumentName = "Soprano" }
    \new Voice = "soprano"
      \sopranoMusic

  \new Lyrics \lyricsto "soprano" { \sopranoLyrics }
  \new GrandStaff = "GrandStaff_altos"
    \with { \accepts Lyrics } <<
\new Staff = "Staff_altoI" \with { instrumentName = "Alto I" }
\new Voice = "altoI"
\altoIMusic
\new Lyrics \lyricsto "altoI" { \altoILyrics }
\new Staff = "Staff_altoII" \with { instrumentName = "Alto II" }
\new Voice = "altoII"
\altoIIMusic
\new Lyrics \lyricsto "altoII" { \altoIIlyrics }
>>
\new Staff = "Staff_tenor" \with { instrumentName = "Tenor" }
\new Voice = "tenor"
\tenorMusic
\new Lyrics \lyricsto "tenor" { \tenorLyrics }
>>
\new StaffGroup = "StaffGroup_strings" <<
\new GrandStaff = "GrandStaff_violins" <<
\new Staff = "Staff_violinI" \with { instrumentName = "Violin I" }
\violinIMusic
\new Staff = "Staff_violinII" \with { instrumentName = "Violin II" }
\violinIIMusic
>>
\new Staff = "Staff_viola" \with { instrumentName = "Viola" }
\violaMusic
\new Staff = "Staff_cello" \with { instrumentName = "Cello" }
\celloMusic
\new Staff = "Staff_bass" \with { instrumentName = "Double Bass" }
\bassMusic
>>
>>
\layout { }
Modello per pianoforte con melodia e testo

Ecco un tipico formato per canzoni: un rigo con linea melodica e testo, e sotto l'accompagnamento per pianoforte.

melody = \relative c' { 
\clef treble 
\key c \major 
\time 4/4 

a b c d 
} 

text = \lyricmode { 
Aaa Bee Cee Dee 
} 

upper = \relative c' { 

Putting lyrics inside the staff

Lyrics can be moved vertically to place them inside the staff. The lyrics are moved with \override LyricText.extra-offset = #'(0 . dy) and there are similar commands to move the extenders and hyphens. The offset needed is established with trial and error.

<<
\new Staff <<
\new Voice = "voc" \relative c' { \stemDown a bes c8 b c4 }
>>
\new Lyrics \with {
\override LyricText.extra-offset = #'(0 . 8.6)
Vocal music

\override LyricExtender.extra-offset = #'(0 . 8.6) \\
\override LyricHyphen.extra-offset = #'(0 . 8.6) \\
} \lyricsto "voc" \{ La la -- la __ la \} \\
}

\Staff \\
\noteheads.s2/accidentals.natural/\noteheads.s2/\timesig.C44/\clefs.G/\noteheads.s2 \\
\noteheads.s2/\accidentals.flat/\noteheads.s2 \\

Modello per coro SATB - quattro righi

Modello per coro SATB (quattro righi)

global = { \\
'\key c \major \\
'\time 4/4 \\
'\dynamicUp \\
} \\
sopranonotes = \relative c' { \\
\c2 \p < d c d \f \\
} \\
sopranowords = \lyricmode \{ do do do do \} \\
altonotes = \relative c' { \\
\c2\p d c d \\
} \\
altonwords = \lyricmode \{ re re re re \} \\
tenornotes = { \\
'\clef "G_8" \\
\c2\mp d c d \\
} \\
tenorwords = \lyricmode \{ mi mi mi mi \} \\
bassnotes = { \\
'\clef bass \\
\c2\mf d c d \\
} \\
basswords = \lyricmode \{ mi mi mi mi \}

\score { \\
'\new ChoirStaff << \\
'\new Staff << \\
'\new Voice = "soprano" << \\
'\global \\
'\sopranonotes \\
} \lyricsto "soprano" \sopranowords \\
} \\
\new Staff << \\
\new Voice = "alto" << \\
'\global \\
'\altonotes \\
}
Modello di rigo singolo con note, testo e accordi

Ecco il modello di un comune spartito semplificato (lead sheet): include linea melodica, testo vocale e sigle degli accordi.

```
melody = \relative c' { \\
clef treble \\
key c \major \\
time 4/4 \\

a4 b c d }
```

```
text = \lyricmode { \\
Aaa Bee Cee Dee }
```
Vocal music

}`

harmonies = \chordmode {
  a2 c
}

\score {
  <<
    \new ChordNames {
      \set chordChanges = ##t
      \harmonies
    }
    \new Voice = "one" { \autoBeamOff \melody }
    \new Lyrics \lyricsto "one" \text
  >>
  \layout { }
  \midi { }
}

Single staff template with notes, lyrics, chords and frets

Here is a simple lead sheet template with melody, lyrics, chords and fret diagrams.

verseI = \lyricmode {
  \set stanza = "1."
  This is the first verse
}

verseII = \lyricmode {
  \set stanza = "2."
  This is the second verse.
}

theChords = \chordmode {
  % insert chords for chordnames and fretboards here
  c2 g4 c
}

staffMelody = \relative c' {
  \key c \major
  \clef treble
  % Type notes for melody here
  c4 d8 e f4 g
  \bar "|."
}

\score {
  <<
Vocal music

\context ChordNames { \theChords }
\context FretBoards { \theChords }
\new Staff { 
  \context Voice = "voiceMelody" { \staffMelody }
}
\new Lyrics = "lyricsI" { 
  \lyricsto "voiceMelody" \verseI
}
\new Lyrics = "lyricsII" { 
  \lyricsto "voiceMelody" \verseII
}

1. This is the first verse
2. This is the second verse.

**Modello di rigo singolo con note e testo**

Questo piccolo modello presenta una semplice linea melodica con un testo. Copialo e incollalo, aggiungi le note e le parole. Questo esempio disabilita la disposizione automatica delle travature, come è consuetudine per le parti vocali. Per usare la disposizione automatica delle travature, cambia o commenta la relativa linea di codice.

```latex
melody = \relative c' { 
  \clef treble
  \key c \major
  \time 4/4

  a4 b c d
}
```

```latex
text = \lyricmode { 
  Aaa Bee Cee Dee
}
```

```latex
\score{ 
  \new Voice = "one" { 
    \autoBeamOff
    \melody
  }
  \new Lyrics \lyricsto "one" \text
}
```

This page contains a code snippet for creating vocal music with chord names and lyrics. The code defines contexts for chord names and fretboards, a staff with a voice melody, and lyrics for different verses. The score includes a melody line with a clef, key signature, and time signature, as well as lyrics in Italian. The page also includes a graphic of a musical notation with chords C, G, and C, indicating the first and second verses. The text describes how to use this model to create a simple melodic line with text, and how to enable automatic bar line placement if desired.
Skips in lyric mode

The syntax for skips is only available in note mode and chord mode. In other situations, for example, when entering lyrics, using the \skip command is recommended.

```
<< \relative c'' { a1 | a } \new Lyrics \lyricmode { \skip 1 bla1 } >>
```

Skips in lyric mode (2)

Although skips cannot be used in \lyricmode (it is taken to be a literal “s”, not a space), double quotes (""), or underscores (_) are available.

So for example:

```
<< \relative c'' { a4 b c d } \new Lyrics \lyricmode { a4 "" _ gap } >>
```

Usare arpeggioBracket per rendere i divisi più visibili

Si può usare arpeggioBracket per indicare la divisione delle voci quando non ci sono gambi che forniscono questa informazione. Questo caso è frequente nella musica corale.

```
\include "english.ly"

\score {
  \relative c'' {
    \key a \major
    \time 2/2
    << \new Voice = "upper"
    << { \voiceOne \arpeggioBracket a2( b2
```
Usare le etichette per produrre musica mensurale e moderna
dallo stesso sorgente

Usando le etichette (tag), è possibile usare la stessa musica per produrre sia la musica mensurale che quella moderna. In questo frammento, viene introdotta la funzione `menrest`, che permette alle pause mensurali di essere posizionate precisamente sul rigo come nell’originale, ma con le pause moderne nella posizione standard. Le etichette vengono usate per produrre diversi tipi di stanghetta alla fine della musica, ma possono essere usate anche quando sono necessarie altre differenze: per esempio se si vogliono usare “pause d’intero” (R1, R\breve, etc.) nella musica moderna, ma pause normali (r1, r\breve, etc.) nella versione mensurale. La conversione di musica mensurale nel suo equivalente moderno viene solitamente chiamata trascrizione.

\begin{verbatim}
menrest = #(define-music-function (note)
  (ly:music?)
  #(\tag #'mens $(make-music 'RestEvent note)
    \tag #'mod $(make-music 'RestEvent note 'pitch '())
  )

MenStyle = {
  \autoBeamOff
  \override NoteHead.style = #'petrucci
  \override Score.BarNumber.transparent = ##t
  \override Stem.neutral-direction = #up
}

finalis = \section

Music = \relative c' {
  \set Score=tempoHideNote = ##t
}
\end{verbatim}
\key f \major
\time 4/4
g1 d’2 \menrest bes4 bes2 a2 r4 g4 fis2.
\finalis
}

MenLyr = \lyricmode { So farre, deere life, deare life }
ModLyr = \lyricmode { So far, dear life, dear life }

\score {
\keepWithTag #'mens {
  <<
    \new MensuralStaff
    {
      \new MensuralVoice = Cantus
      \clef "mensural-c1" \MenStyle \Music
    }
    \new Lyrics \lyricsto Cantus \MenLyr
  >>
}
}

\score {
\keepWithTag #'mod {
  \new ChoirStaff <<
    \new Staff
    {
      \new Voice = Sop \with {
        \remove "Note_heads_engraver"
        \consists "Completion_heads_engraver"
        \remove "Rest_engraver"
        \consists "Completion_rest_engraver" }
      \shiftDurations #1 #0 { \autoBeamOff \Music }
    }
    \new Lyrics \lyricsto Sop \ModLyr
  >>
}
}
Allineare verticalmente gli ossia e il testo vocale

Questo frammento mostra come usare le proprietà di contesto `alignBelowContext` e `alignAboveContext` per controllare il posizionamento del testo vocale e degli ossia.

\begin{verbatim}
\paper {
  \ragged-right = #t
}

\relative c' <<
  \new Staff = "1" { c4 c s2 }
  \new Staff = "2" { c4 c s2 }
  \new Staff = "3" { c4 c s2 }
  { \skip 2
    \lyrics {
      \set alignBelowContext = #'1'
      \lyrics4 below
    }
    \new Staff \with {
      alignAboveContext = #'3'
      fontSize = #-2
      \override StaffSymbol.staff-space = #(magstep -2)
      \remove "Time_signature_engraver"
    } {
      \tuplet 6/4 {
        \override TextScript.padding = #3
        c8[^"ossia above" d e d e f]
      }
    }
  }
\end{verbatim}

Vertically centered common lyrics

In a vocal piece where there are several (two, four or more) lines of lyrics and common lyrics for all voices at some point, the common lyrics may be made to appear vertically centered, as shown in the following example:
dropLyrics = {
    \override LyricText.extra-offset = #'(0 . -4.5)
    \override LyricHyphen.extra-offset = #'(0 . -4.5)
    \override LyricExtender.extra-offset = #'(0 . -4.5)
    \override StanzaNumber.extra-offset = #'(0 . -4.5)
}
raiseLyrics = {
    \revert LyricText.extra-offset
    \revert LyricHyphen.extra-offset
    \revert LyricExtender.extra-offset
    \revert StanzaNumber.extra-offset
}
skipFour = \repeat unfold 4 { \skip 8 }
lyricsA = \lyricmode {
    The first verse has
    \dropLyrics
    \set stanza = #" All:" the com -- mon __ words
    \raiseLyrics
    used in all four.
}
lyricsB = \lyricmode { In stan -- za two, \skipFour al -- so ap -- pear. }
lyricsC = \lyricmode { By the third verse, \skipFour are get -- ting dull. }
lyricsD = \lyricmode { Last stan -- za, and \skipFour get used once more. }

melody = \relative c' {
    c4 d e f |
    g f e8( e f) d |
    c4 e d c |
}

\score {
    <<
    \new Voice = m \melody
    \new Lyrics \lyricsto m \lyricsA
    \new Lyrics \lyricsto m \lyricsB
    \new Lyrics \lyricsto m \lyricsC
    \new Lyrics \lyricsto m \lyricsD
    >>
}
Modello per complesso vocale

Ecco una tipica partitura corale a quattro parti, SATB. Se il complesso è più ampio, è spesso comodo scrivere gli elementi comuni in un’unica sezione, che verrà poi inclusa in tutte le parti. Ad esempio, l’indicazione di tempo e l’armatura di chiave sono quasi sempre le stesse per tutte le parti. Come nel modello dell’“Inno”, le quattro voci sono ripartite in due soli righi.

\paper {
  \top-system-spacing.basic-distance = #10
  \score-system-spacing.basic-distance = #20
  \system-system-spacing.basic-distance = #20
  \last-bottom-spacing.basic-distance = #10
}

\global = {
  \key c \major
  \time 4/4
}

sopMusic = \relative {
  c''4 c c8[( b)] c4
}
sopWords = \lyricmode {
  hi hi hi hi
}

altoMusic = \relative {
  e'4 f d e
}
altoWords = \lyricmode {
  ha ha ha ha
}

tenorMusic = \relative {
  g4 a f g
}
tenorWords = \lyricmode {
  hu hu hu hu
}

bassMusic = \relative {
  c4 c g c
}
bassWords = \lyricmode {

The first verse has used in all four.
In stanza two, All: the common words also appear.
By the third verse, All: the common words are getting dull.
Last stanza, and the four parts are used once more.
ho ho ho ho

\score {
    \new ChoirStaff <<
    \new LyricStems = "sopranos" \with {
        \override VerticalAxisGroup Staff-affinity = \down
    }
    \new Staff = "women" <<
    \new Voice = "sopranos" {
        \voiceOne << \global \sopMusic >>
    }
    \new Voice = "altos" {
        \voiceTwo << \global \altoMusic >>
    }
    >>
    \new LyricStems = "altos"
    \new LyricStems = "tenors" \with {
        \override VerticalAxisGroup Staff-affinity = \down
    }
    \new Staff = "men" <<
    \clef bass
    \new Voice = "tenors" {
        \voiceOne << \global \tenorMusic >>
    }
    \new Voice = "basses" {
        \voiceTwo << \global \bassMusic >>
    }
    >>
    \new LyricStems = "basses"
    \context LyricStems = "sopranos" \lyricsto "sopranos" \sopWords
    \context LyricStems = "altos" \lyricsto "altos" \altoWords
    \context LyricStems = "tenors" \lyricsto "tenors" \tenorWords
    \context LyricStems = "basses" \lyricsto "basses" \bassWords
}
Modello per gruppo vocale con riduzione per pianoforte automatica

Questo modello aggiunge una riduzione automatica per pianoforte alla tipica partitura vocale SATB illustrata in “Modello per complesso vocale”. Si dimostra così uno dei punti di forza di LilyPond – è possibile usare una definizione musicale più di una volta. Qualsiasi modifica venga fatta alle note delle voci (ad esempio, tenorMusic) verrà applicata anche alla riduzione per pianoforte.

```latex
\global{%
  \key c \major
  \time 4/4
}

sopMusic = \relative {
  c''4 c c8[(b)] c4
}
sopWords = \lyricmode {
  hi hi hi hi
}

altoMusic = \relative {
  e'4 f d e
}
altoWords =\lyricmode {
  ha ha ha ha
}

tenorMusic = \relative {
  g4 a f g
}
tenorWords = \lyricmode {
  hu hu hu hu
}```
bassMusic = \relative { 
c4 c g c 
}
bassWords = \lyricmode { 
ho ho ho ho 
}

\score { 
<< 
  \new ChoirStaff << 
  \new Lyrics = "sopranos" \with { 
    % This is needed for lyrics above a staff 
    \override VerticalAxisGroup.staff-affinity = #DOWN 
  } 
  \new Staff = "women" << 
  \new Voice = "sopranos" { \voiceOne << \global \sopMusic >> } 
  \new Voice = "altos" { \voiceTwo << \global \altoMusic >> } 
  >> 
  \new Lyrics = "altos" 
  \new Lyrics = "tenors" \with { 
    % This is needed for lyrics above a staff 
    \override VerticalAxisGroup.staff-affinity = #DOWN 
  } 
  \new Staff = "men" << 
  \clef bass 
  \new Voice = "tenors" { \voiceOne << \global \tenorMusic >> } 
  \new Voice = "basses" { \voiceTwo << \global \bassMusic >> } 
  >> 
  \new Lyrics = "basses" 
  \context Lyrics = "sopranos" \lyricsto "sopranos" \sopWords 
  \context Lyrics = "altos" \lyricsto "altos" \altoWords 
  \context Lyrics = "tenors" \lyricsto "tenors" \tenorWords 
  \context Lyrics = "basses" \lyricsto "basses" \bassWords 
  >> 
  \new PianoStaff << 
  \new Staff << 
    \set Staff.printPartCombineTexts = ##f 
    partCombine 
    << \global \sopMusic >> 
    << \global \altoMusic >> 
    >> 
  \new Staff << 
    \clef bass 
    \set Staff.printPartCombineTexts = ##f 
    partCombine 
    << \global \tenorMusic >> 
    << \global \bassMusic >> 
    >> 
  >>
Modello per gruppo vocale con testo allineato sotto e sopra i righi

Questo modello è fondamentalmente analogo al semplice modello “Complesso vocale”, con l’unica differenza che qui tutti i versi del testo sono posizionati usando `alignAboveContext` e `alignBelowContext`.

```
\global = {
  \key c \major
  \time 4/4
}

\sopMusic = \relative c' { c4 c c8[\(b\)] c4 }
\sopWords = \lyricmode { hi hi hi hi }

\altoMusic = \relative c' { e4 f d e }
\altoWords = \lyricmode { ha ha ha ha }

\tenorMusic = \relative c' { g4 a f g }
\tenorWords = \lyricmode { hu hu hu hu }
```
\score {
  \new ChoirStaff <<
    \new Staff = "women" <<
      \new Voice = "sopranos" { \voiceOne << \global \sopMusic >> }
      \new Voice = "altos" { \voiceTwo << \global \altoMusic >> }
    >>
    \new Lyrics \with { alignAboveContext = "women" } \lyricsto "sopranos" \sopWords
    \new Lyrics \with { alignBelowContext = "women" } \lyricsto "altos" \altoWords
    \%
    \%
    \%
    \%
}

\score {
  \new Staff = "men" <<
    \clef bass
    \new Voice = "tenors" { \voiceOne << \global \tenorMusic >> }
    \new Voice = "basses" { \voiceTwo << \global \bassMusic >> }
    >>
    \new Lyrics \with { alignAboveContext = "men" } \lyricsto "tenors" \tenorWords
    \new Lyrics \with { alignBelowContext = "men" } \lyricsto "basses" \bassWords
    \%
    \%
    \%
    
    
    
    
}
Modello per gruppo vocale con strofa e ritornello

Questo modello crea una partitura che inizia con una sezione solistica e prosegue in un ritornello a due voci. Illustra anche l’uso delle pause spaziatrici all’interno della variabile `global` per definire i cambi di tempo (e altri elementi comuni a tutte le parti) nel corso di tutta la partitura.

```latex
\global = {
    \key g \major
    \time 3/4
    s2.*2
    \break

    % refrain
    \time 2/4
    s2*2
    \bar "|."
}

SoloNotes = \relative g' {\clef "treble"

    % verse
    g4 g g |
    b4 b b |

    % refrain
    R2*2 |
}

SoloLyrics = \lyricmode {
    One two three |
    four five six |
}

SopranoNotes = \relative c'' {\clef "treble"

    % verse
    R2.*2 |

    % refrain
    c4 c |
    g4 g |
}

SopranoLyrics = \lyricmode {
    la la |
    la la |
}

BassNotes = \relative c {
\cleft "bass"

% verse
R2.*2 |

% refrain
c4 e |
d4 d |
}

BassLyrics = \lyricmode {
  dum dum |
  dum dum |
}

\score {
  <<
    \new Voice = "SoloVoice" << \global \SoloNotes >>
    \new Lyrics \lyricsto "SoloVoice" \SoloLyrics

    \new ChoirStaff <<
      \new Voice = "SopranoVoice" << \global \SopranoNotes >>
      \new Lyrics \lyricsto "SopranoVoice" \SopranoLyrics

      \new Voice = "BassVoice" << \global \BassNotes >>
      \new Lyrics \lyricsto "BassVoice" \BassLyrics
    >>
  >>

  \layout {
    ragged-right = ##t
    \context { \Staff
      % these lines prevent empty staves from being printed
      \RemoveEmptyStaves
      \override VerticalAxisGroup.remove-first = ##t
    }
  }
}

One two three four five six

la la la la

\text{dum dum dum dum}
Chords

Sezione “Chord notation” in Guida alla Notazione

Adding a figured bass above or below the notes

When writing a figured bass, you can place the figures above or below the bass notes, by defining the BassFigureAlignmentPositioning.direction property (exclusively in a Staff context). Choices are #UP (or #1), #CENTER (or #0) and #DOWN (or #-1).

This property can be changed as many times as you wish. Use \once \override if you don’t want the override to apply to the whole score.

```latex
bass = {
  \clef bass
  g4 b, c d
e  d8 c d2
}
```

```latex
continuo = \figuremode {
  <,>4 <6>4 <5/>4
  \override Staff.BassFigureAlignmentPositioning.direction = #UP
  \% bassFigureStaffAlignmentUp
  < _,+ >4 <6>
  \set Staff.useBassFigureExtenders = ##t
  \override Staff.BassFigureAlignmentPositioning.direction = #DOWN
  \% bassFigureStaffAlignmentDown
  <4>4. <4>8 < _,+ >4
}
```

```latex
\score { << \new Staff = bassStaff \bass
  \context Staff = bassStaff \continuo
  >> }
```

Aggiungere stanghette al contesto ChordNames

Per mostrare le stanghette nel contesto ChordNames, aggiungere l’incisore Bar_engraver.

```latex
\new ChordNames \with {
  \override BarLine.bar-extent = #'(-2 . 2)
  \consists "Bar_engraver"
}
```

```latex
\chordmode {
  f1:maj7 f:7 bes:7
}
```
F\(^{\#1}\) | F\(^7\) | B\(^{\flat}7\)

**Adjusting figured bass alteration glyphs**

In figured bass, specially designed glyphs for 6\(^{\flat}\), 7\(^{\flat}\), and 9\(^{\flat}\) are used by default. Similarly, specially designed glyphs for symbols 2\(+\), 4\(+\), and 5\(+\) are used by default if plus signs appear after the number.

To change that, pass an alist to `figuredBassPlusStrokedAlist` and set the glyph in question to \#f (or omit it).

\`\figures\`
\`\set figuredBassPlusDirection = \#RIGHT\`
\`<6\(\flat\)> <7\(\flat\)> <9\(\flat\)> r\`
\`<2\(+\)> <4\(+\)> <5\(+\)> r\`

\`\set figuredBassPlusStrokedAlist = \#'(2 . "figbass.twoplus")\`
\`;;; (4 . "figbass.fourplus")\`
\`;;; (5 . "figbass.fiveplus")\`
\`(6 . "figbass.sixstrokedit")\`
\`;;; (7 . "figbass.sevenstrokedit")\`
\`;;; (9 . "figbass.ninestrokedit")\`
\`<6\(\flat\)> <7\(\flat\)> <9\(\flat\)> r\`
\`<2\(+\)> <4\(+\)> <5\(+\)> r\`

\`6 7 9 2 4 5 6 7 9 2 4 5\`

**Modifica del separatore dell’accordo**

L’elemento che separa le diverse parti di un accordo può essere impostato su qualsiasi testo di tipo markup.

\`\chords\`
\`c:7sus4\`
\`\set chordNameSeparator = \markup \{ \typewriter | }\`
\`c:7sus4\`

\`C\(^7\)sus4\  C\(^7\)sus4\`

**Changing the chord names to German or semi-German notation**

The english naming of chords (default) can be changed to german (\germanChords replaces B and Bes with H and B) or semi-german (\semiGermanChords replaces B and Bes with H and Bb).

\`scm = \chordmode \{
\`c1/c | cis/cis\`
\`b1/b | bis/bis | bes/bes\`
Cambiare la posizione delle alterazioni del basso continuo

Le alterazioni e i segni più possono apparire prima o dopo i numeri, a seconda delle proprietà figuredBassAlterationDirection e figuredBassPlusDirection.
Eccezioni dei nomi degli accordi

La proprietà chordNameExceptions permette di salvare un elenco di notazioni speciali per accordi specifici.

% modify maj9 and 6(add9)
% Exception music is chords with markups
chExceptionMusic = {
    <c e g b d'>1-\markup { \super "maj9" }
    <c e g a d'>1-\markup { \super "6(add9)" }
}

% Convert music to list and prepend to existing exceptions.
chExceptions = #(append
    (sequential-music-to-chord-exceptions chExceptionMusic #t)
    ignatzekExceptions)

theMusic = \chordmode {
    g1:maj9 g1:6.9
    \set chordNameExceptions = #chExceptions
    g1:maj9 g1:6.9
}

\layout {
    ragged-right = ##t
}

<<
\new ChordNames \theMusic
\new Voice \theMusic
>>

Nome dell’accordo di settima maggiore

La formattazione dell’accordo di settima maggiore può essere regolata con la proprietà majorSevenSymbol.
Chord names alternative

Chord names are generated from a list of pitches. The functions which construct these names can be customised.

Here are shown chords following Ignatzek (pp. 17-18, 1995), used by default since LilyPond 1.7.20, compared with an alternative Jazz chord notation and Harald Banter’s (1987) notation. A smaller font is used in the latter case, as these tend to be overly verbose.

This mirrors the mechanism originally used in early LilyPond versions (pre-1.7); not having been properly maintained, however, some features have been lost (mainly chord exception lists) and bugs have been introduced.

%%% Legacy chord naming functions (formerly in scm/chord-generic-names.scm)
%%% Copyright (C) 2003--2015 Jan Nieuwenhuizen <janneke@gnu.org>

#(set-global-staff-size 19.7)

#(define-public (banter-chordnames pitches bass inversion context)
  (old_chord->markup 'banter pitches bass inversion context))

#(define-public (jazz-chordnames pitches bass inversion context)
  (old_chord->markup 'jazz pitches bass inversion context))

#(define (define-translator-property symbol type? description)
  (if (not (and (symbol? symbol)
                 (procedure? type?)
                 (string? description)))
    (ly:error "error in call of define-translator-property")
  (if (not (equal? (object-property symbol 'translation-doc) '#f))
    (ly:error (G_ "symbol ~S redefined") symbol))
  (set-object-property! symbol 'translation-type? type?)
  (set-object-property! symbol 'translation-doc description) symbol)

#(for-each
  (lambda (x)
    (apply define-translator-property x))
  '((chordNameExceptionsFull ,list? "An alist of full chord exceptions. Contains @code{(@var{chord} . @var{markup})} entries.
    (chordNameExceptionsPartial ,list? "An alist of partial chord exceptions. Contains @code{(@var{chord} . @var{prefix-markup} @var{suffix-markup})} entries.
    )))

#(define-public (old_chord->markup
"Entry point for @code{Chord_name_engraver}.
@var{pitches}, @var{bass}, and @var{inversion} are lily pitches."

(define (default-note-namer pitch)
  (note-name->markup pitch #f))

(define (markup-or-empty-markup markup)
  "Return MARKUP if markup, else empty-markup"
  (if (markup? markup) markup empty-markup))

(define (accidental->markup alteration)
  "Return accidental markup for ALTERATION."
  (if (= alteration 0)
      (make-line-markup (list empty-markup))
      (conditional-kern-before
       (alteration->text-accidental-markup alteration)
       (= alteration FLAT) 0.094725)))

(define (list-minus a b)
  "Return list of elements in A that are not in B."
  (lset-difference eq? a b))

(define (markup-join markups sep)
  "Return line-markup of MARKUPS, joining them with markup SEP"
  (if (pair? markups)
      (make-line-markup (list-insert-separator markups sep))
      empty-markup))

(define (conditional-kern-before markup bool amount)
  "Add AMOUNT of space before MARKUP if BOOL is true."
  (if bool
      (make-line-markup
       (list (make-hspace-markup amount)
             markup))
      markup))

(define (step-nr pitch)
  (let* ((pitch-nr (+ (* 7 (ly:pitch-octave pitch))
                        (ly:pitch-notename pitch)))
         (root-nr (+ (* 7 (ly:pitch-octave (car pitches)))
                     (ly:pitch-notename (car pitches))))
         (+ 1 (- pitch-nr root-nr))))

(define (next-third pitch)
  (ly:pitch-transpose pitch
   (ly:make-pitch 0 2 (if (or (= (step-nr pitch) 3)
                          (= (step-nr pitch) 5))
                          FLAT 0))))

(define (step-alteration pitch)
  (let* ((diff (ly:pitch-diff (ly:make-pitch 0 0 0) (car pitches)))
         (normalized-pitch (ly:pitch-transpose pitch diff)))

  (ly:pitch-transpose pitch
   (ly:make-pitch 0 2 (if (or (= (step-nr pitch) 3)
                          (= (step-nr pitch) 5))
                          FLAT 0))))

  (ly:pitch-transpose pitch
   (ly:make-pitch 0 2 (if (or (= (step-nr pitch) 3)
                          (= (step-nr pitch) 5))
                          FLAT 0))))

  (ly:pitch-transpose pitch
   (ly:make-pitch 0 2 (if (or (= (step-nr pitch) 3)
                          (= (step-nr pitch) 5))
                          FLAT 0))))

  (ly:pitch-transpose pitch
   (ly:make-pitch 0 2 (if (or (= (step-nr pitch) 3)
                          (= (step-nr pitch) 5))
                          FLAT 0))))

  (ly:pitch-transpose pitch
   (ly:make-pitch 0 2 (if (or (= (step-nr pitch) 3)
                          (= (step-nr pitch) 5))
                          FLAT 0))))
(alteration (ly:pitch-alteration normalized-pitch))
(if (= (step-nr pitch) 7) (+ alteration SEMI-TONE) alteration)))

(define (pitch-unalter pitch)
  (let ((alteration (step-alteration pitch)))
    (if (= alteration 0)
      pitch
      (ly:make-pitch (ly:pitch-octave pitch) (ly:pitch-notename pitch)
      (- (ly:pitch-alteration pitch) alteration))))))

(define (step-even-or-altered? pitch)
  (let ((nr (step-nr pitch)))
    (if (!= (modulo nr 2) 0)
      (!= (step-alteration pitch) 0)
      #t)))

(define (step->markup-plusminus pitch)
  (let ((alt (step-alteration pitch)))
    (make-line-markup
      (list
        (number->string (step-nr pitch))
        (cond
          ((= alt DOUBLE-FLAT) "--")
          ((= alt FLAT) "-")
          ((= alt NATURAL) ")")
          ((= alt SHARP) "+")
          ((= alt DOUBLE-SHARP) "++"))))))

(define (step->markup-accidental pitch)
  (make-line-markup
    (list (accidental->markup (step-alteration pitch))
    (make-simple-markup (number->string (step-nr pitch))))))

(define (step->markup-ignatzek pitch)
  (make-line-markup
    (if (and (= (step-nr pitch) 7)
      (= (step-alteration pitch) 1))
      (list (ly:context-property context 'majorSevenSymbol))
      (list (accidental->markup (step-alteration pitch))
        (make-simple-markup (number->string (step-nr pitch)))))))))

;; tja, kennok
(define (make-sub->markup step->markup)
  (lambda (pitch)
    (make-line-markup (list (make-simple-markup "no")
      (step->markup pitch)))))

(define (step-based-sub->markup step->markup pitch)
  (make-line-markup (list (make-simple-markup "no") (step->markup pitch)))))

(define (get-full-list pitch)
  (if (<= (step-nr pitch) (step-nr (last pitches)))
(cons pitch (get-full-list (next-third pitch)))
'(())

(define (get-consecutive nr pitches)
  (if (pair? pitches)
      (let* ((pitch-nr (step-nr (car pitches)))
             (next-nr (if (not (modulo pitch-nr 2) 0) (+ pitch-nr 2) nr)))
        (if (<= pitch-nr nr)
            (cons (car pitches) (get-consecutive next-nr (cdr pitches)))
                 '(()))
       '(())))

;;; FIXME -- exceptions no longer work. -vv

(define (full-match exceptions)
  (if (pair? exceptions)
      (let* ((e (car exceptions))
             (e-pitches (car e)))
        (if (equal? e-pitches pitches)
            e
            (full-match (cdr exceptions))))
#ifndef))

(define (partial-match exceptions)
  (if (pair? exceptions)
      (let* ((e (car exceptions))
             (e-pitches (car e)))
        (if (equal? e-pitches (take pitches (length e-pitches)))
            e
            (partial-match (cdr exceptions))))
#ifndef))

;;; FIXME: exceptions don't work anyway.
(if #f (begin
  (write-me "pitches: " pitches)))
(let* ((ly:context-property context 'chordNameExceptionsFull))
  (full-exceptions (full-match full-exceptions))
  (full-markup (if full-exception (cadr full-exception) '(())))
  (partial-exceptions (ly:context-property context 'chordNameExceptionsPartial))
  (partial-markup-prefix (if partial-exception (markup-or-empty-markup
                                               (cadr partial-exception))
                           empty-markup))
  (partial-markup-suffix (if (and partial-exception (pair? (caddr partial-exception)))
                           (markup-or-empty-markup (caddr partial-exception))
                           empty-markup))
  (root (car pitches))
  (full (get-full-list root))
;;; kludge alert: replace partial matched lower part of all with
;; 'normal' pitches from full
;; (all pitches)
(all (append (take full (length partial-pitches))
  (drop pitches (length partial-pitches))))

(highest (last all))
(missing (list-minus full (map pitch-unalter all)))
(consecutive (get-consecutive 1 all))
(rest (list-minus all consecutive))
(altered (filter step-even-or-altered? all))
(cons-alt (filter step-even-or-altered? consecutive))
(base (list-minus consecutive altered))

(if #f (begin
  (write-me "full:" full)
  ;; (write-me "partial-pitches:" partial-pitches)
  (write-me "full-markup:" full-markup)
  (write-me "partial-markup-prefix:" partial-markup-prefix)
  (write-me "partial-markup-suffix:" partial-markup-suffix)
  (write-me "all:" all)
  (write-me "altered:" altered)
  (write-me "missing:" missing)
  (write-me "consecutive:" consecutive)
  (write-me "rest:" rest)
  (write-me "base:" base)))

(case style
  ((banter)
   ;; root
   ;; + steps:altered + (highest all -- if not altered)
   ;; + subs:missing
   (let* ((root->markup default-note-namer)
          (step->markup step->markup-plusminus)
          (sub->markup (lambda (x)
                         (step-based-sub->markup step->markup x)))
          (sep (make-simple-markup "/")))
    (if
     (pair? full-markup)
     (make-line-markup (list (root->markup root) full-markup))
     (make-line-markup
      (list
       (root->markup root)
       partial-markup-prefix
       (make-super-markup
        (markup-join
         (append
          (map step->markup
               (append altered)))))))
(if (and (> (step-nr highest) 5) (not (step-even-or-altered? highest)))
    (list highest) '()))
(map sub->markup missing) sep)))))))))
((jazz)
    ;; root
    ;; + steps:(highest base) + cons-alt
    ;; + 'add'
    ;; + steps:rest
    (let* ((root->markup default-note-namer)
        (step->markup step->markup-ignatzek)
        (sep (make-simple-markup " "))
        (add-prefix (make-simple-markup " add")))

    (if (pair? full-markup)
        (make-line-markup (list (root->markup root) full-markup))

        (make-line-markup
            (list (root->markup root)
                partial-markup-prefix
                (make-super-markup
                    (make-line-markup
                        (list
                            ;; kludge alert: omit <= 5
                            ;; (markup-join (map step->markup
                            ;;     (cons (last base) cons-alt)) sep)

                            ;; This fixes:
                            ;; c   C5   -> C
                            ;; c:2  C5  2  -> C2
                            ;; c:3- Cm5  -> Cm
                            ;; c:6.9 C5 6add9 -> C6 add 9 (add?)
                            ;; ch = \chords { c c:2 c:3- c:6.97 }
                        (markup-join (map step->markup
                            (let ((tb (last base)))
                                (if (> (step-nr tb) 5)
                                    (cons tb cons-alt)
                                    cons-alt))) sep)
                    
                        (if (pair? rest)
                            add-prefix
                            empty-markup)
                    (markup-join (map step->markup rest) sep)
                partial-markup-suffix))))))))))
(else empty-markup))))

% Here begins the actual snippet:

chs = \transpose c' c' {
    \begin{verbatim}
    <c e g>1
    <c es g>  % m = minor triad
    <c e gis> \break
    <c e g bes> \break
    % triangle = maj
    <c es ges beses> \break
    <c e gis bes> \break
    <c es g b> \break
    <c e gis b> \break
    \end{verbatim}
    <c es ges bes> \break
    <c e g a>  % 6 = major triad with added sixth
    <c es g a>  % m6 = minor triad with added sixth
    <c e g bes d'> \break
    <c es g bes d' f' a' > \break
    <c es ges bes d'> \break
    <c e g bes des' f' > \break
    <c e g bes dis'> \break
    <c e g bes d' f'> \break
    <c e g bes d' fis'> \break
    <c e g bes d' f' a'> \break
    <c e g bes d' fis' as'> \break
    <c e g bes dis' fis'> \break
    <c e g bes d' f' as'> \break
    <c e g bes des' f' as'> \break
    <c e g bes d' fis'> \break
    <c e g des' f' as'> \break
    <c e g bes d' f' a'> \break
    <c e g bes des' f' a'> \break
    <c e g b d'> \break
    <c e g d'> % add9
    \end{verbatim}
<c es g f'>
<c e g b fis'> % Lydian
<c e g bes des' ees' fis' aes'> % altered chord

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
alternate Jazz notation

efullmusicJazzAlt = {
  <c e gis>1-\markup { "+" }
  <c e g b>-\markup {
    \normal-size-super
    \override #'(font-family . math) "N"
    \override #'(font-family . math) "M"
  }
  %c:3.5.7 = \markup { \override #'(font-family . math) "M" }
  %c:3.5.7 = \markup { \normal-size-super "maj7" }

  <c es ges>-\markup { \super "o" } % should be $\circ$ ?
  <c es ges bes>-\markup { \super \combine "o" "/" }
  <c es ges beses>-\markup { \super "o7" }
}

efullJazzAlt = #(sequential-music-to-chord-exceptions efullmusicJazzAlt #f)

epartialmusicJazzAlt = {
  <d>1-\markup { \normal-size-super "2" }
  <e>-\markup { "m" }
  <f>-\markup { \normal-size-super "sus4" }
  <g>-\markup { \normal-size-super "5" }
  % TODO, partial exceptions
  <c es f>-\markup { "m" }-\markup { \normal-size-super "sus4" }
  <c d es>-\markup { "m" }-\markup { \normal-size-super "sus2" }
}

epartialJazzAlt = #(sequential-music-to-chord-exceptions epartialmusicJazzAlt #f)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
\score {
  \new ChordNames {
    \set chordNameFunction = #ignatzek-chord-names
    \set instrumentName = "Ignatzek"
    \set shortInstrumentName = "Def"
    \chs
  }

  \new ChordNames {
    \set chordNameFunction = #jazz-chordnames
}
%% This is the Banter (1987) style. It gives exceedingly
%% verbose (wide) names, making the output file take up to 4 pages.

\new ChordNames {
  \set chordNameFunction = #banter-chordnames
  \override ChordName.font-size = #-3
  \set instrumentName = "Banter"
  \set shortInstrumentName = "Ban"
  \chs
}

\new Staff \transpose c c' { \chs }
>>
\layout {
  #(layout-set-staff-size 16)
  system-system-spacing.basic-distance = #0
  \context {
    \ChordNames
    \consists "Instrument_name_engraver"
  }
  \context {
    \Score
    \remove "Bar_number_engraver"
  }
}
<table>
<thead>
<tr>
<th>Chord</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>C7 #5</td>
<td>C7 b5</td>
</tr>
<tr>
<td>Cm △</td>
<td>Cm △ 9</td>
</tr>
<tr>
<td>C △ 9</td>
<td>C △ 9</td>
</tr>
<tr>
<td>C7 #5</td>
<td>C7 b5</td>
</tr>
<tr>
<td>Alt</td>
<td></td>
</tr>
<tr>
<td>Ban</td>
<td></td>
</tr>
<tr>
<td>Def</td>
<td></td>
</tr>
<tr>
<td>C7 #5</td>
<td>C7 b5</td>
</tr>
<tr>
<td>Cm △</td>
<td>Cm △ 9</td>
</tr>
<tr>
<td>C △ 9</td>
<td>C △ 9</td>
</tr>
<tr>
<td>C7 #5</td>
<td>C7 b5</td>
</tr>
<tr>
<td>C</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chord</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>C6</td>
<td>C6</td>
</tr>
<tr>
<td>Cm6</td>
<td>Cm6</td>
</tr>
<tr>
<td>C9</td>
<td>C9</td>
</tr>
<tr>
<td>Cm9</td>
<td>Cm9</td>
</tr>
<tr>
<td>Alt</td>
<td></td>
</tr>
<tr>
<td>Ban</td>
<td></td>
</tr>
<tr>
<td>Def</td>
<td></td>
</tr>
<tr>
<td>C6</td>
<td>C6</td>
</tr>
<tr>
<td>Cm6</td>
<td>Cm6</td>
</tr>
<tr>
<td>C9</td>
<td>C9</td>
</tr>
<tr>
<td>Cm9</td>
<td>Cm9</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chord</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cm13</td>
<td>Cm13</td>
</tr>
<tr>
<td>C11 k3</td>
<td>C11 k3</td>
</tr>
<tr>
<td>Cm7 b5 9</td>
<td>Cm7 b5 9</td>
</tr>
<tr>
<td>C7 b9</td>
<td>C7 b9</td>
</tr>
<tr>
<td>Alt</td>
<td></td>
</tr>
<tr>
<td>Ban</td>
<td></td>
</tr>
<tr>
<td>Def</td>
<td></td>
</tr>
<tr>
<td>Cm13</td>
<td>Cm13</td>
</tr>
<tr>
<td>C11 k3</td>
<td>C11 k3</td>
</tr>
<tr>
<td>Cm7 b5 9</td>
<td>Cm7 b5 9</td>
</tr>
<tr>
<td>C7 b9</td>
<td>C7 b9</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chord</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>C7 #9</td>
<td>C7 #9</td>
</tr>
<tr>
<td>C11 #9</td>
<td>C11 #9</td>
</tr>
<tr>
<td>C7 #11</td>
<td>C7 #11</td>
</tr>
<tr>
<td>C11 b13</td>
<td>C11 b13</td>
</tr>
<tr>
<td>Alt</td>
<td></td>
</tr>
<tr>
<td>Ban</td>
<td></td>
</tr>
<tr>
<td>Def</td>
<td></td>
</tr>
<tr>
<td>C7 #9</td>
<td>C7 #9</td>
</tr>
<tr>
<td>C11 #9</td>
<td>C11 #9</td>
</tr>
<tr>
<td>C7 #11</td>
<td>C7 #11</td>
</tr>
<tr>
<td>C11 b13</td>
<td>C11 b13</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chord</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>C7 b9</td>
<td>C7 b9</td>
</tr>
<tr>
<td>C11 b9</td>
<td>C11 b9</td>
</tr>
<tr>
<td>C7 #11</td>
<td>C7 #11</td>
</tr>
<tr>
<td>C11 b13</td>
<td>C11 b13</td>
</tr>
<tr>
<td>Alt</td>
<td></td>
</tr>
<tr>
<td>Ban</td>
<td></td>
</tr>
<tr>
<td>Def</td>
<td></td>
</tr>
<tr>
<td>C7 b9</td>
<td>C7 b9</td>
</tr>
<tr>
<td>C11 b9</td>
<td>C11 b9</td>
</tr>
<tr>
<td>C7 #11</td>
<td>C7 #11</td>
</tr>
<tr>
<td>C11 b13</td>
<td>C11 b13</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chord</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>C7 b9 13</td>
<td>C7 b9 13</td>
</tr>
<tr>
<td>C11 b9 13</td>
<td>C11 b9 13</td>
</tr>
<tr>
<td>C7 #11</td>
<td>C7 #11</td>
</tr>
<tr>
<td>C11 b13</td>
<td>C11 b13</td>
</tr>
<tr>
<td>Alt</td>
<td></td>
</tr>
<tr>
<td>Ban</td>
<td></td>
</tr>
<tr>
<td>Def</td>
<td></td>
</tr>
<tr>
<td>C7 b9 13</td>
<td>C7 b9 13</td>
</tr>
<tr>
<td>C11 b9 13</td>
<td>C11 b9 13</td>
</tr>
<tr>
<td>C7 #11</td>
<td>C7 #11</td>
</tr>
<tr>
<td>C11 b13</td>
<td>C11 b13</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chord</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>C7 b9 13</td>
<td>C7 b9 13</td>
</tr>
<tr>
<td>C11 b9 13</td>
<td>C11 b9 13</td>
</tr>
<tr>
<td>C7 #11</td>
<td>C7 #11</td>
</tr>
<tr>
<td>C11 b13</td>
<td>C11 b13</td>
</tr>
<tr>
<td>Alt</td>
<td></td>
</tr>
<tr>
<td>Ban</td>
<td></td>
</tr>
<tr>
<td>Def</td>
<td></td>
</tr>
<tr>
<td>C7 b9 13</td>
<td>C7 b9 13</td>
</tr>
<tr>
<td>C11 b9 13</td>
<td>C11 b9 13</td>
</tr>
<tr>
<td>C7 #11</td>
<td>C7 #11</td>
</tr>
<tr>
<td>C11 b13</td>
<td>C11 b13</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chord</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>C7 b9</td>
<td>C7 b9</td>
</tr>
<tr>
<td>C11 b9</td>
<td>C11 b9</td>
</tr>
<tr>
<td>C7 #11</td>
<td>C7 #11</td>
</tr>
<tr>
<td>C11 b13</td>
<td>C11 b13</td>
</tr>
<tr>
<td>Alt</td>
<td></td>
</tr>
<tr>
<td>Ban</td>
<td></td>
</tr>
<tr>
<td>Def</td>
<td></td>
</tr>
<tr>
<td>C7 b9</td>
<td>C7 b9</td>
</tr>
<tr>
<td>C11 b9</td>
<td>C11 b9</td>
</tr>
<tr>
<td>C7 #11</td>
<td>C7 #11</td>
</tr>
<tr>
<td>C11 b13</td>
<td>C11 b13</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chord</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>C7 b9 13</td>
<td>C7 b9 13</td>
</tr>
<tr>
<td>C11 b9 13</td>
<td>C11 b9 13</td>
</tr>
<tr>
<td>C7 #11</td>
<td>C7 #11</td>
</tr>
<tr>
<td>C11 b13</td>
<td>C11 b13</td>
</tr>
<tr>
<td>Alt</td>
<td></td>
</tr>
<tr>
<td>Ban</td>
<td></td>
</tr>
<tr>
<td>Def</td>
<td></td>
</tr>
<tr>
<td>C7 b9 13</td>
<td>C7 b9 13</td>
</tr>
<tr>
<td>C11 b9 13</td>
<td>C11 b9 13</td>
</tr>
<tr>
<td>C7 #11</td>
<td>C7 #11</td>
</tr>
<tr>
<td>C11 b13</td>
<td>C11 b13</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chord</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>C7 #11</td>
<td>C7 #11</td>
</tr>
<tr>
<td>C9 #11</td>
<td>C9 #11</td>
</tr>
<tr>
<td>C7 #9</td>
<td>C7 #9</td>
</tr>
<tr>
<td>C9 #13</td>
<td>C9 #13</td>
</tr>
<tr>
<td>Alt</td>
<td></td>
</tr>
<tr>
<td>Ban</td>
<td></td>
</tr>
<tr>
<td>Def</td>
<td></td>
</tr>
<tr>
<td>C7 #11</td>
<td>C7 #11</td>
</tr>
<tr>
<td>C9 #11</td>
<td>C9 #11</td>
</tr>
<tr>
<td>C7 #9</td>
<td>C7 #9</td>
</tr>
<tr>
<td>C9 #13</td>
<td>C9 #13</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chord</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>C7 #9</td>
<td>C7 #9</td>
</tr>
<tr>
<td>C11 b9</td>
<td>C11 b9</td>
</tr>
<tr>
<td>C7 #11</td>
<td>C7 #11</td>
</tr>
<tr>
<td>C11 b13</td>
<td>C11 b13</td>
</tr>
<tr>
<td>Alt</td>
<td></td>
</tr>
<tr>
<td>Ban</td>
<td></td>
</tr>
<tr>
<td>Def</td>
<td></td>
</tr>
<tr>
<td>C7 #9</td>
<td>C7 #9</td>
</tr>
<tr>
<td>C11 b9</td>
<td>C11 b9</td>
</tr>
<tr>
<td>C7 #11</td>
<td>C7 #11</td>
</tr>
<tr>
<td>C11 b13</td>
<td>C11 b13</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chord</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>C7 #11</td>
<td>C7 #11</td>
</tr>
<tr>
<td>C9 #11</td>
<td>C9 #11</td>
</tr>
<tr>
<td>C7 #9</td>
<td>C7 #9</td>
</tr>
<tr>
<td>C9 #13</td>
<td>C9 #13</td>
</tr>
<tr>
<td>Alt</td>
<td></td>
</tr>
<tr>
<td>Ban</td>
<td></td>
</tr>
<tr>
<td>Def</td>
<td></td>
</tr>
<tr>
<td>C7 #11</td>
<td>C7 #11</td>
</tr>
<tr>
<td>C9 #11</td>
<td>C9 #11</td>
</tr>
<tr>
<td>C7 #9</td>
<td>C7 #9</td>
</tr>
<tr>
<td>C9 #13</td>
<td>C9 #13</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chord</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>C7 #11</td>
<td>C7 #11</td>
</tr>
<tr>
<td>C9 #11</td>
<td>C9 #11</td>
</tr>
<tr>
<td>C7 #9</td>
<td>C7 #9</td>
</tr>
<tr>
<td>C9 #13</td>
<td>C9 #13</td>
</tr>
<tr>
<td>Alt</td>
<td></td>
</tr>
<tr>
<td>Ban</td>
<td></td>
</tr>
<tr>
<td>Def</td>
<td></td>
</tr>
<tr>
<td>C7 #11</td>
<td>C7 #11</td>
</tr>
<tr>
<td>C9 #11</td>
<td>C9 #11</td>
</tr>
<tr>
<td>C7 #9</td>
<td>C7 #9</td>
</tr>
<tr>
<td>C9 #13</td>
<td>C9 #13</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chord</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>C7 #11</td>
<td>C7 #11</td>
</tr>
<tr>
<td>C9 #11</td>
<td>C9 #11</td>
</tr>
<tr>
<td>C7 #9</td>
<td>C7 #9</td>
</tr>
<tr>
<td>C9 #13</td>
<td>C9 #13</td>
</tr>
<tr>
<td>Alt</td>
<td></td>
</tr>
<tr>
<td>Ban</td>
<td></td>
</tr>
<tr>
<td>Def</td>
<td></td>
</tr>
<tr>
<td>C7 #11</td>
<td>C7 #11</td>
</tr>
<tr>
<td>C9 #11</td>
<td>C9 #11</td>
</tr>
<tr>
<td>C7 #9</td>
<td>C7 #9</td>
</tr>
<tr>
<td>C9 #13</td>
<td>C9 #13</td>
</tr>
</tbody>
</table>
Chords with stretched fingering for FretBoards and TabVoice

Sometimes chords with a stretched fingering are required. If not otherwise specified the context-property maximumFretStretch is set to 4, though. Resulting in a warning about "No string for pitch ..." and the note is omitted. You may set maximumFretStretch to an appropriate value or explicely assign string-numbers to all notes of a chord.

%% The code below will print two warnings, which may be omitted by uncommenting: %#(for-each (lambda (x) (ly:expect-warning "No string for pitch")) (iota 2))

```plaintext
mus = {
    <c' bes'>
    <c'\'2 bes'>
    \set maximumFretStretch = 5
    <c' bes'>
    <c'\'2 bes'\'1>
}

ew FretBoards \mus
new TabVoice \mus
```

Clusters

Clusters are a device to denote that a complete range of notes is to be played.

```plaintext
fragment = \relative c' {
    c4 f <e d'>4
    <g a>8 <e a> a4 c2 <d b>4
    e2 c
}

ew Staff \fragment
new Staff \makeClusters \fragment
```
Controllare il posizionamento delle diteggiature di un accordo

Il posizionamento dei numeri della diteggiatura può essere regolato in modo preciso. Perché l’orientamento funzioni, occorre usare il costrutto per gli accordi <> anche per le note singole.
Si può impostare in modo simile l’orientamento dei numeri di corda e delle diteggiature della mano destra.

\relative c' {
\set fingeringOrientations = #'(left)
  <c-1 e-3 a-5>4
\set fingeringOrientations = #'(down)
  <c-1 e-3 a-5>4
\set fingeringOrientations = #'(down right up)
  <c-1 e-3 a-5>4
\set fingeringOrientations = #'(up)
  <c-1 e-3 a-5>4
\set fingeringOrientations = #'(left)
  <c-1>2
\set fingeringOrientations = #'(down)
  <e-3>2
\set stringNumberOrientations = #'(up left down)
  <f\3 a\2 c\1>1
\set strokeFingerOrientations = #'(down right up)
  <c\rightHandFinger #1 e\rightHandFinger #2 c\rightHandFinger #4 >
}

Cross-staff chords - beaming problems workaround

Sometimes it is better to use stems from the upper staff for creating cross-staff chords, because no problems with automatic beam collision avoidance then arise. If the stems from the lower staff were used in the following example, it would be necessary to change the automatic beam collision avoidance settings so that it doesn’t detect collisions between staves using \override Staff.Beam.collision-voice-only = ##t

\new PianoStaff <<
\new Staff = up
  \relative c' {
    <<
      { r4

\freeDynamics""
Customizing the chord grid style

Custom divisions of chord squares can be defined through the \measure-division-lines-alist and \measure-division-chord-placement-alist properties of ChordSquare. These are both alists. Their keys are measure divisions, namely lists which give the fraction of the measure that each chord (or rest, or skip) represents. More precisely, a measure division alist is made of positive, exact numbers adding up to 1, for example: '(1/2 1/4 1/4). The exactness requirement means that, e.g., 1/2 is valid but not 0.5.

The values in \measure-division-lines-alist are lists of lines, which are represented as (x1 y1 x2 y2). The line starts at the point (x1 . y1) and ends at (x2 . y2). Coordinates are expressed in the [-1, 1] scale relative to the extent of the square.

The values in \measure-division-chord-placement-alist are lists of (x . y) pairs giving the placement of the respective chords.

This example defines a peculiar chord grid style that has a rule for measures divided in three equal parts.
\override ChordSquare.measure-division-lines-alist =
  #'(((1) . ())
     ((1/3 1/3 1/3) . ((-1 -0.4 0 1) (0 -1 1 0.4))))
\override ChordSquare.measure-division-chord-placement-alist =
  #'(((1) . ((0 . 0)))
     ((1/3 1/3 1/3) . ((-0.7 . 0.5) (0 . 0) (0.7 . -0.5))))
}
\chordmode {
  \time 3/4
  c2.
  c4 c4 c4
}

Customizing the no-chord symbol

By default, rests in a ChordNames context cause the “N.C.” symbol to be printed. This markup can be customized.

<<
  \chords {
    R1
    \set noChordSymbol = "---"
    R1
    \set noChordSymbol = \markup \italic "Ssh!"
    R1
  }
  { R1*3 }
>>

Displaying complex chords

Here is a way to display a chord where the same note is played twice with different accidentals.

fixA = {
  \once \override Stem.length = #11
}

fixB = {
  \once \override NoteHead.X-offset = #1.7
  \once \override Stem.length = #7
  \once \override Stem.rotation = #'(45 0 0)
Manually break figured bass extenders for only some numbers

Figured bass often uses extenders to indicate continuation of the corresponding step. However, in this case LilyPond is in greedy-mode and uses extenders whenever possible. To break individual extenders, one can simply use a modifier \! to a number, which breaks any extender attributed to that number right before the number.

```
bassfigures = \figuremode {  
  \set useBassFigureExtenders = ##t  
  <6 4>4 <6 4\!> <6 4\!> | <6\! 4\!> <6 4\!> <6 4\!> <6 4>  
}
```

Print ChordNames with same root and different bass as slash and bass-note

To print subsequent ChordNames only differing in its bass note as slash and bass note use the here defined engraver. The behaviour may be controlled in detail by the \chordChanges context property.

```
#(define Bass_changes_equal_root_ engraver  
(lambda (ctx)  
  "For sequential @code{ChordNames} with same root, but different bass, the root markup is dropped: D D/C D/B -> D /C /B  
  The behaviour may be controlled by setting the @code{chordChanges} context-property."
  (let (((chord-pitches '())  
      (last-chord-pitches '())  
      (bass-pitch #f))  
    (make- engraver  
      ((initialize this- engraver)  
        (chordChanges ctx)  
  ```
(let ((chord-note-namer (ly:context-property ctx 'chordNoteNamer)))
  ;; Set 'chordNoteNamer, respect user setting if already done
  (ly:context-set-property! ctx 'chordNoteNamer
    (if (procedure? chord-note-namer)
        chord-note-namer
        note-name->markup))))

(listeners
  ((note-event this-engraver event)
    (let* ((pitch (ly:event-property event 'pitch))
            (pitch-name (ly:pitch-notename pitch))
            (pitch-alt (ly:pitch-alteration pitch))
            (bass (ly:event-property event 'bass #f))
            (inversion (ly:event-property event 'inversion #f)))
      ;; Collect notes of the chord
      ;; - to compare inversed chords we need to collect the bass note
      ;;  as usual member of the chord, whereas an added bass must be
      ;;  treated separate from the usual chord-notes
      ;; - notes are stored as pairs containing their
      ;;  pitch-name (an integer), i.e. disregarding their octave and
      ;;  their alteration
      (cond (bass (set! bass-pitch pitch))
            (inversion
             (set! bass-pitch pitch)
             (set! chord-pitches
              (cons (cons pitch-name pitch-alt) chord-pitches)))
            (else
             (set! chord-pitches
              (cons (cons pitch-name pitch-alt) chord-pitches))))))

(acknowledgers
  ((chord-name-interface this-engraver grob source-engraver)
    (let ((chord-changes (ly:context-property ctx 'chordChanges #f)))
      ;; If subsequent chords are equal apart from their bass,
      ;; reset the 'text-property.
      ;; Equality is done by comparing the sorted lists of this chord's
      ;; elements and the previous chord. Sorting is needed because
      ;; inverted chords may have a different order of pitches.
      ;; 'chord-changes' needs to be true
      (if (and bass-pitch
                   chord-changes
                   (equal?
                    (sort chord-pitches car<)
                    (sort last-chord-pitches car<)))
        (ly:grob-set-property! grob 'text
          (make-line-markup
           (list
            (ly:context-property ctx 'slashChordSeparator)
            ((ly:context-property ctx 'chordNoteNamer)
              bass-pitch
              (ly:context-property ctx 'chordNameLowercaseMinor))))))
      (set! last-chord-pitches chord-pitches)
      (set! chord-pitches '())
      (set! bass-pitch #f))))
Mostrare gli accordi nei cambi

Per impostazione predefinita, ogni accordo inserito viene visualizzato; tale comportamento può essere modificato in modo che i nomi degli accordi siano mostrati solo all’inizio delle linee e quando l’accordo cambia.
Canzoniere semplice
Mettendo insieme nomi degli accordi, melodia e testo si ottiene un canzoniere (in inglese “lead sheet”):;

Modello di rigo singolo con note, testo e accordi
Ecco il modello di un comune spartito semplificato (lead sheet): include linea melodica, testo vocale e sigle degli accordi.

Ecco il modello di un comune spartito semplificato (lead sheet): include linea melodica, testo vocale e sigle degli accordi.

melody = \relative c' {
  \clef treble
  \key c \major
  \time 4/4

  a4 b c d
}

text = \lyricmode {
  Aaa Bee Cee Dee
}
harmonies = \chordmode {
  a2 c
}

\score {
  <<
    \new ChordNames {
      \set chordChanges = ##t
      \harmonies
    }
    \new Voice = "one" { \autoBeamOff \melody }
    \new Lyrics \lyricsto "one" \text
  >>
  \layout { }
  \midi { }
}

\begin{diagram}
\begin{array}{c}
A & C \\
\end{array}
\end{diagram}

Aaa Bee Cee Dee

**Single staff template with notes, lyrics, chords and frets**

Here is a simple lead sheet template with melody, lyrics, chords and fret diagrams.

verseI = \lyricmode {
  \set stanza = "1."
  This is the first verse
}

verseII = \lyricmode {
  \set stanza = "2."
  This is the second verse.
}

theChords = \chordmode {
  % insert chords for chordnames and fretboards here
  c2 g4 c
}

staffMelody = \relative c' {
  \key c \major
  \clef treble
  % Type notes for melody here
  c4 d8 e f4 g
  \bar "|."
}

\score {
  <<

Modello di rigo singolo con note e accordi

Vuoi preparare uno spartito semplificato (lead sheet) con melodia e accordi? La tua ricerca è finita!

**melody** = \relative c' {\
  \clef treble\
  \key c \major\
  \time 4/4\
  f4 e8 [c] d4 g a2 ~ a
}

**harmonies** = \chordmode {\
  c4:m f:min7 g:maj c:aug\
  d2:dim b4:5 e:sus
}

```
\score {\
  <<<\new ChordNames {\
    \set chordChanges = ##t\
    \harmonies\
  }\
  \new Staff \melody\
  >>>\layout { }
}
```
Vertically centering paired figured bass extenders

Where figured bass extender lines are being used by setting `useBassFigureExtenders` to true, pairs of congruent figured bass extender lines are vertically centered if `figuredBassCenterContinuations` is set to true.

```latex
<< \relative c' { 
  c8 c b b a a c16 c b b 
  c8 c b b a a c16 c b b 
  c8 c b b a a c b b } \figures { \set useBassFigureExtenders = ##t
  <6+ 4 3>4 <6 4 3>8 r
  <6+ 4 3>4 <6 4 3>8 <4 3+>16 r
  \set figuredBassCenterContinuations = ##t
  <6+ 4 3>4 <6 4 3>8 r
  <6+ 4 3>4 <6 4 3>8 <4 3+>16 r
  \set figuredBassCenterContinuations = ##f
  <6+ 4 3>4 <6 4 3>8 r
  <6+ 4 3>4 <6 4 3>8 <4 3+>8 } >>
```

Volta sotto gli accordi

Aggiungendo l’incisore `Volta_engraver` al rigo, è possibile inserire le volte sotto gli accordi.

```latex
\score { 
  \chords { 
    c1 
    c1 
  } 
  \new Staff \with { 
    \consists "Volta_engraver" 
  } 
  \repeat volta 2 { c'1 } 
```
\alternative{c'}

\layout{}
\context{}
\Score
\remove"Volta_engraver"

C C
Keyboards

Sezione “Keyboard and other multi-staff instruments” in *Guida alla Notazione*

Accordion-discant symbols

This snippet has been obsoleted by predefined markup commands, see 'Discant symbols' in the Notation Reference. It’s still useful as a simple demonstration of how to combine symbols: the placement of the symbols added with \markup can be tweaked by changing the \translate-scaled arguments. \translate-scaled is used here rather than \translate in order to let the positioning of the symbol parts adapt to changes of font-size.

```
discant = \markup {
  \musicglyph "accordion.discant"
}
dot = \markup {
  \musicglyph "accordion.dot"
}

\layout { ragged-right = ##t }

% 16 voets register
accBasson = \markup {
  \combine
  \discant
  \translate-scaled #'(0 . 0.5) \dot
}

% een korig 8 en 16 voets register
accBandon = \markup {
  \combine
  \discant
  \combine
    \translate-scaled #'(0 . 0.5) \dot
    \translate-scaled #'(0 . 1.5) \dot
}

accVCello = \markup {
  \combine
  \discant
  \combine
    \translate-scaled #'(0 . 0.5) \dot
    \translate-scaled #'(0 . 1.5) \dot
    \translate-scaled #'(1 . 1.5) \dot
}

% 4-8-16 voets register
accHarmon = \markup {
  \combine
  \discant
  \combine
    \translate-scaled #'(0 . 0.5) \dot
```
\begin{verbatim}
\text{accTrombon} = ^\text{\textbackslash markup} \{ \\
\text{\textbackslash discant} \\
\text{\textbackslash translate-scaled \#'(0 . 0.5) \textbackslash dot} \\
\text{\textbackslash translate-scaled \#'(0 . 1.5) \textbackslash dot} \\
\text{\textbackslash translate-scaled \#'(0 . 1.5) \textbackslash dot} \\
\text{\textbackslash translate-scaled \#'(1 . 1.5) \textbackslash dot} \\
\text{\textbackslash translate-scaled \#'(-1 . 1.5) \textbackslash dot} \\
\}
\end{verbatim}

\textit{\% eenkorig 4 en 16 voets register} 

\text{accOrgan} = ^\text{\textbackslash markup} \{ \\
\text{\textbackslash discant} \\
\text{\textbackslash translate-scaled \#'(0 . 0.5) \textbackslash dot} \\
\text{\textbackslash translate-scaled \#'(0 . 1.5) \textbackslash dot} \\
\}

\text{accMaster} = ^\text{\textbackslash markup} \{ \\
\text{\textbackslash discant} \\
\text{\textbackslash translate-scaled \#'(0 . 0.5) \textbackslash dot} \\
\text{\textbackslash translate-scaled \#'(0 . 1.5) \textbackslash dot} \\
\text{\textbackslash translate-scaled \#'(1 . 1.5) \textbackslash dot} \\
\text{\textbackslash translate-scaled \#'(-1 . 1.5) \textbackslash dot} \\
\text{\textbackslash translate-scaled \#'(0 . 2.5) \textbackslash dot} \\
\}

\text{accAccord} = ^\text{\textbackslash markup} \{ \\
\text{\textbackslash discant} \\
\text{\textbackslash translate-scaled \#'(0 . 1.5) \textbackslash dot} \\
\text{\textbackslash translate-scaled \#'(1 . 1.5) \textbackslash dot} \\
\text{\textbackslash translate-scaled \#'(-1 . 1.5) \textbackslash dot} \\
\text{\textbackslash translate-scaled \#'(0 . 2.5) \textbackslash dot} \\
\}
\end{verbatim}
accMusette = \markup {
   \combine
   \disct
   \combine
   \translate-scaled #'(0 . 1.5) \dot
   \combine
   \translate-scaled #'(1 . 1.5) \dot
   \translate-scaled #'(1 . 1.5) \dot
}

accCeleste = \markup {
   \combine
   \disct
   \combine
   \translate-scaled #'(0 . 1.5) \dot
   \translate-scaled #'(1 . 1.5) \dot
}

accOboe = \markup {
   \combine
   \disct
   \combine
   \translate-scaled #'(0 . 1.5) \dot
   \translate-scaled #'(0 . 2.5) \dot
}

accClarinet = \markup {
   \combine
   \disct
   \translate-scaled #'(0 . 1.5) \dot
}

accPiccolo = \markup {
   \combine
   \disct
   \translate-scaled #'(0 . 2.5) \dot
}

accViolin = \markup {
   \combine
   \disct
   \translate-scaled #'(0 . 1.5) \dot
   \combine
   \translate-scaled #'(1 . 1.5) \dot
   \translate-scaled #'(0 . 2.5) \dot
}

\relative c' {
   c4 d \accBassoon e f
   c4 d \accBandon e f
   c4 d \accVCello e f
Simboli di registro della fisarmonica

I simboli di registro della fisarmonica sono disponibili sia come \narkup sia come eventi musicali autonomi (perché i cambi di registro capitano solitamente tra reali eventi musicali). I registri bassi non sono troppo standardizzati. I comandi disponibili si trovano nella sezione «Registri della fisarmonica» della Guida alla notazione.

\(\text{\texttt{#(use-modules \ (lily \ accreg)\)}}\)

\(\text{\texttt{\new \ PianoStaff \ relative \{}}\)
\(\text{\texttt{\ new \ Staff \ relative \{}}\)
\(\text{\texttt{\ new \ Staff \ relative \{}}\)
\(\text{\texttt{\ new \ Staff \ relative \{}}\)
Changing the text for sustain markings

Staff.pedalSustainStrings can be used to set the text used for pedal down and up. Note that the only valid strings are those found in the list of pedal glyphs - the values used this snippet constitute an exhaustive list.

sustainNotes = { c4\sustainOn d e\sustainOff\sustainOn f\sustainOff }
Controllare il posizionamento delle diteggiature di un accordo

Il posizionamento dei numeri della diteggiatura può essere regolato in modo preciso. Perché l’orientamento funzioni, occorre usare il costrutto per gli accordi <> anche per le note singole. Si può impostare in modo simile l’orientamento dei numeri di corda e delle diteggiature della mano destra.

```latex
\relative c' {
    \set fingeringOrientations = #'(left)
    <c-1 e-3 a-5>4
    \set fingeringOrientations = #'(down)
    <c-1 e-3 a-5>4
    \set fingeringOrientations = #'(down right up)
    <c-1 e-3 a-5>4
    \set fingeringOrientations = #'(up)
    <c-1 e-3 a-5>4
    \set fingeringOrientations = #'(left)
    <c-1>2
    \set fingeringOrientations = #'(down)
    <e-3>2
    \set stringNumberOrientations = #'(up left down)
    <f\3 a\2 c\1>1
    \set strokeFingerOrientations = #'(down right up)
    <c\rightHandFinger #1 e\rightHandFinger #2 c\rightHandFinger #4 >
}
```
Creating slurs across voices

In some situations, it may be necessary to create slurs between notes from different voices. The solution is to add invisible notes to one of the voices, using \hideNotes.

This example is measure 235 of the Ciaconna from Bach’s 2nd Partita for solo violin, BWV 1004.

\relative c' {
    \slurUp
    \hideNotes a
    \unHideNotes f[
        \hideNotes a
        \unHideNotes fis[
            \unHideNotes g[
                \unHideNotes gis][
f][
    ]}
}

Cross-staff chords - beaming problems workaround

Sometimes it is better to use stems from the upper staff for creating cross-staff chords, because no problems with automatic beam collision avoidance then arise. If the stems from the lower staff were used in the following example, it would be necessary to change the automatic beam collision avoidance settings so that it doesn’t detect collisions between staves using \override

\new PianoStaff <<
\new Staff = up
\relative c' {
    \override Stem.cross-staff = ##t
    \override Stem.length = #19 % this is in half-spaces,
    % so it makes stems 9.5 staffspaces long
    \override Stem.Y-offset = #-6 % stems are normally lengthened
    % upwards, so here we must lower the stem by the amount
    % equal to the lengthening - in this case (19 - 7) / 2
    % (7 is default stem length)
Tremoli attraverso i righi

Dato che \repeat tremolo si aspetta esattamente due argomenti musicali per i tremoli di accordi, la nota o l’accordo che cambiano rigo in un tremolo che attraversa i righi devono essere posti tra parentesi graffe insieme al comando \change Staff.

\new PianoStaff <<
  \new Staff = "up" \relative c'' {
    \key a \major
    \time 3/8
    s4.
  }
  \new Staff = "down" \relative c'' {
    \key a \major
    \time 3/8
    \voiceOne
    \repeat tremolo 6 {
      <a e'>32
    }
    \change Staff = "up"
    \voiceTwo
    <cis a' dis>32
  }
>>
Fine-tuning pedal brackets

The appearance of pedal brackets may be altered in different ways.

```latex
\paper { ragged-right = #f }
\relative c'' {
  \once \override \Staff\.PianoPedalBracket.shorten-pair = #'(-7 . -2)
  \once \override \Staff\.PianoPedalBracket.edge-height = #'(0 . 3)
}\)
```

Indicare accordi trasversali al rigo con la parentesi quadrata dell’arpeggio

Una parentesi quadrata dell’arpeggio può indicare che delle note su due righi diversi devono essere suonate con la stessa mano. Per farlo, bisogna far sì che PianoStaff accetti gli arpeggi trasversali ai righi e gli arpeggi siano impostati nella forma della parentesi quadrata nel contesto PianoStaff.

(Debussy, Les collines d’Anacapri, m. 65)

```latex\new PianoStaff <<\set PianoStaff.connectArpeggios = ##t \override PianoStaff.Arpeggio.stencil = #ly:arpeggio::brew-chord-bracket \new Staff { \relative c' { \key b \major \time 6/8 \b8-(\arpeggio fis'-.\> cis-. \e-. gis-. b-.)!\fermata\laissezVibrer \bar "||" } } \new Staff { \relative c' {```
Modello per combo jazz

Ecco un modello piuttosto complesso, per un gruppo jazz. Si noti che tutti gli strumenti sono in \key c \major. Si tratta della tonalità reale; sarà trasposta automaticamente includendo la musica all’interno di una sezione \transpose.

\header {  
  title = "Song"  
  subtitle = "(tune)"  
  composer = "Me"  
  meter = "moderato"  
  piece = "Swing"  
  tagline = \markup {  
    \column {  
      "LilyPond example file by Amelie Zapf,"  
      "Berlin 07/07/2003"  
    }  
  }  
}  
%
% To make the example display in the documentation
\paper {  
  paper-width = 130  
}  
%#(set-global-staff-size 16)
\include "english.ly"
sl = {
    \override NoteHead.style = #'slash
    \hide Stem
}
sl = {
    \revert NoteHead.style
    \undo \hide Stem
}
crOn = \override NoteHead.style = #'cross
crOff = \revert NoteHead.style

%%% insert chord name style stuff here.

jazzChords = { }

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Keys'n'thangs %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

global = { \time 4/4 }

Key = { \key c \major }

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Horns %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% ------- Trumpet -------
trpt = \transpose c d \relative c'' { 
    \Key
    c1 | c | c |
}
trpHarmony = \transpose c' d { 
   \jazzChords
}

trumpet = {
   \global
   \clef treble
   << 
   \trpt
   >>
}

% ------- Alto Saxophone -------
alto = \transpose c a \relative c' { 
   \Key
   c1 | c | c |
}
altoHarmony = \transpose c' a { 
   \jazzChords
}
altoSax = {
   \global
   \clef treble
   <<
   \alco
% ------ Baritone Saxophone ------
bari = \transpose c a' \relative c {  
  \Key
  c1
  c1
  \sl
d4-"Solo" d d d
  \nsl
}
bariHarmony = \transpose c' a \chordmode {  
  \jazzChords s1 s d2:maj e:m7
}
bariSax = {  
  \global
  \clef treble
  <<
  \bari
  >>
}

% ------ Trombone ------
tbone = \relative c {  
  \Key
  c1 | c | c
}
tboneHarmony = \chordmode {  
  \jazzChords
}
trombone = {  
  \global
  \clef bass
  <<
  \tbone
  >>
}

% ############ Rhythm Section ############

% ------ Guitar ------
gtr = \relative c'' {  
  \Key
  c1
  \sl
  b4 b b b
  \nsl
  c1
}
gtrHarmony = \chordmode {  
  \jazzChords
s1 c2:min7+ d2:maj9
}
guitar = {
  \global
  \clef treble
  <<
  \gtr
  >>
}

%% ------ Piano ------
rhUpper = \relative c'' {
  \voiceOne
  \Key
  c1 | c | c
}
rhLower = \relative c' {
  \voiceTwo
  \Key
  e1 | e | e
}
lhUpper = \relative c' {
  \voiceOne
  \Key
  g1 | g | g
}
lhLower = \relative c {
  \voiceTwo
  \Key
  c1 | c | c
}

PianoRH = {
  \clef treble
  \global
  <<
  \new Voice = "one" \rhUpper
  \new Voice = "two" \rhLower
  >>
}
PianoLH = {
  \clef bass
  \global
  <<
  \new Voice = "one" \lhUpper
  \new Voice = "two" \lhLower
  >>
}
piano = {
  <<
\new Staff = "upper" \PianoRH 
\new Staff = "lower" \PianoLH

% ------ Bass Guitar -------
Bass = \relative c {
  \Key
  c | c | c
}
bass = {
  \global
  \clef bass
  <<
    \Bass
  >>
}

% ------ Drums ------
up = \drummode {
  \voiceOne
  hh4 <hh sn> hh <hh sn>
  hh4 <hh sn> hh <hh sn>
  hh4 <hh sn> hh <hh sn>
}
down = \drummode {
  \voiceTwo
  bd4 s bd s
  bd4 s bd s
  bd4 s bd s
}

drumContents = {
  \global
  <<
    \new DrumVoice \up
    \new DrumVoice \down
  >>
}

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% It All Goes Together Here %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

\score {
  <<
    \new StaffGroup = "horns" <<
    \new Staff = "trumpet" \with { instrumentName = "Trumpet" }
    \trumpet
    \new Staff = "altosax" \with { instrumentName = "Alto Sax" }
    \altosax
    \new ChordNames = "barichords" \with { instrumentName = "Trumpet" }
    \bariHarmony
    \new Staff = "barisax" \with { instrumentName = "Bari Sax" }
}
Song
(tune)

Me

moderato

Swing
Laissez vibrer ties

Laissez vibrer ties have a fixed size. Their formatting can be tuned using 'tie-configuration.'

\relative c' {
    <c e g>4\laissezVibrer r <c f g>\laissezVibrer r
    <c d f g>4\laissezVibrer r <c d f g>4.\laissezVibrer r8

    <c d e f>4\laissezVibrer r
    \override LaissezVibrerTieColumn.tie-configuration
    = #'((-7 ,DOWN)
        (-5 ,DOWN)
        (-3 ,UP)
        (-1 ,UP))
    <c d e f>4\laissezVibrer r
}
Modello per pianoforte (semplice)

Ecco un comune doppio pentagramma per pianoforte con un po' di note.

```
upper = \relative c' \{ 
  \clef treble
  \key c \major
  \time 4/4

  a4 b c d
}

lower = \relative c \{
  \clef bass
  \key c \major
  \time 4/4

  a2 c
}
```

```
\score { 
  \new PianoStaff \with { instrumentName = "Piano" } 
  \new Staff = "upper" \upper  
  \new Staff = "lower" \lower  
  \layout { } 
  \midi { } 
}
```

Modello per pianoforte con testo al centro

Invece di destinare un rigo a parte alla linea melodica e al suo testo, è possibile collocare il testo al centro di un doppio pentagramma per pianoforte.

```
upper = \relative c' \{ 
  \clef treble
  \key c \major
  \time 4/4

  a4 b c d
}
```
Modello per pianoforte con melodia e testo

Ecco un tipico formato per canzoni: un rigo con linea melodica e testo, e sotto l’accompagnamento per pianoforte.

```
lower = \relative c { 
  \clef bass 
  \key c \major 
  \time 4/4 
  a2 c 
}

text = \lyricmode { 
  Aaa Bee Cee Dee 
}

\score { 
  \new PianoStaff << 
    \new Staff = upper { \new Voice = "singer" \upper } 
    \new Lyrics \lyricsto "singer" \text 
    \new Staff = lower { \lower } 
  >> 
  \layout { } 
  \midi { } 
}

Aaa Bee Cee Dee
```

```
upper = \relative c'' { 
  \clef treble 
  \key c \major 
  \time 4/4 
  a4 b c d 
}

text = \lyricmode { 
  Aaa Bee Cee Dee 
}
```
Removing brace on first line of piano score

This snippet removes the first brace from a PianoStaff or a GrandStaff.
It may be useful when cutting and pasting the engraved image into existing music.
It uses `\alterBroken`.

```latex
\texttt{someMusic} = \{ \\
\ \ \ \ \texttt{\once \override Staff.Clef.stencil = ##f} \\
\ \ \ \ \texttt{\once \override Staff.TimeSignature.stencil = ##f} \\
\ \ \ \ \texttt{\repeat unfold 3 c1 \break} \\
\ \ \ \ \texttt{\repeat unfold 5 c1 \break} \\
\ \ \ \ \texttt{\repeat unfold 5 c1} \\
\}
```

```latex
\texttt{\score { \new PianoStaff \score { \new Voice = \"mel\" { \autoBeamOff \melody } \new Lyrics \lyricsto \text \new PianoStaff << \new Staff = \"upper\" \upper \new Staff = \"lower\" \lower >> >> \context { \Staff \RemoveEmptyStaves } } \midi { } }
```
Using autochange with more than one voice

Using autochange with more than one voice.

\score {
    \new PianoStaff <<
    \new Staff = "up" {
        \set Timing.beamExceptions = #'()\set Timing.beatStructure = #'(4)
        \new Voice {
            voiceOne
            \autoChange
            \relative c' {
                g8 a b c d e f g
                g,8 a b c d e f g
            }
        }
    }
}
Modello per gruppo vocale con riduzione per pianoforte automatica

Questo modello aggiunge una riduzione automatica per pianoforte alla tipica partitura vocale SATB illustrata in “Modello per complesso vocale”. Si dimostra così uno dei punti di forza di LilyPond – è possibile usare una definizione musicale più di una volta. Qualsiasi modifica venga fatta alle note delle voci (ad esempio, tenor\texttt{Music}) verrà applicata anche alla riduzione per pianoforte.

```latex
\new Voice {
\voiceTwo
\autoChange
\relative c' {
  g8 a b c d e f g
  g,,8 a b c d e f g
}
}

\new Staff = "down" {
  \clef bass
}
```

![Musical notation image]

```latex
\global {
  \key c \major
  \time 4/4
}
```

```latex
sopMusic = \relative {
  c''4 c c8[(b)] c4
}
```

```latex
sopWords = \lyricmode {
  hi hi hi hi
}
```
Keyboards

altoMusic = \relative { 
   e'4 f d e 
 } 

altoWords = \lyricmode { 
   ha ha ha ha 
 } 

tenorMusic = \relative { 
   g4 a f g 
 } 

tenorWords = \lyricmode { 
   hu hu hu hu 
 } 

bassMusic = \relative { 
   c4 c g c 
 } 

bassWords = \lyricmode { 
   ho ho ho ho 
 } 

\score { 
   \new ChoirStaff << 
   \new Lyrics = "sopranos" \with { 
      \override VerticalAxisGroup.staff-affinity = #DOWN 
   } 
   \new Staff = "women" << 
   \new Voice = "sopranos" \with { 
      \override VerticalAxisGroup.staff-affinity = #DOWN 
   } 
   \new Voice = "altos" \with { 
      \override VerticalAxisGroup.staff-affinity = #DOWN 
   } 
   \new Lyrics = "altos" 
   \new Lyrics = "tenors" \with { 
      \override VerticalAxisGroup.staff-affinity = #DOWN 
   } 
   \new Staff = "men" << 
   \new PianoStaff << 
   \new Staff << 
   \set Staff.printPartCombineTexts = ##f
\partCombine
<< \global \sopMusic >>
<< \global \altoMusic >>
>>
\new Staff <<
\clef bass
\set Staff.printPartCombineTexts = ##f
\partCombine
<< \global \tenorMusic >>
<< \global \bassMusic >>
>>

}
Percussion

Sezione “Percussion” in Guida alla Notazione

Adding drum parts

Using the powerful pre-configured tools such as the \drummode function and the DrumStaff context, inputting drum parts is quite easy: drums are placed at their own staff positions (with a special clef symbol) and have note heads according to the drum. Attaching an extra symbol to the drum or restricting the number of lines is possible.

```
drh = \drummode {
  cymc4.^^"crash" hhc16^^"h.h." hh hhc8 hho hhc8 hh16 hh
  hhc4 r4 r2
}
drl = \drummode {
  bd4 sn8 bd bd4 << bd ss >>
  bd8 tommh tommh bd toml toml bd tomfh16 tomfh
}
timb = \drummode {
  timh4 ssh timl8 ssh r timh r4
  ssh8 timl r4 cb8 cb
}
```

```
\score { <<
  \new DrumStaff \with {
    instrumentName = "timbales"
    drumStyleTable = #timbales-style
    \override StaffSymbol.line-count = #2
    \override BarLine.bar-extent = #'(1 . 1)
  }
  \override BarLine.bar-extent = #'(1 . 1)
  \new DrumStaff \with { instrumentName = "drums" }
  \new DrumVoice \stemUp \drh }
  \new DrumVoice \stemDown \drl }
} >>
```

```
\layout { }
\midi { \tempo 4 = 120 }
```
Cow and ride bell example

Two different bells, entered with 'cb' (cowbell) and 'rb' (ridebell).
\paper { tagline = ##f }

#(define mydrums '((ridebell default #f 3)
                 (cowbell default #f -2)))

\new DrumStaff \with { instrumentName = #'Different Bells' }

\drummode {
  \set DrumStaff.drumStyleTable = #((alist->hash-table mydrums)
  \set DrumStaff.clefPosition = 0.5
  \override DrumStaff.StaffSymbol.line-positions = '#(-2 3)
  \override Staff.BarLine.bar-extent = '#(-1.0 . 1.5)

  \time 2/4
  rb8 8 cb8 16 rb16-> ~ |
  16 8 16 cb8 8 |}

Different Bells

Customized drum notation in printed and MIDI output

Customized drum “pitch” names (suitable for a custom drum style, for example) may be used both in printed and MIDI output by defining such variables as drumPitchNames, drumStyleTable and midiDrumPitches, as demonstrated here. In short, this snippet:

- defines some "pitch" names,
- defines how they will be rendered,
- tells LilyPond to use them for layout,
- assigns pitches to the names,
- tells LilyPond to use them for MIDI output.

%% This snippet tries to amend
%% NR 2.5.1 Common notation for percussion - Custom percussion staves
%% http://lilypond.org/doc/v2.16/Documentation/notation/common-notation-for-percussion#custom-percussion-staves

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% To use custom drum pitch names for your score and midi you need to follow
%% this route:
%%
%% LAYOUT:

%% (1) Define a name and put it in 'drumPitchNames'
%% This can be done at toplevel with

```lilypond`

```
%\begin{verbatim}
\% (2) Define how it should be printed
\% Therefore put them into a top-level list, where each entry should
\% be of the form:
\% (my-name
\%   note-head-style-or-default
\%   articulation-type-or-\#f
\%   staff-position)
\%
\% Example:
\% #(define my-style
\% '(
\%   (my-name default tenuto -1)
\%   ; ...
\% ))
\% (3) Tell LilyPond to use these custom definitions, with
\% drumStyleTable = #(alist->hash-table my-style)
\% in a \layout or \with block
\%
\% Now we’re done for layout. Here is a short but complete example:
\% \new DrumStaff
\% \with { drumStyleTable = #(alist->hash-table my-style) }
\% \drummode { my-name }
\%
\% MIDI:

%\begin{verbatim}
% (1) Again at top-level, assign a pitch to your custom note name
% midiDrumPitches.my-name = ges
% Note that you have to use the name, which is in drumPitchNames, no alias
% (2) Tell LilyPond to use this pitch(es), with
% drumPitchTable = #(alist->hash-table midiDrumPitches)
%
% Example:
% \score {
%   \new DrumStaff
%   \with {
%     drumStyleTable = #(alist->hash-table my-style)
%     drumPitchTable = #(alist->hash-table midiDrumPitches)
%   }
%   \drummode { my-name4 }
%   \layout {}
%   \midi {}
% }
%
% TESTING
%
% To test whether all is fine, run the following sequence in terminal:
% lilypond my-file.ly
\end{verbatim}

"
%% midi2ly my-file.midi
%% gedit my-file-midi.ly
%%
%% This will do the following:
%% 1. create pdf and midi
%% 2. transform the midi back to a .ly-file
%%   (note: midi2ly is not always good in correctly identifying enharmonic pitches)
%% 3. open this file in gedit (or use another editor)
%% Now watch what you've got.
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% FULL EXAMPLE
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%(define djembe
  '((dbass default #f -2)
    (dbassmute default stopped -2)
    (dopen default #f 0)
    (dopenmute default stopped 0)
    (dslap default #f 2)
    (dslapmute default stopped 2)))

midiDrumPitches.dbass = g
midiDrumPitches.dbassmute = fis
midiDrumPitches.dopen = a
midiDrumPitches.dopenmute = gis
midiDrumPitches.dslap = b
midiDrumPitches.dslapmute = ais

one = \drummode { r4 dba4 do ds r dbm dom dsm }

\score {

  \new DrumStaff
  \with {
    \override StaffSymbol.line-count = #3
    instrumentName = "Djembe 
    drumStyleTable = #(alist->hash-table djembe)
    drumPitchTable = #(alist->hash-table midiDrumPitches)
Heavily customized polymetric time signatures

Though the polymetric time signature shown was not the most essential item here, it has been included to show the beat of this piece (which is the template of a real Balkan song!).

```latex
\begin{music}
  melody = \relative c' \key g \major
  \compoundMeter #'(\(3 8\) \(2 8\) \(2 8\) \(3 8\) \(2 8\) \(2 8\) \(2 8\) \(2 8\) \(3 8\) \(2 8\) \(2 8\) \(2 8\) \(3 8\) \(2 8\) \(2 8\) \(2 8\) \(3 8\) \(2 8\) \(2 8\))
    c8 c c d4 c8 c b c b a4 g fis8 e d c b' c d e4-- fis8 g \break
    c, 4. d4 c4 d4. c4 d c2 d4. e4-- d4
    c4. d4 c4 d4. c4 d c2 d4. e4-- d4 \break
    c4. d4 c4 d4. c4 d c2 d4. e4-- d4 \break
  \end{music}

```
High and Low woodblock example

Two Woodblocks, entered with `wbh` (high woodblock) and `wbl` (low woodblock). The length of the barline has been altered with an `\override` command otherwise it would be too short. The positions of the two stafflines also have to be explicitly defined.

```latex
\paper { tagline = ##f }

% These lines define the position of the woodblocks in the stave;
% if you like, you can change it or you can use special note heads
% for the woodblocks.
#(define mydrums '((hiwoodblock default #f 3)
  (lowwoodblock default #f -2)))

woodstaff = {
  % This defines a staff with only two lines.
  % It also defines the positions of the two lines.
  \override Staff.StaffSymbol.line-positions = #'(2 3)

  % This is necessary; if not entered,
  % the barline would be too short!
  \override Staff.BarLine.bar-extent = #'(1.0 . 1.5)
  % small correction for the clef:
  \set DrumStaff.clefPosition = 0.5
}

\new DrumStaff {
  % with this you load your new drum style table
  \set DrumStaff.drumStyleTable = #(alist->hash-table mydrums)
}

\woodstaff

\drummode {
  \time 2/4
  wb18 16 16 8-> 8 |
  wb18 16 16-> ~ 16 16 r8 |
}
```
Modello per combo jazz

Ecco un modello piuttosto complesso, per un gruppo jazz. Si noti che tutti gli strumenti sono in \key c \major. Si tratta della tonalità reale; sarà trasposta automaticamente includendo la musica all'interno di una sezione \transpose.

\header {
  \title = "Song"
  \subtitle = "(tune)"
  \composer = "Me"
  \meter = "moderato"
  \piece = "Swing"
  \tagline = \markup {
    \column {
      "LilyPond example file by Amelie Zapf,"
      "Berlin 07/07/2003"
    }
  }
}

% To make the example display in the documentation
\paper {
  paper-width = 130
}

%%% Some macros %%%%%%%%

\sl = {
  \override NoteHead.style = #'slash
  \hide Stem
}
\nsl = {
  \revert NoteHead.style
  \undo \hide Stem
}
}\crOn = \override NoteHead.style = #'cross
\crOff = \revert NoteHead.style

%%% insert chord name style stuff here.

\jazzChords = { }

%%% Keys'n'thangs %%%%%%%%

\global = { \time 4/4 }
\Key = { \key c \major }

% ############## Horns ##############

% ------ Trumpet ------
\trpt = \transpose c d \relative c' {

\Key
  c1 | c | c |
}

trpHarmony = \transpose c' d {
  \jazzChords
}

trumpet = {
  \global
  \clef treble
  <<
    \trpt
  >>
}

% ------ Alto Saxophone ------

alto = \transpose c a \relative c' {
  \Key
  c1 | c | c |
}

altoHarmony = \transpose c' a {
  \jazzChords
}

altoSax = {
  \global
  \clef treble
  <<
    \alto
  >>
}

% ------ Baritone Saxophone ------

bari = \transpose c a' \relative c {
  \Key
  c1
  c1
  \sl
  d4^"Solo" d d d
  \nsl
}

bariHarmony = \transpose c' a \chordmode {
  \jazzChords s1 s d2:maj e:m7
}

bariSax = {
  \global
  \clef treble
  <<
    \bari
  >>
}

% ------ Trombone ------

tbone = \relative c {
\Key
  c1 | c | c
}
tboneHarmony = \chordmode {
  \jazzChords
}
trombone = {
  \global
  \clef bass
  <<
  \tbone
  >>
}

% ############ Rhythm Section ############

% ------ Guitar ------
gtr = \relative c'' {
  \Key
  c1
  \sl
  b4 b b b
  \ns1
  c1
}
gtrHarmony = \chordmode {
  \jazzChords
  s1 c2:min7+ d2:maj9
}
guitar = {
  \global
  \clef treble
  <<
  \gtr
  >>
}

%% ------ Piano ------

rhUpper = \relative c'' {
  \voiceOne
  \Key
  c1 | c | c
}
rhLower = \relative c' {
  \voiceTwo
  \Key
  e1 | e | e
}
lhUpper = \relative c' {
  \voiceOne
  \Key
\begin{verbatim}
g1 | g | g
\}
lhLower = \relative c { \voiceTwo \Key c1 | c | c
\}
PianoRH = { \clef treble \global \new Voice = \textsc{one} \rhUpper \new Voice = \textsc{two} \rhLower
\}
PianoLH = { \clef bass \global \new Voice = \textsc{one} \lhUpper \new Voice = \textsc{two} \lhLower
\}
piano = { \new Staff = \textsc{upper} \PianoRH \new Staff = \textsc{lower} \PianoLH
\}
\%
\% ------- Bass Guitar -------
Bass = \relative c { \Key c1 | c | c
\}
bass = { \global \clef bass \new Bass
\}
\%
\% ------- Drums -------
up = \drummode { \voiceOne hh4 <hh sn> hh <hh sn> hh4 <hh sn> hh <hh sn> hh4 <hh sn> hh <hh sn>
\}
\end{verbatim}
down = drummode {
    \voiceTwo
    bd4 s bd s
    bd4 s bd s
    bd4 s bd s
}

 drumContents = {
    \global
    <<
    \new DrumVoice \up
    \new DrumVoice \down
    >>
} %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% It All Goes Together Here %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

\score {
    <<
    \new StaffGroup = "horns" <<
    \new Staff = "trumpet" \with { instrumentName = "Trumpet" }
    \trumpet
    \new Staff = "altosax" \with { instrumentName = "Alto Sax" }
    \altoSax
    \new ChordNames = "barichords" \with { instrumentName = "Trumpet" }
    \bariHarmony
    \new Staff = "barisax" \with { instrumentName = "Bari Sax" }
    \bariSax
    \new Staff = "trombone" \with { instrumentName = "Trombone" }
    \trombone
    >>
    \new StaffGroup = "rhythm" <<
    \new ChordNames = "chords" \gtrHarmony
    \new Staff = "guitar" \with { instrumentName = "Guitar" }
    \guitar
    \new PianoStaff = "piano" \with {
        instrumentName = "Piano"
        midiInstrument = "acoustic grand"
    }
    \piano
    \new Staff = "bass" \with { instrumentName = "Bass" }
    \bass
    \new DrumStaff \with { instrumentName = "Drums" }
    \drumContents
    >>
    >>
    \context { \Staff \RemoveEmptyStaves }
    \context {
        \Score
        \override BarNumber.padding = #3
    }
}
Song
(tune)

moderato

Swing

\override RehearsalMark.padding = \#2
\skipBars = \##f

\midi { }
}

Drums
Bass
Piano
Trombone
Bari Sax
Alto Sax
Trumpet

Cm$^{\Delta}$

C$^{\Delta}$

\times 44
\clef percussion
\times 44
\clef G
\times 44
\clef F
\times 44
\clef F

C$^{\Delta}$

D$^{\Delta}$ $^{9}$
Percussion beaters

Graphic symbols for percussion instruments are not natively supported; however it is possible to include such symbols, either as an external EPS file or as embedded PostScript code inside a markup, as demonstrated in this example.

```
stick = \markup {
  \with-dimensions #'(0 . 5) #'(0 . 5)
  \postscript "
  0 6 translate
  0.8 -0.8 scale
  0 0 0 setrgbcolor
  [] 0 setdash
  1 setlinewidth
  0 setlinejoin
  0 setlinecap
  gsave [1 0 0 1 0 0] concat
  gsave [1 0 0 1 -3.5406095 -199.29342] concat
  gsave
  0 0 0 setrgbcolor
  newpath
  7.1434065 200.94354 moveto
  7.2109628 200.90454 7.2785188 200.86554 7.3460747 200.82654 curveto
  8.8625264 203.92115 8.0029664 202.43233 7.1434065 200.94354 curveto
  closepath
eofill
  grestore
  gsave
  <<
    /ShadingType 3
    /ColorSpace /DeviceRGB
    /Coords [113.13708 207.87465 0 113.13708 207.87465 16.162441]
    /Extend [true true]
    /Domain [0 1]
    /Function <<
    /FunctionType 3
    /Functions
    [ ]
    <<
    /FunctionType 2
```

\score { 
\b1 \"\textsc{stick}\"
}
Percussion example

A short example taken from Stravinsky’s *L’Histoire du soldat*.

```lilypond
#(define mydrums '((bassdrum default #f 4)
  (snare default #f -4)
  (tambourine default #f 0)))

global = {
\time 3/8 s4.
\time 2/4 s2*2
\time 3/8 s4.
\time 2/4 s2
}

drumsA = {
\context DrumVoice <<
  \global
  \drummode {
    \autoBeamOff
    \stemDown sn8 \stemUp tamb s8 |
    sn4 \stemDown sn4 |
    \stemUp tamb8 \stemDown sn8 \stemUp sn16 \stemDown sn \stemUp sn8 |
    \stemDown sn8 \stemUp tamb s8 |
    \stemUp sn4 s8 \stemUp tamb
  }
}

drumsB = {
\drummode {
  s4 bd8 s2*2 s4 bd8 s4 bd8 s8
}

\layout {
  indent = 40
  \context {
    \DrumStaff
    drumStyleTable = #(alist->hash-table mydrums)
  }
}

\score {
  \new StaffGroup <<
    \new DrumStaff \with {
      instrumentName = \markup \center-column {
        "Tambourine"
        "et"
        "caisse claire s. timbre"
      }
    }
  }
```
Printing music with different time signatures

In the following snippet, two parts have a completely different time signature, yet remain synchronized.

The bar lines can no longer be printed at the Score level; to allow independent bar lines in each part, the Timing_translator is moved from the Score context to the Staff context.

If bar numbers are required, the Bar_number_engraver should also be moved, since it relies on properties set by the Timing_translator; a \with block can be used to add bar numbers to the relevant staff.

```
paper {  
  indent = #0  
  ragged-right = ##t  
}

global = { \time 3/4 { s2.*3 } \bar "" \break { s2.*3 } }

\layout {  
  \context {  
    \Score  
    \remove "Timing_translator"  
    \remove "Bar_number_engraver"  
    \override SpacingSpanner.uniform-stretching = ##t  
    \override SpacingSpanner.strict-note-spacing = ##t  
    proportionalNotationDuration = #(ly:make-moment 1/64)  
  }  
  \context {  
    \Staff  
    \consists "Timing_translator"  
  }  
  \context {  
    \Voice  
    \remove "Forbid_line_break_engraver"  
    tupletFullLength = ##t  
  }  
}  
```
Bassklarinette = \new Staff \with { 
  \consists "Bar_number_engraver"
  \barNumberVisibility = #(every-nth-bar-number-visible 2)
  \override BarNumber.break-visibility = #end-of-line-invisible
} <<
  \global {
  \bar "|
  \clef treble
  \time 3/8
d''4.

  \bar "|
  \time 3/4
r8 des''2(r c''8)

  \bar "|
  \time 7/8
r4. ees''2 ~

  \bar "|
  \time 2/4
\tupletUp
\tuplet 3/2 { ees''4 r4 d''4 ~ }

  \bar "|
  \time 3/8
\tupletUp
\tuplet 4/3 { d''4 r4 }

  \bar "|
  \time 2/4
e''2

  \bar "|
  \time 3/8
es''4.

  \bar "|
  \time 3/4
r8 d''2 r8
  \bar "|
} >>

Perkussion = \new StaffGroup <<
  \new Staff <<
  \global {
  \bar "|
  \clef percussion
  \time 3/4
r4 c'2 ~
\bar "|" c'2.
\bar "|" R2.
\bar "|" r2 g'4 ~
\bar "|" g'2. ~
\bar "|" g'2.
}
\new Staff <<
\global {
\bar "|"
\clef percussion
\time 3/4
R2.
\bar "|"
g'2. ~
\bar "|"
g'2.
\bar "|"
r4 g'2 ~
\bar "|"
g'2 r4
\bar "|"
g'2.
}
>>
>>

\score {
  <<
  \Bassklarinette
  \Perkussion
  >>
}
Tam-tam example

A tam-tam example, entered with 'tt'

#(define mydrums '((tamtam default #f 0)))

\new DrumStaff \with { instrumentName = #"Tamtam" }

\drummode { 
  \set DrumStaff.drumStyleTable = #(alist->hash-table mydrums) 
  \override Staff.StaffSymbol.line-positions = #( 0 ) 
  \override Staff.BarLine.bar-extent = #(-1.5 . 1.5)

  tt 1 \pp \laissezVibrer
}

Tamtam

\pp
Tambourine example

A tambourine example, entered 'tamb'
\paper{ tagline = ##f }

#(define mydrums '((tambourine default #f 0)))

\new DrumStaff \with { instrumentName = #'"Tambourine" }

\drummode {
  \set DrumStaff.drumStyleTable = #(alist->hash-table mydrums)
  \override Staff.StaffSymbol.line-positions = #(0)
  \override Staff.BarLine.bar-extent = #(-1.5 . 1.5)

  \time 6/8
  tamb8. 16 8 8 8 8 |
  tamb4. 8 8 8 |
  % the trick with the scaled duration and the shorter rest
  % is neccessary for the correct ending of the trill-span!
  tamb2.*5/6 \startTrillSpan s8 \stopTrillSpan |}

\score { 
  \newStaff { \Tambourine {360} 6/8
  \tamb8. 16 8 8 8 8 |
  \tamb4. 8 8 8 |
  \tamb2.*5/6 \startTrillSpan s8 \stopTrillSpan |
}
Fretted strings

Sezione “Fretted string instruments” in Guida alla Notazione

Adding fingerings to a score

Fingering instructions can be entered using a simple syntax.

```
\relative c'' {
   c4-1 d-2 f-4 e-3
}
```

Adding fingerings to tablatures

To add fingerings to tablatures, use a combination of `\markup` and `\finger`.

```latex
\new TabStaff {
   \tabFullNotation
   \stemUp
   e8`\one b\2 <g\3 e'\1^-\2>[ b\2 e\4]
   <a\3 fis'\1^-\2>-^\threeTwo[ b\2 e\4]
}
```
Adding markups in a tablature

By default markups does not show in a tablature.

To make them appear, simply use the command \revert TabStaff.TextScript.stencil

\% http://lsr.di.unimi.it/LSR/Item?id=919
\% by P.P.Schneider on June 2014

\begin{verbatim}
high = { r4 r8 <g c'> q r8 r4 }

low = { c4 r4 c8 r8 g,8 b, }

pulse = { s8"1" s""& s""2" s""& s""3" s""& s""4" s""& }

\score {
  \new TabStaff {
    \repeat unfold 2 << \high \ \low \ \pulse >>
  }
  \context {
    \TabStaff
    \clef moderntab
    \revert TextScript.stencil
    \override TextScript.font-series = #'bold
    \override TextScript.font-size = #-2
    \override TextScript.color = #red
  }
  \context {
    \Score
    proportionalNotationDuration = #(ly:make-moment 1/8)
  }
}
\end{verbatim}

\begin{verbatim}
1 & 2 & 3 & 4 & 1 & 2 & 3 & 4 &
T|---0---|---0---
A|3 3 2 3 3 3 3 3 |
B|3 3 2 3 3 3 3 3 |
\end{verbatim}

Far sì che la diteggiatura appaia dentro il rigo

Per impostazione predefinita, le diteggiature orientate verticalmente sono poste fuori dal rigo; questo comportamento tuttavia può essere disabilitato. Occorre fare attenzione alle situazioni in cui le diteggiature e i gambi sono rivolti nella stessa direzione: normalmente le diteggiature evitano soltanto i gambi con travature. Questa impostazione predefinita può essere cambiata in modo da evitare tutti i gambi oppure nessuno. L’esempio seguente mostra queste due opzioni, così come tornare al comportamento predefinito.

\begin{verbatim}
\relative c' {
  <c-1 e-2 g-3 b-5>2
  \override Fingering.staff-padding = #'()
  <c-1 e-2 g-3 b-5>4 g'-0
  a8[-1 b]2 g-0 r
\end{verbatim}
Automatic Fretboards Barre

When automatic fretboards are used, barre indicators will be drawn whenever one finger is responsible for multiple strings.

If no finger indications are given in the chord from which the automatic fretboard is created, no barre indicators will be included, because there is no way to identify where barres should be placed.

\new FretBoards {
  \chordmode {
    \override FretBoard.fret-diagram-details.orientation = #'landscape
    \override FretBoard.fret-diagram-details.orientation = #'opposing-landscape
  }
}

Cambiare l’orientamento della tastiera

I diagrammi dei tasti possono essere orientati in tre modi.

\include "predefined-guitar-fretboards.ly"

<>
\chords {
  c1
  c1
  c1
}
\new FretBoards {
  \chordmode {
    c1
    \override FretBoard.fret-diagram-details.orientation = #'landscape
    c1
    \override FretBoard.fret-diagram-details.orientation = #'opposing-landscape
    c1
  }
}
Glissando di accordi in intavolatura

I glissati (o slide) di accordi vengono indicati sia nel rigo (contesto Staff) sia nell’intavolatura (contesto TabStaff). I numeri di corda sono necessari per TabStaff, perché i calcoli automatici della corda sono diversi per gli accordi e per le note singole.

myMusic = \relative c' { 
   <c e g>\glissando <f a c>
}

score { 
  \new Staff { 
    \clef "treble_8" 
    myMusic 
  } 
  \new TabStaff \myMusic 
} 

score { 
  \new Staff { 
    \clef "treble_8" 
    myMusic 
  } 
  \new TabStaff \with { \override Glissando.style = #'none } { 
    myMusic 
  } 
}
Cambi di accordo nei diagrammi dei tasti

Si può impostare il contesto FretBoards in modo che mostri il diagramma solo quando l'accordo cambia o all'inizio di una nuova linea.

\include "predefined-guitar-fretboards.ly"

\myChords = \chordmode {
    c1 c1 \break
    \set chordChanges = \#t
    c1 c1 \break
    c1 c1
}

<<
\new ChordNames { \myChords }
\new FretBoards { \myChords }
\new Staff { \myChords }
>>
Chords with stretched fingering for FretBoards and TabVoice

Sometimes chords with a stretched fingering are required. If not otherwise specified the context-property maximumFretStretch is set to 4, though. Resulting in a warning about "No string for pitch ..." and the note is omitted. You may set maximumFretStretch to an appropriate value or explicitly assign string-numbers to all notes of a chord.

%% The code below will print two warnings, which may be omitted by uncommenting:
%%(for-each (lambda (x) (ly:expect-warning "No string for pitch")) (iota 2))

\begin{verbatim}
\mus = {
  \<c' bes'>
  \<c'2 bes'>
  \set maximumFretStretch = 5
  \<c' bes'>
  \<c'2 bes'1>
}

<<
  \new FretBoards \mus
  \new TabVoice \mus
>>
\end{verbatim}

Controllare il posizionamento delle diteggiature di un accordo

Il posizionamento dei numeri della diteggiatura può essere regolato in modo preciso. Perché l’orientamento funzioni, occorre usare il costrutto per gli accordi <> anche per le note singole. Si può impostare in modo simile l’orientamento dei numeri di corda e delle diteggiature della mano destra.

\begin{verbatim}
\relative c' {
  \set fingeringOrientations = #'(left)
  \<c-1 e-3 a-5>4
  \set fingeringOrientations = #'(down)
  \<c-1 e-3 a-5>4
  \set fingeringOrientations = #'(down right up)
  \<c-1 e-3 a-5>4
  \set fingeringOrientations = #'(up)
  \<c-1 e-3 a-5>4
  \set fingeringOrientations = #'(left)
  \<c-1>2
  \set fingeringOrientations = #'(down)
  \<e-3>2
  \set stringNumberOrientations = #'(up left down)
  \<f\3 a\2 c\1>1
  \set strokeFingerOrientations = #'(down right up)
  \<c\rightHandFinger #1 e\rightHandFinger #2 c\rightHandFinger #4 >
}\end{verbatim}
Personalizzare la tastiera del diagramma dei tasti


\include "predefined-guitar-fretboards.ly"
\storePredefinedDiagram #default-fret-table \chordmode { c' }
\n#guitar-tuning
\n"x;1-1-(;3-2;3-3;3-4;1-1-);"

% shorthand
\oo = #(define-music-function
\n  (grob-path value)
\n  (list? scheme?)
\n  #{ \once \override $grob-path = #value #})

<<
\new ChordNames {
  \chordmode { c1 | c | c | d }
}
\new FretBoards {
  % Set global properties of fret diagram
  \override FretBoards.FretBoard.size = #'1.2
  \override FretBoard.fret-diagram-details.finger-code = #'in-dot
  \override FretBoard.fret-diagram-details.dot-color = #'white
  \chordmode {
    c
    \oo FretBoard.size #1.0
    \oo FretBoard.fret-diagram-details.barre-type #'straight
    \oo FretBoard.fret-diagram-details.dot-color #'black
    \oo FretBoard.fret-diagram-details.finger-code #'below-string
    c
    \oo FretBoard.fret-diagram-details.barre-type #'none
    \oo FretBoard.fret-diagram-details.number-type #'arabic
    \oo FretBoard.fret-diagram-details.orientation #'landscape
    \oo FretBoard.fret-diagram-details.mute-string #'M
    \oo FretBoard.fret-diagram-details.label-dir #LEFT
    \oo FretBoard.fret-diagram-details.dot-color #'black
    c
    \oo FretBoard.fret-diagram-details.finger-code #'below-string
    \oo FretBoard.fret-diagram-details.dot-radius #0.35
    \oo FretBoard.fret-diagram-details.dot-position #0.5
Personalizzare il diagramma dei tasti di tipo markup

Le proprietà del diagramma dei tasti si possono impostare tramite `fret-diagram-details`. Per diagrammi di tipo markup, gli override possono essere applicati all’oggetto `Voice.TextScript` o direttamente al markup.

\chords { c1 | c | c | d }
\new Voice = "mel" {
\textLengthOn
% Set global properties of fret diagram
\override TextScript.size = #'1.2
\override TextScript.fret-diagram-details.finger-code = #'in-dot
\override TextScript.fret-diagram-details.dot-color = #'white

%% C major for guitar, no barre, using defaults
% terse style
\markup { \fret-diagram-terse "x;3-3;2-2;o;1-1;o;" }

%% C major for guitar, barred on third fret
% verbose style
% size 1.0
% roman fret label, finger labels below string, straight barre
\markup {
% standard size
\override #'(size . 1.0) {
\override #'(fret-diagram-details . ( % (number-type . roman-lower)
(finger-code . in-dot)
(barre-type . straight))) {
\fret-diagram-verbose #'(mute 6)
(place-fret 5 3 1)
(place-fret 4 5 2)
(place-fret 3 5 3)
(place-fret 2 5 4)
(place-fret 1 3 1)
(barre 5 1 3))

%%%% C major for guitar, barred on third fret
%%%% verbose style
%%%% landscape orientation, arabic numbers, M for mute string
%%%% no barre, fret label down or left, small mute label font

c'\markup {
\override #'(fret-diagram-details . (  
  (finger-code . below-string)  
  (number-type . arabic)  
  (label-dir . -1)  
  (mute-string . "M")  
  (orientation . landscape)  
  (barre-type . none)  
  (xo-font-magnification . 0.4)  
  (xo-padding . 0.3))) {  
  \fret-diagram-verbose #(  
    (mute 6)  
    (place-fret 5 3 1)  
    (place-fret 4 5 2)  
    (place-fret 3 5 3)  
    (place-fret 2 5 4)  
    (place-fret 1 3 1)  
    (barre 5 1 3))  
}
}

%%%% simple D chord
%%%% terse style
%%%% larger dots, centered dots, fewer frets
%%%% label below string
d'\markup {
\override #'(fret-diagram-details . (  
  (finger-code . below-string)  
  (dot-radius . 0.35)  
  (dot-position . 0.5)  
  (fret-count . 3))) {  
  \fret-diagram-terse "x;x;o;2-1;3-2;2-3;"  
}
}
}

>>
Definire diagrammi dei tasti predefiniti per altri strumenti

Si possono aggiungere diagrammi dei tasti predefiniti per nuovi strumenti oltre a quelli standard per chitarra. Questo frammento mostra come farlo definendo una nuova accordatura e alcuni diagrammi predefiniti per il cuatro venezuelano.

Mostra anche come includere le diteggiature negli accordi usati come punti di riferimento per la consultazione degli accordi e come mostrarle nel diagramma dei tasti e in TabStaff, ma non nella musica.

Questi diagrammi non sono trasponibili perché contengono informazioni sulle corde. È prevista una correzione in futuro.

% add FretBoards for the Cuatro
% Note: This section could be put into a separate file
% predefined-cuatro-fretboards.ly
% and included into each of your compositions

cuartoTuning = #`(,(ly:make-pitch 0 6 0)
    ,(ly:make-pitch 1 3 SHARP)
    ,(ly:make-pitch 1 1 0)
    ,(ly:make-pitch 0 5 0))

dSix = { <a\4 b\1 d\3 fis\2> }
dMajor = { <a\4 d\1 d\3 fis \2> }
amajSeven = { <a\4 cis\1 e\3 g\2> }
dmajSeven = { <a\4 c\1 d\3 fis\2> }
gMajor = { <b\4 b\1 d\3 g\2> }

\storePredefinedDiagram #default-fret-table \dSix #cuatroTuning
    #"o;o;o;o;"
\storePredefinedDiagram #default-fret-table \dMajor #cuatroTuning
    #"o;o;o;3-3;"
\storePredefinedDiagram #default-fret-table \amajSeven #cuatroTuning
    #"o;2-2;1-1;2-3;"
\storePredefinedDiagram #default-fret-table \dmajSeven #cuatroTuning
    #"o;o;o;1-1;"
\storePredefinedDiagram #default-fret-table \gMajor #cuatroTuning
    #"2-2;o;1-1;o;"

% end of potential include file /predefined-cuatro-fretboards.ly
\begin{verbatim}
\%(set-global-staff-size 16)

primerosNames = \chordmode {
  d:6 d a:maj7 d:maj7
  g
}
primeros = {
  \dSix \dMajor \aMajSeven \dMajSeven
  \gMajor
}

score {
  <<
    \new ChordNames {
      \set chordChanges = ##t
      \primerosNames
    }

    \new Staff {
      \new Voice \with {
        \remove "New_fingering_engraver"
      }
      \relative c'' {
        \primeros
      }
    }

    \new FretBoards {
      \set Staff.stringTunings = #cuatroTuning
      \override FretBoard
      \override FretBoard.fret-diagram-details.string-count = 4
      \override FretBoard.fret-diagram-details.finger-code = #'in-dot
      \primeros
    }

    \new TabStaff \relative c'' {
      \set TabStaff.stringTunings = #cuatroTuning
      \primeros
    }
  }

  >>

  \layout {
    \context {
      \Score
      \override SpacingSpanner.base-shortest-duration = 
      #(ly:make-moment 1 16)
    }
    \midi { }
Faking a hammer in tablatures

A hammer in tablature can be faked with slurs.

```latex
\score { 
  \new TabStaff { 
    \relative c'' { 
      \tabFullNotation 
      c4( d) d( d) 
      d2( c) 
    } 
  } 
}
```

Diteggiature, indicazioni di corda e diteggiature della mano destra

Questo esempio combina la diteggiatura per la mano sinistra, le indicazioni di stringa e la diteggiatura della mano destra.

```latex
#(define RH rightHandFinger)
\relative c { 
  \clef "treble_8" 
  <c-3\5\RH #1 >4 
  <e-2\4\RH #2 >4 
  <g-0\3\RH #3 >4 
  <c-1\2\RH #4 >4 
}
```

Flamenco notation

For flamenco guitar, special notation is used:

- a \textit{golpe} symbol to indicate a slap on the guitar body with the nail of the ring finger
• an arrow to indicate (the direction of) strokes
• different letters for fingering (‘p’: thumb, “i”: index finger, “m”: middle finger, “a”: ring finger and “x”: little finger)
• 3- and 4-finger rasgueados; stroke upwards with all fingers, ending with an up- and down using the index finger
• abanicos: strokes (in tuples) with thumb (down), little and index finger (both up). There’s also an abanico 2 where middle and ring finger are used instead of the little finger.
• alza pua: fast playing with the thumb

Most figures use arrows in combination with fingering; with abanicos and rasgueados, note-heads are printed only for the first chord.

This snippet contains some header-like code that can be copied as ‘flamenco.ly’ and included in source files.

%%%%%%% Cut here ----- Start ‘flamenco.ly’

% Text indicators :
abanico = ~\markup\small { \italic Abanico }
rasgueado = ~\markup\small { \italic Ras. }
alzapua = ~\markup\small { \italic Alzapua }

% Finger stroke symbols :
strokeUp = \markup\combine\override #'(thickness . 1.3) \draw-line #'(0 . 2)\raise #2 \arrow-head #Y #UP ##f
strokeDown = \markup\combine\arrow-head #Y #DOWN ##f \override #'(thickness . 1.3) \draw-line #'(0 . 2)

% Golpe symbol :
golpe = \markup { \filled-box #'(0 . 1) #'(0 . 1) #0
\hspace #-.16
\with-color #white
\filled-box #'(0.15 . 0.85) #'(0.15 . 0.85) #0 }

% Strokes, fingers and golpe command :
RHp = \rightHandFinger #1
RHi = \rightHandFinger #2
RHm = \rightHandFinger #3
RHa = \rightHandFinger #4
RHx = \rightHandFinger #5
RHu = \rightHandFinger \strokeUp
RHz = \rightHandFinger \strokeDown
RHg = \rightHandFinger \golpe

% Just handy :) 
tupletOff = {
\once \omit TupletNumber
\once \omit TupletBracket
}

tupletsOff = {


\omit TupletNumber
\override TupletBracket.bracket-visibility = #'if-no-beam
}

tupletsOn = {
  \override TupletBracket.bracket-visibility = #'default
  \undo \omit TupletNumber
}

headsOff = {
  \hide TabNoteHead
  \hide NoteHead
  \override NoteHead.no-ledgers = ##t
}

headsOn = {
  \override TabNoteHead.transparent = ##f
  \override NoteHead.transparent = ##f
  \override NoteHead.no-ledgers = ##f
}

%%%%%%% Cut here ----- End 'flamenco.ly'

part = \relative c' {
  \set strokeFingerOrientations = #'(up)
  \key a \major
  <a, e' a cis e\RHu\RH1>8
  <a e' a cis e\RHd\RH1>8
  r4
  r2^\markup\golpe
  <a e' a cis e\RHu\RH1>8
  <a e' a cis e\RHd\RH1>8
  <a e' a cis e\RHu\RH1\RHg>8
  <a e' a cis e\RHd\RH1>8
  r2
  <a e' a cis e\RHu\RHa>16\rasgueado
  \headsOff
  <a e' a cis e\RHu\RHa>
  <a e' a cis e\RHu\RHm>
  <a e' a cis e\RHd\RH1>~
  \headsOn
  <a e' a cis e>2
  r4

  \tupletOff
  \tuplet 5/4 {
    <a e' a cis e\RHu\RHx>16\rasgueado
    \headsOff
    <a e' a cis e\RHu\RHa>
    <a e' a cis e\RHu\RHm>
    <a e' a cis e\RHu\RH1>
    <a e' a cis e\RHd\RH1>~
\headsOn
}
\tupletsOff
\tuplet 3/2 {
  <a e' a cis e>2
  r\^4
}\tupletsOff
\tuplet 3/2 {
  <a e' a cis e>RH\^d\RHp>8\abanico
  \headsOff
  <a e' a cis e>\RHu\RHx>
  <a e' a cis e>\RHu\RHi>
  \headsOn
}
\tuplet 3/2 {
  <a e' a cis e>\RH\^d\RHp>8
  \headsOff
  <a e' a cis e>\RHu\RHx>
  <a e' a cis e>\RHu\RHi>
  \headsOn
}
\tuplet 3/2 {
  <a e' a cis e>\RH\^d\RHp>8
  \headsOff
  <a e' a cis e>\RHu\RHx>
  <a e' a cis e>\RHu\RHi>
  \headsOn
}
\tuplet 3/2 {
  <a e' a cis e>\RH\^d\RHp>8
  \headsOff
  <a e' a cis e>\RHu\RHx>
  <a e' a cis e>\RHu\RHi>
  \headsOn
}
\tupletsOff
\override Beam.positions = #'(2 . 2)
\tuplet 3/2 {
  a\^8\RHp\alzapua
  <e' a>\RHu\RHg>
  <e a>\RHd>
}
\tuplet 3/2 {
  a,\^8\RHp
  <e' a>\RHu\RHg>
  <e a>\RHd>
}
\tuplet 3/2 {
  a,\^8\RHp
  <e' a>\RHu\RHg>
  <e a>\RHd>
}
\tuplet 3/2 {
  a,\^8\RHp

\begin{verbatim}
<e' a\Rh\Rh>  
<e a\Rhd>
}
\tupletsOn
<a, e' a\Rh\Rh>1
\bar "|."
}
\score {
\new StaffGroup <<
\context Staff = "part" <<
\clef "G_8"
{
  \part
}>>
\context TabStaff {
  \part
}>>
\layout {
  \ranged-right = ###t
}
}

Fret diagrams explained and developed
This snippet shows many possibilities for obtaining and tweaking fret diagrams.
\end{verbatim}
\chords {  
a2 a
\repeat unfold 3 {  
c c c d d
}
}

\new Voice = "mel" {  
\textLengthOn  
% Set global properties of fret diagram  
\override TextScript.size = #.1.2
\override TextScript.fret-diagram-details.finger-code = #'below-string
\override TextScript.fret-diagram-details.dot-color = #'black

%%% A chord for ukulele  
a'2^\markup {  
  \override #'(fret-diagram-details . (  
               (string-count . 4)  
               (dot-color . white)  
               (finger-code . in-dot))) {  
    \fret-diagram "4-2-2;3-1-1;2-o;1-o;"
  }
}

%%% A chord for ukulele, with formatting defined in definition string  
% 1.2 * size, 4 strings, 4 frets, fingerings below string  
% dot radius .35 of fret spacing, dot position 0.55 of fret spacing  
a'2^\markup {  
  \override #'(fret-diagram-details . (  
               (dot-color . white)  
               (open-string . "o"))) {  
    \fret-diagram "s:1.2;w:4;h:3;f:2;d:0.35;p:0.55;4-2-2;3-1-1;2-o;1-o;"
  }
}

%%% These chords will be in normal orientation

%%% C major for guitar, barred on third fret  
% verbose style  
% roman fret label, finger labels below string, straight barre  
c'2^\markup {  
  % 110% of default size  
  \override #(size . 1.1) {  
    \override #'(fret-diagram-details . (  
               (number-type . roman-lower)  
               (finger-code . below-string)  
               (barre-type . straight))) {  
      \fret-diagram-verbose #'((mute 6)  
         (place-fret 5 3 1)  
         (place-fret 4 5 2)  
         (place-fret 3 5 3)  
         (place-fret 2 5 4)
%% C major for guitar, barred on third fret
%% Double barre used to test barre function
% verbose style
\C major for guitar, with capo on third fret
% verbose style
\C major for guitar, with capo on third fret
% verbose style
\C major for guitar, with capo on third fret

\ OVERRIDE #'(fret-diagram-details . (  
  (fingercode . below-string)  
  (dot-radius . 0.35)  
  (string-thickness-factor . 0.3)  
  (dot-position . 0.5)  
  (fret-count . 3))) {  
  \fret-diagram-terse "x;x;o;2-1;3-2;2-3;"  
}  
\ OVERRIDE #'(fret-diagram-details . (  
  (fingercode . below-string)  
  (dot-radius . 0.35)  
  (dot-position . 0.5)  
  (top-fret-thickness . 7)  
  (fret-count . 3))) {  
  \fret-diagram-terse "x;x;o;2-1;3-2;2-3;"  
}  
\ OVERRIDE TextScript.fret-diagram-details.orientation = #'landscape

%%%% C major for guitar, barred on third fret
% verbose style
% roman fret label, finger labels below string, straight barre
c'2\markup {
  % 110% of default size
 \ OVERRIDE #'(size . 1.1) {  
    \ OVERRIDE #'(fret-diagram-details . (  
       (number-type . roman-lower)  
       (fingercode . below-string)  
       (barre-type . straight))) {  
      \fret-diagram-verbose #'(mute 6)  
       (place-fret 5 3 1)  
       (place-fret 4 5 2)  
       (place-fret 3 5 3)  
       (place-fret 2 5 4)  
       (place-fret 1 3 1)  
       (barre 5 1 3))  
    }  
}  
}  
}  
}  

(\open 1))
%% C major for guitar, barred on third fret
%% Double barre used to test barre function
% verbose style
\markup {
\override \(size\) = \(1.1\) {
\override \(fret-diagram-details\) = \(\) {
(fret-label-font-mag . 0.6)
(fret-label-vertical offset . 0)
(label-dir . -1)
(mute-string . "M")
(xo-font-magnification . 0.4)
(xo-padding . 0.3))) {
\fret-diagram-verbose #'(mute 6)
(place-fret 5 3 1)
(place-fret 4 5 2)
(place-fret 3 5 3)
(place-fret 2 5 4)
(place-fret 1 3 1)
(barre 4 2 5)
(barre 5 1 3))
}
}

%% C major for guitar, with capo on third fret
% verbose style
\markup {
\override \(size\) = \(1.1\) {
\override \(fret-diagram-details\) = \(\) {
(fret-label-font-mag . 0.9)
(finger-code . in-dot)
(fret-label-font-mag . 0.6)
(fret-label-vertical offset . 0)
(label-dir . -1)
(capo 3)
(open 5)
(reply \(5\))\(\(5\))
(place-fret 4 5 1)
(place-fret 3 5 2)
(place-fret 2 5 3)
(open 1))
}
}
}
%% simple D chord
d'2^\markup{\override \(\text{fret-diagram-details} . (\)
(finger-code . below-string)
(dot-radius . 0.35)
(dot-position . 0.5)
(fret-count . 3))) { \fret-diagram-terse "x;x;o;2-1;3-2;2-3;" }
}

%% simple D chord, large top fret thickness
d'2^\markup{\override \(\text{fret-diagram-details} . (\)
(finger-code . below-string)
(dot-radius . 0.35)
(dot-position . 0.5)
(top-fret-thickness . 7)
(fret-count . 3))) { \fret-diagram-terse "x;x;o;2-1;3-2;2-3;" }
}

% These chords will be in opposing-landscape orientation
\override TextScript.fret-diagram-details.orientation = #'opposing-landscape

%% C major for guitar, barred on third fret
% verbose style
% roman fret label, finger labels below string, straight barre
c'2^\markup{% 110\% of default size
\override \(\text{size} . 1.1) { \override \(\text{fret-diagram-details} . (\)
(number-type . roman-lower)
(finger-code . below-string)
(barre-type . straight))) { \fret-diagram-verbose #'((mute 6)
(place-fret 5 3 1)
(place-fret 4 5 2)
(place-fret 3 5 3)
(place-fret 2 5 4)
(place-fret 1 3 1)
(barre 5 1 3))
}
}

%% C major for guitar, barred on third fret
% Double barre used to test barre function
% verbose style
c'2^\markup{% 110\% of default size
\override #'(size . 1.1) {
  \override #'(fret-diagram-details . ( 
    (number-type . arabic)
    (dot-label-font-mag . 0.9)
    (fing[er-code . in-dot)
    (fret-label-font-mag . 0.6)
    (fret-label-vertical-offset . 0)
    (label-dir . -1)
    (mute-string . "M")
    (xo-font-magnification . 0.4)
    (xo-padding . 0.3))
  )

  \fret-diagram-verbose #'( (mute 6)
    (place-fret 5 3 1)
    (place-fret 4 5 2)
    (place-fret 3 5 3)
    (place-fret 2 5 4)
    (place-fret 1 3 1)
    (barre 4 2 5)
    (barre 5 1 3))

}

%% C major for guitar, with capo on third fret
% verbose style
c'2^\markup {
  % 110% of default size
  \override #'(size . 1.1) {
    \override #'(fret-diagram-details . ( 
      (number-type . roman-upper)
      (dot-label-font-mag . 0.9)
      (fing[er-code . none)
      (fret-label-vertical-offset . 0.5)
      (xo-font-magnification . 0.4)
      (xo-padding . 0.3))

    \fret-diagram-verbose #'( (mute 6)
      (capo 3)
      (open 5)
      (place-fret 4 5 1)
      (place-fret 3 5 2)
      (place-fret 2 5 3)
      (open 1))


  }
}

%% simple D chord
d'2^\markup {
  \override #'(fret-diagram-details . ( 
    (finger-code . below-string)
    (dot-radius . 0.35)
    (dot-position . 0.5)
Tabelle alternative per i diagrammi dei tasti

Si possono creare tabelle alternative per i diagrammi dei tasti. Queste possono essere usate per avere diagrammi alternativi per uno stesso accordo.

Per poter usare una tabella alternativa, deve prima essere creata. Quindi si aggiungono i diagrammi alla tabella.

La tabella può essere vuota oppure ricopiata da una tabella esistente. La tabella da usare nel mostrare i diagrammi predefiniti viene selezionata dalla proprietà \predefinedDiagramTable.

\include "predefined-guitar-fretboards.ly"

\% Make a blank new fretboard table
\%(define custom-fretboard-table-one
  (make-fretboard-table))

\% Make a new fretboard table as a copy of default-fret-table
\%(define custom-fretboard-table-two
(make-fretboard-table default-fret-table))

% Add a chord to custom-fretboard-table-one
\storePredefinedDiagram #custom-fretboard-table-one
  \chordmode {c}
  #guitar-tuning
  "3-(;3;5;5;5;3-);"

% Add a chord to custom-fretboard-table-two
\storePredefinedDiagram #custom-fretboard-table-two
  \chordmode {c}
  #guitar-tuning
  "x;3;5;5;5;o;"

<<
\chords {
  c1 | d1 |
  c1 | d1 |
  c1 | d1 |
}
\new FretBoards {
  \chordmode {
        \set predefinedDiagramTable = #default-fret-table
        c1 | d1 |
        \set predefinedDiagramTable = #custom-fretboard-table-one
        c1 | d1 |
        \set predefinedDiagramTable = #custom-fretboard-table-two
        c1 | d1 |
  }
}
\new Staff {
  \clef "treble_8"
  <<
  \chordmode {
    c1 | d1 |
    c1 | d1 |
    c1 | d1 |
  }
  {
    s1 \markup "Default table" | s1 |
    s1 \markup \column {"New table" "from empty"} | s1 |
    s1 \markup \column {"New table" "from default"} | s1 |
  }
  >>
}
Armonici su corde premute in intavolatura

Questo frammento mostra come scrivere su intavolatura armonici su corde premute.

pinchedHarmonics = {
  \textSpannerDown
  \override TextSpanner.bound-details.left.text = \markup { \halign #-0.5 \text "PH" }
  \override TextSpanner.style = #'dashed-line
  \override TextSpanner.dash-period = #0.6
  \override TextSpanner.bound-details.right.attach-dir = #1
  \override TextSpanner.bound-details.right.text = \markup { \draw-line #'(0 . 1) }
  \override TextSpanner.bound-details.right.padding = #-0.5
}

harmonics = {
  %artificial harmonics (AH)
  \textLengthOn
  <\parenthesize b b'\harmonic>4 \markup { \text "AH 16" }
  <\parenthesize g g'\harmonic>4 \markup { \text "AH 17" }
  <\parenthesize d' d''\harmonic>2 \markup { \text "AH 19" }
  %pinched harmonics (PH)
  \pinchedHarmonics
  a'\harmonic startTextSpan
  d'\harmonic stopTextSpan
  %tapped harmonics (TH)
  <\parenthesize g \g'\harmonic>4 \markup { \text "TH 17" }
  <\parenthesize a \a'\harmonic>4 \markup { \text "TH 19" }
  <\parenthesize c' \c''\harmonic>2 \markup { \text "TH 17" }
  %touch harmonics (TCH)
  a4( <e'\harmonic>2. )_\markup { \text "TCH" }
}

frettedStrings = {
  %artificial harmonics (AH)
  \harmonicByFret #4 g4\3
  \harmonicByFret #5 d4\4
  \harmonicByFret #7 g2\3
  %pinched harmonics (PH)
  \harmonicByFret #7 d2\4
}
Guitar slides

Unlike glissandos, slides may go from an imprecise point of the fretboard to a specific fret. A good way to do this is to add a hidden grace note before the note which is actually played, as demonstrated in the following example.

%%% Hide fret number: useful to draw slide into/from a casual point of  
%%% the fretboard.
hideFretNumber = {
  \once \hide TabNoteHead
  \once \hide NoteHead
  \once \hide Stem
  \once \override NoteHead.no-ledgers = ##t
  \once \override Glissando.bound-details.left.padding = #0.3
}

music= \relative c' {
Ritmi di accompagnamento per chitarra

Per la musica per chitarra, è possibile mostrare i ritmi di accompagnamento, insieme alle note della melodia e ai nomi e ai diagrammi degli accordi.

\include "predefined-guitar-fretboards.ly"

\new ChordNames {
  \chordmode {
    c1 | f | g | c
  }
}

\new FretBoards {
  \chordmode {
    c1 | f | g | c
  }
}

\new Voice \with {
  \consists "Pitch_squash_ engraver"
} {
  \relative c'' {
    \improvisationOn
    c4 c8 c c4 c8 c
  }
}
Martellato e strappato

Il martellato (hammer on) e lo strappato (pull off) si possono ottenere con le legature di portamento.

\new TabStaff {
\relative c' {
    d4( e\2)
    a( g)
}
}

Martellato e strappato usando accordi

Quando il martellato o lo strappato si applicano a delle note in un accordo, viene disegnato un solo arco. Ma è possibile avere un “doppio arco” impostando la proprietà doubleSlurs su #t.
Martellato e strappato usando le voci

L’arco del martellato o dello strappato è rivolto in alto nella prima e terza voce, mentre è rivolto in basso nella seconda e quarta voce.

How to change fret diagram position

If you want to move the position of a fret diagram, for example, to avoid collision, or to place it between two notes, you have various possibilities:

1) modify #'padding or #'extra-offset values (as shown in the first snippet)
2) you can add an invisible voice and attach the fret diagrams to the invisible notes in that voice (as shown in the second example).

If you need to move the fret according with a rhythmic position inside the bar (in the example, the third beat of the measure) the second example is better, because the fret is aligned with the third beat itself.

harmonies = \chordmode
{
  a8:13
  % THE FOLLOWING IS THE COMMAND TO MOVE THE CHORD NAME
  \once \override ChordNames.ChordName.extra-offset = #'(10 . 0)
  b8:13 s2.
  % THIS LINE IS THE SECOND METHOD
  s4 s4 b4:13
}

\score
{

new ChordNames \harmonies
new Staff
\markup { \fret-diagram "6-x;5-0;4-2;3-0;2-0;1-2;" }
\override TextScript.extra-offset = #'(10 . 0)
b4.-\markup { \fret-diagram "6-x;5-2;4-4;3-2;2-2;1-4;" } b4. a8\break
\markup { \fret-diagram "6-x;5-2;4-4;3-2;2-2;1-4;" } b4. a8\break
\markup { \fret-diagram "6-x;5-2;4-4;3-2;2-2;1-4;" } b4. a8\break
<<
A^9_13
\column { "LilyPond example file by Amelie Zapf," 
"Berlin 07/07/2003"
}
>>
>>

Modello per combo jazz
Ecco un modello piuttosto complesso, per un gruppo jazz. Si noti che tutti gli strumenti sono in \key c \major. Si tratta della tonalità reale; sarà trasposta automaticamente includendo la musica all'interno di una sezione \transpose.

\header {
  title = "Song"
  subtitle = "(tune)"
  composer = "Me"
  meter = "moderato"
  piece = "Swing"
  tagline = \markup {
    \column {
      "LilyPond example file by Amelie Zapf,"
      "Berlin 07/07/2003"
    }
  }
}

% To make the example display in the documentation
\paper {
\paper-width = 130
}
\set-global-staff-size 16
\include "english.ly"

%----------------------------------------------------------
% Some macros %-------------------------------------------%
%----------------------------------------------------------

sl = {
  \override NoteHead.style = #'slash
  \hide Stem
}

nsl = {
  \revert NoteHead.style
  \undo \hide Stem
}

crOn = \override NoteHead.style = #'cross

crOff = \revert NoteHead.style

%% insert chord name style stuff here.

jazzChords = { }

%----------------------------------------------------------
% Keys'n'thangs %-------------------------------------------%
%----------------------------------------------------------

global = { \time 4/4 }

Key = { \key c \major }

% #---------- Horns #----------

% ------ Trumpet ------

trpt = \transpose c d \relative c' '{
  \Key
  c1 | c | c |
}

trpHarmony = \transpose c' d {
  \jazzChords
}

trumpet = {
  \global
  \clef treble
  <<
  \trpt
  >>
}

% ------ Alto Saxophone ------

alto = \transpose c a \relative c' '{
  \Key
  c1 | c | c |
}

altoHarmony = \transpose c' a {

\jazzChords
}

altoSax = {
  \global
  \clef treble
  <<
    \alto
  >>
}

% ------ Baritone Saxophone ------
bari = \transpose c a' \relative c {
  \Key
c1
c1
  \sl
d4-'Solo" d d d
  \nsl
}

bariHarmony = \transpose c' a \chordmode {
  \jazzChords s1 s d2:maj e:m7
}

bariSax = {
  \global
  \clef treble
  <<
    \bari
  >>
}

% ------ Trombone ------
tbone = \relative c {
  \Key
c1 \  c\  c
}

tboneHarmony = \chordmode {
  \jazzChords
}

trombone = {
  \global
  \clef bass
  <<
    \tbone
  >>
}

% ############ Rhythm Section ############

% ------ Guitar ------
gtr = \relative c'' {
  \Key
c1
\sl
b4 b b b
\nsl
c1
}
gtrHarmony = \chordmode {
\jazzChords
s1 c2:min7+ d2:maj9
}
guitar = {
\global
\clef treble
<<
 \gtr
>>
}

%% ------ Piano ------
rhUpper = \relative c'\prime {
 \voiceOne
 \Key
c1 | c | c
}
rhLower = \relative c' {
 \voiceTwo
 \Key
e1 | e | e
}
lhUpper = \relative c' {
 \voiceOne
 \Key
g1 | g | g
}
lhLower = \relative c {
 \voiceTwo
 \Key
c1 | c | c
}
PianoRH = {
 \clef treble
 \global
<<
 \new Voice = "one" \rhUpper
 \new Voice = "two" \rhLower
>>
}
PianoLH = {
 \clef bass
 \global
<<
\new Voice = "one\" \lhUpper
\new Voice = "two\" \lhLower

piano = {
  new Staff = "upper\" PianoRH
  new Staff = "lower\" PianoLH
};

% ------ Bass Guitar ------
Bass = \relative c {
  \Key c c c
  bass = {
    \global
    \clef bass
    Bass
  }
};

% ------ Drums ------
up = \drummode {
  \voiceOne
  hh4 <hh sn> hh <hh sn>
  hh4 <hh sn> hh <hh sn>
  hh4 <hh sn> hh <hh sn>
}

down = \drummode {
  \voiceTwo
  bd4 s bd s
  bd4 s bd s
  bd4 s bd s
}

drumContents = {
  \global
  new DrumVoice \up
  new DrumVoice \down
};

%%%%%%% It All Goes Together Here %%%%%%%%%%%%%%%%%%%%%%

\score {
  new StaffGroup = "horns"
\new Staff = "trumpet" \with { instrumentName = "Trumpet" }
\trumpet
\new Staff = "altosax" \with { instrumentName = "Alto Sax" }
\altosax
\new ChordNames = "barichords" \with { instrumentName = "Trumpet" }
\bariHarmony
\new Staff = "barisax" \with { instrumentName = "Bari Sax" }
\barisax
\new Staff = "trombone" \with { instrumentName = "Trombone" }
\trombone
>

\new StaffGroup = "rhythm" <<
\new ChordNames = "chords" \gtrHarmony
\new Staff = "guitar" \with { instrumentName = "Guitar" }
\guitar
\new PianoStaff = "piano" \with {
    instrumentName = "Piano"
    midiInstrument = "acoustic grand"
}
\piano
\new Staff = "bass" \with { instrumentName = "Bass" }
\bass
\new DrumStaff \with { instrumentName = "Drums" }
\drumContents
>>

\layout {
    \context { \Staff \RemoveEmptyStaves }
    \context {
        \Score
        \override BarNumber.padding = #3
        \override RehearsalMark.padding = #2
        skipBars = ##t
    }
    \midi { }
}

Song
(tune)

Me

moderato

Swing
Laissez vibrer ties

Laissez vibrer ties have a fixed size. Their formatting can be tuned using 'tie-configuration.

\relative c' {
  <c e g>4\laissezVibrer r <c f g>\laissezVibrer r
  <c d f g>4\laissezVibrer r <c d f g>4.\laissezVibrer r8
  \override LaissezVibrerTieColumn.tie-configuration
  = #'((-7 ,DOWN)
  (-5 ,DOWN)
  (-3 ,UP)
  (-1 ,UP))
  <c d e f>4\laissezVibrer r
}

{\times 4/4} {clefs.F /clefs.G /noteheads.s0/noteheads.s0/noteheads.s0}
Let TabStaff print the topmost string at bottom

In tablatures usually the first string is printed topmost. If you want to have it at the bottom change the stringOneTopmost-context-property. For a context-wide setting this could be done in layout as well.

```latex
\begin{verbatim}
\layout {
  \context {Score
    stringOneTopmost = ##f}
  \context {TabStaff
    tablatureFormat = #fret-letter-tablature-format}
}

m = {
  \cadenzaOn
  e, b, e gis! b e'
  \bar "||"
}

\new Staff { \clef "G_8" <>_"default" \m <>_"italian (historic)" \m }
\new TabStaff {
  \m
  \set Score.stringOneTopmost = ##f
  \set TabStaff.tablatureFormat = #fret-letter-tablature-format
  \m
}
\end{verbatim}
```

Letter tablature formatting

Tablature can be formatted using letters instead of numbers.

```latex
\begin{verbatim}
music = \relative c {
  c4 d e f
}
```

\end{verbatim}
Armonici su corde a vuoto in intavolatura

Questo frammento mostra come scrivere armonici su corde a vuoto.

openStringHarmonics = {
  \textSpannerDown
  \override TextSpanner.staff-padding = #3
  \override TextSpanner.dash-fraction = #0.3
  \override TextSpanner.dash-period = #1

  %first harmonic
  \override TextSpanner.bound-details.left.text =
    \markup\small "1st harm."
  \harmonicByFret #12 e,\6\startTextSpan
  \harmonicByRatio #1/2 e,\6\stopTextSpan

  %second harmonic
  \override TextSpanner.bound-details.left.text =
    \markup\small "2nd harm."
  \harmonicByFret #7 e,\6\startTextSpan
  \harmonicByRatio #1/3 e,\6
  \harmonicByFret #19 e,\6
  \harmonicByRatio #2/3 e,\6\stopTextSpan
  \%harmonicByFret #19 < e,\6 a,\5 d\4 >
  \%harmonicByRatio #2/3 < e,\6 a,\5 d\4 >

  %third harmonic
  \override TextSpanner.bound-details.left.text =
Posizionamento delle diteggiature della mano destra

È possibile avere un maggior controllo sul posizionamento delle diteggiature della mano destra impostando una specifica proprietà, come illustrato nell’esempio seguente.

```latex
#(define RH rightHandFinger)
\relative c { \clef "treble_8"
\set strokeFingerOrientations = #'(up down)
<\RH #1 e\RH #2 g\RH #3 c\RH #4 >4

\set strokeFingerOrientations = #'(up right down)
<\RH #1 e\RH #2 g\RH #3 c\RH #4 >4

\set strokeFingerOrientations = #'(left)
<\RH #1 e\RH #2 g\RH #3 c\RH #4 >2
```
Polifonia in intavolatura

La polifonia in TabStaff funziona proprio come nel rigo normale.

upper = \relative c' {  
  \time 12/8  
  \key e \minor  
  \voiceOne  
  r4. r8 e, fis g16 b g e e' b c b a g fis e  
}

lower = \relative c {  
  \key e \minor  
  \voiceTwo  
  r16 e d c b a g4 fis8 e fis g a b c  
}

\score {  
  <<  
    \new StaffGroup = "tab with traditional" <<  
    \new Staff = "guitar traditional" <<  
      \clef "treble_8"  
      \new Voice = "upper" \upper  
      \new Voice = "lower" \lower  
    >>  
    \new TabStaff = "guitar tab" <<  
      \new TabVoice = "upper" \upper  
      \new TabVoice = "lower" \lower  
    >>  
  >>  
  \new StaffGroup = "guitar staff group" <<  
  \new Staff = "guitar staff" <<  
    \clef "treble_8"  
    \new Voice = "upper" \upper  
    \new Voice = "lower" \lower  
  >>  
  >>  
  >>  
}
Slides in tablature

Slides can be typeset in both Staff and TabStaff contexts:

```latex
slides = {
    c'8\3\{\text{glissando} d'8\3
    c'8\3\text{glissando} d'8\3
    hideNotes
    \text{grace} \{ g16\text{glissando} \}
    \text{unHideNotes}
    c'4\3
    \text{afterGrace} d'4\3\text{glissando} \{
    \text{stemDown} \text{hideNotes}
    g16 \}
    \text{unHideNotes}
}
```

```latex
\score { << \new Staff { \clef "treble_8" \slides }
\new TabStaff { \slides } >>
\layout { \context { \Score
    \override Glissando.minimum-length = #4
    \override Glissando.springs-and-rods = #ly:spanner::set-spacing-rods
    \override Glissando.thickness = #2
    \omit StringNumber
    \% or:
    \%\override StringNumber.stencil = ##f
} }
}
```

Comportamento di gambi e travature in intavolatura

La direzione dei gambi nell'intavolatura è regolata nello stesso modo della notazione tradizionale. Le travature possono essere rese orizzontali, come illustrato in questo esempio.

```latex
\new TabStaff {
    \relative c {
        \tabFullNotation
        g16 b d g b d g b
        \stemDown
        \override Beam.concaveness = #10000
    }
```
String number extender lines

Make an extender line for string number indications, showing that a series of notes is supposed to be played all on the same string.

```
stringNumberSpanner =
  #(define-music-function (StringNumber) (string?))
  #{
    \override TextSpanner.style = #'solid
    \override TextSpanner.font-size = #-5
    \override TextSpanner.bound-details.left.stencil-align-dir-y = #CENTER
    \markup { \circle \number $StringNumber }
  }
```

```
\relative c {
  \clef "treble_8"
  \stringNumberSpanner "5"
  \textSpannerDown
  a8\startTextSpan
  b c d e f\stopTextSpan
  \stringNumberSpanner "4"
  g\startTextSpan a
  bes4 a g2\stopTextSpan
}```
Unfretted strings

Sezione “Unfretted string instruments” in Guida alla Notazione

Creating slurs across voices

In some situations, it may be necessary to create slurs between notes from different voices. The solution is to add invisible notes to one of the voices, using \hideNotes.

This example is measure 235 of the Ciaconna from Bach’s 2nd Partita for solo violin, BWV 1004.

\relative c' { \vskip 10.5pt
<<
{ d16(a') s a s a[ s a] s a[ s a]
} \slurUp
bes,16[ s e]( \hideNotes a) \unHideNotes f[(
\hideNotes a) \unHideNotes fis}( \hideNotes a) \unHideNotes g[(
\hideNotes a) \unHideNotes gis]( \hideNotes a)
}
>>}

Dotted harmonics

Artificial harmonics using \harmonic do not show dots. To override this behavior, set the context property harmonicDots.

\relative c''' { \vskip 10.5pt \time 3/4 \key f \major \set harmonicDots = ##t <bes f'\harmonic>2. ~ <bes f'\harmonic>4. <a e'\harmonic>8( <gis dis'\harmonic> <g d'\harmonic>) <fis cis'\harmonic>2. <bes f'\harmonic>2. }
Snap-pizzicato or Bartok pizzicato

A snap-pizzicato (also known as “Bartok pizzicato”) is a “strong pizzicato where the string is plucked vertically by snapping and rebounds off the fingerboard of the instrument” (Wikipedia). It is denoted by a circle with a vertical line going from the center upwards outside the circle.

\relative c' {  
c4\snappizzicato  
<c' e g>4\snappizzicato  
<c' e g>4^\snappizzicato  
<c, e g>4_\snappizzicato
}

Modello per quartetto d’archi (semplice)

Questo modello presenta un semplice quartetto d’archi. Impiega anche una sezione \global per definire il tempo e l’armatura di chiave.

\global{  
\time 4/4  
\key c \major
}

violinOne = \new Voice \relative c' {  
c2 d  
e1  
\bar "|."  
}

violinTwo = \new Voice \relative c' {  
g2 f  
e1  
\bar "|."  
}

viola = \new Voice \relative c' {  
\clef alto  
e2 d  
c1  
\bar "|."  
}
cello = \new Voice \relative c' { 
  \clef bass 
  c2 b 
  a\_\_\_\_ \_\_\_\_ \_\_\_\_ 
  \bar "|.|." 
} 

\score { 
  \new StaffGroup << 
    \new Staff \with { \instrumentName = "Violin 1" } 
    \global \violinOne >> 
    \new Staff \with { \instrumentName = "Violin 2" } 
    \global \violinTwo >> 
    \new Staff \with { \instrumentName = "Viola" } 
    \global \viola >> 
    \new Staff \with { \instrumentName = "Cello" } 
    \global \cello >> 
  >> 
  \layout { } 
  \midi { } 
} 

\Modello per quartetto d'archi con parti separate

Il frammento di codice del "Modello per quartetto d'archi" crea un bel quartetto, ma cosa fare se si ha bisogno di creare le singole parti? Questo nuovo modello mostra come usare la funzionalità \tag per dividere facilmente un pezzo in parti staccate.


Non dimenticare di togliere i commenti quando usi i file separati!

%%%%% piece.ly
%%%%% (This is the global definitions file)

global= { 
  \time 4/4 
  \key c \major
Unfretted strings

Violinone = \new Voice {
  \relative c'' {
    c2 d e1
  }\bar "|.
}

Violintwo = \new Voice {
  \relative c'' {
    g2 f e1
  }\bar "|.
}

Viola = \new Voice {
  \relative c' {
    \clef alto
    e2 d c1
  }\bar "|.
}

Cello = \new Voice {
  \relative c' {
    \clef bass
    c2 b a1
  }\bar "|.
}

music = {
  <<
    \tag #'score \tag #'vn1
    \new Staff \with { instrumentName = "Violin 1" } << \global \Violinone >>

    \tag #'score \tag #'vn2
    \new Staff \with { instrumentName = "Violin 2" } << \global \Violintwo>>

    \tag #'score \tag #'vla
    \new Staff \with { instrumentName = "Viola" } << \global \Viola>>

    \tag #'score \tag #'vlc
\new Staff \with { instrumentName = "Cello" }
<< \global \Cello >>
>>
}

% These are the other files you need to save on your computer

% score.ly
% (This is the main file)

% uncomment the line below when using a separate file
% include "piece.ly"

#(set-global-staff-size 14)

\score { 
  \new StaffGroup \keepWithTag #'score \music
  \layout { } 
  \midi { }
}

% Uncomment this block when using separate files

% vn1.ly
% (This is the Violin 1 part file)

\include "piece.ly"
\score { 
  \keepWithTag #'vn1 \music
  \layout { }
}

% vn2.ly
% (This is the Violin 2 part file)

\include "piece.ly"
\score { 
  \keepWithTag #'vn2 \music
  \layout { }
}

% vla.ly
% (This is the Viola part file)

\include "piece.ly"
\score { 
  \keepWithTag #'vla \music
  \layout { }
}
\% vlc.ly
\% (This is the Cello part file)

\include "piece.ly"
\score {
    \keepWithTag #'vlc \music
    \layout { }
}

%}
Winds

Sezione “Wind instruments” in Guida alla Notazione

Cambiare la dimensione dei diagrammi per legni

La dimensione e lo spessore dei diagrammi per legni possono essere modificati.

\relative c' { 
  \textLengthOn
  \override \woodwind-diagram #'piccolo
  #'()

  \override #'(size . 1.5) { \woodwind-diagram #'piccolo
    #'()
  }

  \override #'(thickness . 0.15) { \woodwind-diagram #'piccolo
    #'()
  }
}

Simboli di diteggiatura per strumenti aerofoni

Si possono ottenere simboli speciali combinando glifi esistenti; ciò è utile per gli strumenti aerofoni.

centermarkup = {
  \once \override TextScript.self-alignment-X = #CENTER
  \once \override TextScript.X-offset =#(lambda (g) (+ (ly:self-alignment-interface::centered-on-x-parent g)
                        (ly:self-alignment-interface::x-aligned-on-self g)))
It is possible to indicate special articulation techniques such as a flute “tongue slap” by replacing the note head with the appropriate glyph. For that we can draw the accent-like note head with \markup.

\slap =
\define-music-function (music) (ly: music?)
#{
  \override NoteHead.stencil = ly:text-interface::print
  \override NoteHead.text =
  \markup
  \translate #'(1 . 0)
  \thickness #'(1 . 0)
  \overlay {
    \draw-line #'-1 . 0)
    \draw-line #'-1 . -0.4)
  }
  \stem-attachment =
  \lambda (grob) (let* (stem (ly: grob-object grob 'stem))
    (dir (ly: grob-property stem 'direction UP))
    (is-up (eqv? dir UP)))
  (cons dir (if is-up 0 -0.8))))
}\revert NoteHead.stencil
Diagrammi grafici e testuali per i legni

In molti casi, le chiavi diverse da quelle della colonna centrale possono essere visualizzate per nome oltre che in forma grafica.

Grafico della diteggiatura per flauto dolce

L'esempio seguente illustra come realizzare grafici delle diteggiature per strumenti aerofoni.

```plaintext
\relative c' {
  c4 \slap c d r
  \slap { g4 a } b r
}

\relative c'' {
  \textLengthOn
  c1~\markup
    \woodwind-diagram
      #'piccolo
      #'((cc . (one three))
      (1h . (gis))
      (rh . (ees)))

  c1~\markup
    \override #'(graphical . #f) {
      \woodwind-diagram
        #'piccolo
        #'((cc . (one three))
          (1h . (gis))
          (rh . (ees)))
    }
}

\centermarkup = {

  \boxedtext{Eb}
}

% range chart for paetzold contrabass recorder

centermarkup = {

\once \override TextScript.self-alignment-X = #CENTER
\once \override TextScript.X-offset = #(lambda (g)
  (+ (ly:self-alignment-interface::centered-on-x-parent g)
      (ly:self-alignment-interface::x-aligned-on-self g)))
}

\score {\new Staff \with {
  \remove "Time_signature_engraver"
  \omit Stem
  \omit Flag
  \consists "Horizontal_bracket_engraver"
}
{
\clef bass
\set Score.timing = ##f
f,1\times\quarter \glissando
\clef violin
gis'1\times\quarter
\stemDown a'4\markup {1})
\centermarkup
\once \override TextScript.padding = #2
bes'1\times\quarter \markup {\override #'(baseline-skip . 1.7) \column
  { \fontsize #-5 \slashed-digit #0 \finger 1 \finger 2
    \finger 3 \finger 4 \finger 5 \finger 6 \finger 7} }
b'1\times\quarter
c''4\markup {1})
\centermarkup
\once \override TextScript.padding = #2
cis''1\times\quarter
deh''1\times\quarter
\centermarkup
\once \override TextScript.padding = #2
\once \override Staff.HorizontalBracket.direction = #UP
e'''1\times\quarter \markup {\override #'(baseline-skip . 1.7) \column
  { \fontsize #-5 \slashed-digit #0 \finger 1 \finger 2
    \finger 4 \finger 5} } \startGroup
f'''1\times\quarter \markup {2}) \stopGroup
}
Elenco delle chiavi dei diagrammi per legni

Il seguente frammento produce un elenco di tutte le possibili chiavi e delle loro impostazioni per i diagrammi per legni, come sono definite in `scm/define-woodwind-diagrams.scm`. L’elenco sarà visualizzato nel file di log, ma non nello spartito. Se si desidera che l’output appaia nella console, omettere dai comandi (current-error-port).

```lilypond
#(print-keys-verbose 'piccolo (current-error-port))
#(print-keys-verbose 'flute (current-error-port))
#(print-keys-verbose 'flute-b-extension (current-error-port))
#(print-keys-verbose 'tin-whistle (current-error-port))
#(print-keys-verbose 'oboe (current-error-port))
#(print-keys-verbose 'clarinet (current-error-port))
#(print-keys-verbose 'bass-clarinet (current-error-port))
#(print-keys-verbose 'low-bass-clarinet (current-error-port))
#(print-keys-verbose 'saxophone (current-error-port))
#(print-keys-verbose 'soprano-saxophone (current-error-port))
#(print-keys-verbose 'alto-saxophone (current-error-port))
#(print-keys-verbose 'tenor-saxophone (current-error-port))
#(print-keys-verbose 'baritone-saxophone (current-error-port))
#(print-keys-verbose 'bassoon (current-error-port))
#(print-keys-verbose 'contrabassoon (current-error-port))
\score {c'1}
```

Elenco dei diagrammi per i legni

L’esempio seguente mostra tutti i diagrammi per i legni attualmente definiti in LilyPond.

```lilypond
\layout {
    indent = 0
}

\relative c' {
    \textLengthOn
    c1^
    \markup {
        \center-column {
            'tin-whistle
            " "
            \woodwind-diagram
            #'tin-whistle
            #'()
        }
    }
    c1^
    \markup {
        \center-column {
            'piccolo
        }
    }
```
Winds

\begin{center-column}
\begin{verbatim}
\woodwind-diagram
    # 'piccolo
    # '()
\end{verbatim}
\end{center-column}

\begin{center-column}
\begin{verbatim}
\woodwind-diagram
    # 'flute
    # '()
\end{verbatim}
\end{center-column}

\begin{center-column}
\begin{verbatim}
\woodwind-diagram
    # 'oboe
    # '()
\end{verbatim}
\end{center-column}

\begin{center-column}
\begin{verbatim}
\woodwind-diagram
    # 'clarinet
    # '()
\end{verbatim}
\end{center-column}

\begin{center-column}
\begin{verbatim}
\woodwind-diagram
    # 'bass-clarinet
    # '()
\end{verbatim}
\end{center-column}

\begin{center-column}
\begin{verbatim}
\woodwind-diagram
\end{verbatim}
\end{center-column}
Ancient notation

Sezione “Ancient notation” in *Guida alla Notazione*

Adding a figured bass above or below the notes

When writing a figured bass, you can place the figures above or below the bass notes, by defining the `BassFigureAlignmentPositioning.direction` property (exclusively in a `Staff` context). Choices are #UP (or #1), #CENTER (or #0) and #DOWN (or #-1).

This property can be changed as many times as you wish. Use `\once \override` if you don’t want the override to apply to the whole score.

```lilypond
bass = {
  \clef bass
  g4 b, c d
e  d8 c d2
}

\context Staff = bassStaff 
\override Staff.BassFigureAlignmentPositioning.direction = #UP
%\bassFigureStaffAlignmentUp
< _+>4 <6>
\set Staff.useBassFigureExtenders = ##t
\override Staff.BassFigureAlignmentPositioning.direction = #DOWN
%\bassFigureStaffAlignmentDown
<4>4. <4>8 < _+>4
}

\score {
<<
  \new Staff = bassStaff \bass
  \context Staff = bassStaff \continuo
>>
}
```

![Example of figured bass](image)

Ancient fonts

Shown here are many of the symbols that are included in LilyPond’s ancient notation.

```lilypond
upperStaff = \new VaticanaStaff = "upperStaff" <<
\context VaticanaVoice <<
  \transpose c c {\override NoteHead.style = #'vaticana.punctum
\key es \major
\clef "vaticana-fa2"
c1 des e f ges
```
\override NoteHead.style = #'vaticana.inclinatum
a! b ces'
\bar "|"
\override NoteHead.style = #'vaticana.quilisma
b! des'! ges! fes!
\breathe
\clef "vaticana-fa1"
\override NoteHead.style = #'vaticana.plica
es d
\override NoteHead.style = #'vaticana.reverse.plica
c d
\bar "|"
\override NoteHead.style = #'vaticana.punctum.cavum
es f
\override NoteHead.style = #'vaticana.lpes
g as
\override NoteHead.style = #'vaticana.upes
bes as
\override NoteHead.style = #'vaticana.vupes
g f
\override NoteHead.style = #'vaticana.linea.punctum
\once \override Staff.BarLine.bar-extent = #'(-1 . 1) \bar "|"
es d
\override NoteHead.style = #'vaticana.epiphonus
c d
\override NoteHead.style = #'vaticana.cephalicus
es f
\set Staff.alterationGlyphs =
  #alteration-medicaea-glyph-name-alist
\override Staff.Custos.style = #'medicaea
\override NoteHead.style = #'medicaea.punctum
\clef "medicaea-fa2"
ces des
\bar "|"
e! f! ges
\clef "medicaea-do2"
\override NoteHead.style = #'medicaea.inclinatum
a! b! ces'
\override NoteHead.style = #'medicaea.virga
b! a!
\bar "|"
ges fes
\clef "medicaea-fa1"
\override NoteHead.style = #'medicaea.rvirga
e des ces
\set Staff.alterationGlyphs =
#alteration-hufnagel-glyph-name-alist
\override Staff.Custos.style = #'hufnagel
\override NoteHead.style = #'hufnagel.punctum
\clef "hufnagel-fa2"
ces des es
\bar "|

fes ges
\clef "hufnagel-do2"
\override NoteHead.style = #'hufnagel.lpes
as! bes! ces'
\override NoteHead.style = #'hufnagel.virga
bes! as!
\bar "|

ges! fes!
\clef "hufnagel-do-fa"
\override NoteHead.style = #'hufnagel.punctum
es! des ces des! es! fes!
\bar "||

s32*1
}
>>
>>

lowerStaff = \new MensuralStaff = "lowerStaff" <<
\context MensuralVoice <<
\transpose c c {

\key a \major
cis'1 d'\breve gis'\breve e'\breve e'[ e'\longa fis'\longa ]
\set Staff.forceClef = ##t
\clef "neomensural-c2"
cis1
\bar "|

[ g\breve dis' \longa ]
b'\breve e [ a\longa d'\longa ]
\clef "petrucci-c2"

fis1 ces1
\clef "petrucci-c2"
r'\longa
\set Staff.forceClef = ##t
\clef "mensural-c2"
r'\breve e
\bar "|

r2
\clef "mensural-g"
r4 r8 r16 r16
\override NoteHead.style = #'mensural
\override Rest.style = #'mensural
\clef "petrucci-f"
c8 b, c16 b, c32 b, c64 b, c64 b,
d8 e d16 e d32 e d64 e d64 e
r\longa
\set Staff.forceClef = ##t
\clef "petrucci-f"
r\breve
\bar "|
\breve
\clef "mensural-f"
r2 r4 r8 r16 r16
\set Staff.forceClef = ##t
\clef "mensural-f"
e\breve f g a1
\clef "mensural-g"

[ bes'\longa a'\longa c'\longa \longa ]
e'1 d' c' d' \bar "|
\bar "|
bes'\longa fis'1 as'1 ges'\longa % lig
\set Staff.forceClef = ##t
\clef "mensural-g"
e'2 d' c' \bar "|

\set Staff.forceClef = ##t
\clef "petrucci-g"
c'2 d' e' f'
\clef "petrucci-g"
g' as'! bes'! cis'! bes'! as'! gis'! fis'!
\set Staff.forceClef = ##t
\clef "mensural-g"
es'! des'! cis'!1 \bar "||"
}

>>

>>

\paper {
    line-thickness = #(/ staff-space 5.0)
}

\score {
    <<
        \upperStaff
        \lowerStaff
    >>
\layout {
  indent = 0.0
  \context {
    \Score
      timing = ##f
  }
  \context {
    \MensuralVoice
      \override NoteHead.style = #'neomensural
      \override Rest.style = #'neomensural
      \override Flag.style = #'mensural
      \override Stem.thickness = #1.0
  }
  \context {
    \MensuralStaff
      \revert BarLine.transparent
      alterationGlyphs =
        #alteration-mensural-glyph-name-alist
    clefGlyph = "#clefs.petrucci.c2"
  }
  \context {
    \VaticanaStaff
      \revert BarLine.transparent
      \override StaffSymbol.thickness = #2.0
      alterationGlyphs =
        #alteration-vaticana-glyph-name-alist
    \override Custos.neutral-position = #4
  }
}
Modello per notazione antica – trascrizione moderna di musica gregoriana

Questo esempio mostra come realizzare una trascrizione moderna di musica gregoriana. La musica gregoriana non presenta la suddivisione in misure né gambi; impiega soltanto le teste della minima e della semiminima, e dei segni appositi che indicano pause di diversa lunghezza.

\include "gregorian.ly"

\begin{lyxcode}
\set Score.timing = ##f
\begin{verbatim}
\begin{lyxcode}
\textit{Lo -- rem ip -- sum do -- lor sit a -- met}
\end{verbatim}
\end{lyxcode}
\end{verbatim}
\end{lyxcode}

\score {
\new GregorianTranscriptionStaff <<
\new GregorianTranscriptionVoice = "melody" \chant
\new GregorianTranscriptionLyrics = "one" \lyricsto melody \verba
>>
}

Lorem ipsum dolor sit amet
Ancient time signatures

Time signatures may also be engraved in an old style.

\{ \override Staff.TimeSignature.style = #'neomensural

\}

\[ \begin{array}{c}
\text{\includegraphics[width=0.5\textwidth]{image}}
\end{array} \]

Notazione per canti e salmi

Questa forma di notazione `e utilizzata per i salmi, dove i versi non sono sempre della stessa lunghezza.

\[ \begin{array}{l}
\text{stemOff} = \text{\hide Staff.Stem}
\text{stemOn} = \text{\undo \stemOff}
\end{array} \]

\[ \begin{array}{l}
\text{\score { }
\text{\new Staff { }
\text{\with { }
\text{\remove "Time_signature_engraver" }
\text{\ }
\text{\key g \minor}
\text{\cadenzaOn}
\text{\stemOff a\'\breve bes\'4 g\'4}
\text{\stemOn a\'2 \section}
\text{\stemOff a\'\breve g\'4 a\'4}
\text{\stemOn f\'2 \section}
\text{\stemOff a\'\breve\markups { \italic flexe } }
\text{\stemOn g\'2 \fine}
\text{\}
\text{\}
\text{\}
\end{array} \]

\[ \begin{array}{c}
\text{\includegraphics[width=0.5\textwidth]{image}}
\end{array} \]

Custodes

Custodes may be engraved in various styles.

\[ \begin{array}{l}
\text{\layout { \ragged-right = #\t }}
\end{array} \]

\[ \begin{array}{l}
\text{\new Staff { }
\text{\with { }
\text{\consists "Custos_engraver" }
\text{\relative c' { }
\text{\override Staff.Custos.neutral-position = #4}
\text{\}
\end{array} \]

\[ \begin{array}{l}
\text{\override Staff.Custos.style = #'hufnagel}
\text{c1\"hufnagel" \break}
\text{<d a' f'>1}
\text{\}
\end{array} \]

\[ \begin{array}{l}
\text{\override Staff.Custos.style = #'medicæa}
\text{c1\"medicæa" \break}
\text{<d a' f'>1}
\text{\}
\end{array} \]
Quando si trascrive musica mensurale, un incipit all’inizio del brano è utile per indicare il tempo e l’armatura di chiave originali. I musicisti oggi sono abituati alle stanghette, ma queste non erano note all’epoca della musica mensurale. Come compromesso, spesso le stanghette vengono poste tra i righi, uno stile di formattazione chiamato mensurstriche.

%% With 2.23. this throws:
%% programming error: Loose column does not have right side to attach to.
%% Likely "Hidden BarLine during note yields programming error"
%% https://gitlab.com/lilypond/lilypond/-/issues/4084
%% --Harm

% A short excerpt from the Jubilate Deo by Orlande de Lassus

global = {
  \set Score.skipBars = ##t
  \key g \major
  \time 4/4
% the actual music
\skip 1*8

% let finis bar go through all staves
\override \Score.StaffBarLine.transparent = ##f

% finis bar
\bar "|.

\discantusIncipit = {
  \clef "neomensural-c1"
  \key f \major
  \time 2/2
  c''1.
}

\discantusNotes = {
  \transpose c' c'' {
    \clef "treble"
    d'2. d'4 | 
    b e' d'2 | 
    c'4 e'4.( d'8 c' b | 
    a4) b a2 | 
    b4.( c'8 d'4) c'4 | 
    \once \hide NoteHead
    c'1 | 
    b\breve | 
}
}

\discantusLyrics = \lyricmode {
  Ju -- bi -- la -- te De -- o,
  om -- nis ter -- ra, __ om-
  "..."  
  -us.
}

\altusIncipit = {
  \clef "neomensural-c3"
  \key f \major
  \time 2/2
  r1 f'1.
}

\altusNotes = {
  \transpose c' c'' {
    \clef "treble"
    r2 g2. e4 fis g | 
    a2 g4 e | 
    fis g4.( fis16 e fis4) |
Ju -- bi -- la -- te
De -- o, om -- nis ter -- ra,
"..."
-us.

Ju -- bi -- la -- te
"..."
-us.
\transpose c' c' {
\clef "bass"
R1 |
R1 |
R1 |
R1 |
g2. e4 |
\once \hide NoteHead
e1 |
g\breve |
}
}
bassusLyrics = \lyricmode {
Ju -- bi-
"..."
-\us.
}
\score {
<<
\new StaffGroup = choirStaff <<
\new Voice = "discantusNotes" <<
  \set Staff.instrumentName = "Discantus"
  \incipit \discantusIncipit
  \global \discantusNotes
>>
\new Lyrics \lyricsto discantusNotes { \discantusLyrics }
\new Voice = "altusNotes" <<
  \set Staff.instrumentName = "Altus"
  \global \incipit \altusIncipit
  \altusNotes
>>
\new Lyrics \lyricsto altusNotes { \altusLyrics }
\new Voice = "tenorNotes" <<
  \set Staff.instrumentName = "Tenor"
  \global \incipit \tenorIncipit
  \tenorNotes
>>
\new Lyrics \lyricsto tenorNotes { \tenorLyrics }
\new Voice = "bassusNotes" <<
  \set Staff.instrumentName = "Bassus"
  \global \incipit \bassusIncipit
  \bassusNotes
>>
\new Lyrics \lyricsto bassusNotes { \bassusLyrics }
>>
\layout {
  \context {
    \Score
    \hide BarLine
  }
  \context {
    \Lyrics
    \consists "Bar_engraver"
    \consists "Separating_line_group_engraver"
  }
  \context {
    \Voice
    \hide Slur
    \% Comment in the below \remove command to allow line
    \% breaking also at those bar lines where a note overlaps
    \% into the next measure. The command is commented out in this
    \% short example score, but especially for large scores, you
    \% will typically yield better line breaking and thus improve
    \% overall spacing if you comment in the following command.
    \remove "Forbid_line_break_engraver"
  }
  \indent = 6\cm
  \incipit-width = 4\cm
}
}
Formattazione mensurale (stanghette tra i righi)

La formattazione mensurale, in cui le stanghette non appaiono sui righi ma nello spazio tra i righi, si può ottenere usando StaffGroup al posto di ChoirStaff. La stanghetta sui righi viene nascosta con \hide.

\layout {
    \context {
        \Staff
            \measureBarType = "-span"
    }
}

\new StaffGroup <<
    \new Staff \music
    \new Staff \music
>>

Stili di pausa

Esistono vari stili di pausa.

\new Staff \relative c { 
    \omit Score.TimeSignature
\cadenza\n
\override Staff.Rest.style = #'mensural
r\maxima\markup \typewriter { mensural }
r\longa r\breve r1 r2 r4 r8 r16 s32 s64 s128 s128
\bar ""
\break

\override Staff.Rest.style = #'neomensural
r\maxima\markup \typewriter { neomensural }
r\longa r\breve r1 r2 r4 r8 r16 s32 s64 s128 s128
\bar ""
\break

\override Staff.Rest.style = #'classical
r\maxima\markup \typewriter { classical }
r\longa r\breve r1 r2 r4 r8 r16 r32 r64 r128 s128
\bar ""
\break

\override Staff.Rest.style = #'z
r\maxima\markup \typewriter { z-style }
r\longa r\breve r1 r2 r4 r8 r16 r32 r64 r128 s128
\bar ""
\break

\override Staff.Rest.style = #'default
r\maxima\markup \typewriter { default }
r\longa r\breve r1 r2 r4 r8 r16 r32 r64 r128 s128
}
Usare le etichette per produrre musica mensurale e moderna dallo stesso sorgente

Usando le etichette (tag), è possibile usare la stessa musica per produrre sia la musica mensurale che quella moderna. In questo frammento, viene introdotta la funzione `menrest`, che permette alle pause mensurali di essere posizionate precisamente sul rigo come nell’originale, ma con le pause moderne nella posizione standard. Le etichette vengono usate per produrre diversi tipi di stanghetta alla fine della musica, ma possono essere usate anche quando sono necessarie altre differenze: per esempio se si vogliono usare “pause d’intero” (R1, R\breve, etc.) nella musica moderna, ma pause normali (r1, r\breve, etc.) nella versione mensurale. La conversione di musica mensurale nel suo equivalente moderno viene solitamente chiamata trascrizione.

```plaintext
menrest = #(define-music-function (note) (ly:music?))
  ({
    \tag #'mens $(make-music 'RestEvent note)
    \tag #'mod $(make-music 'RestEvent note 'pitch '())
  })

MenStyle = {
  \autoBeamOff
  \override NoteHead.style = #'petrucci
  \override Score.BarNumber.transparent = ##t
  \override Stem.neutral-direction = #up
}

finalis = \section

Music = \relative c’' {
  \set Score.\tempoHideNote = ##t
  \key f \major
  \time 4/4
  g1 d’2 \menrest bes4 bes2 a2 r4 g4 fis2.
  \finalis
}

MenLyr = \lyricmode { So farre, deere life, deare life }
ModLyr = \lyricmode { So far, dear life, dear life }

\score {
  \keepWithTag #'mens {
    <<
      \new MensuralStaff {
        \new MensuralVoice = Cantus
          \clef "mensural-c1" \MenStyle \Music
      }
      \new Lyrics \lyricsto Cantus \MenLyr
    >>
  }
}

\score {
  \keepWithTag #'mod {
    <<
      \new MensuralStaff {
        \new MensuralVoice = Cantus
          \clef "mensural-c1" \MenStyle \Music
      }
      \new Lyrics \lyricsto Cantus \MenLyr
    >>
  }
}
```
Vertical line as a baroque articulation mark

This short vertical line placed above the note is commonly used in baroque music. Its meaning can vary, but generally indicates notes that should be played with more “weight”. The following example demonstrates how to achieve such a notation.

\texttt{upline} =
\texttt{\tweak stencil}
\texttt{\#(lambda (grob)}
\texttt{\quad (grob-interpret-markup grob \{( \markup \draw-line #'(0 . 1) \}}))
\texttt{\stopted}
\texttt{\relative c' \{}
\texttt{\quad a'4^\texttt{upline} a( c d')_\texttt{upline}}
\texttt{\}}
World music

Sezione “World music” in Guida alla Notazione

Improvvisazione araba

Per improvvisazioni o taqasim a tempo libero, si può omettere l’indicazione di tempo e usare \cadenzaOn. Può essere necessaria la modifica dello stile delle alterazioni, perché l’assenza delle stanghette farà sì che l’alterazione sia contrassegnata una volta sola. Ecco un esempio di quello che potrebbe essere l’inizio di un’improvvisazione hijaz:

\include "arabic.ly"

\relative sol' {
  \key re \kurd
  \accidentalStyle forget
  \cadenzaOn
  sol4 sol sol fad mib sol1 fad8 mib re4. r8 mib1 fad sol
}

\Esempio di makam

Makam è un tipo di melodia proveniente dalla Turchia che usa alterazioni microtonali di 1/9. Consultare il file di inizializzazione ‘ly/makam.ly’ per vedere come sono definiti i nomi delle altezze e le alterazioni.

% Initialize makam settings
\include "makam.ly"

\relative c' {
  \set Staff.keyAlterations = #'((6 , ,(- KOMA)) (3 , ,BAKIYE))
  c4 cc db fk
  gbm4 gfc gfb efk
  fk4 db cc c
}

\Printing text from right to left

It is possible to print text from right to left in a markup object, as demonstrated here.

{ b1^\markup {
  \line { i n g i r u m i m u s n o c t e }
}
  f'\markup {
    \override #'(text-direction . -1)
  
}
Turkish Makam example

This template uses the start of a well-known Turkish Saz Semai that is familiar in the repertoire in order to illustrate some of the elements of Turkish music notation.

```
\line { ingirimimusnocte }
\}
}\n
\honeytongue { ingirimimusnocte }

Turkey Makam example

This template uses the start of a well-known Turkish Saz Semai that is familiar in the repertoire in order to illustrate some of the elements of Turkish music notation.

```
paper { tagline = ##f }
%
\setStaff.extraNatural = ##f
\setStaff.autoBeaming = ##f
\key a \huseyni
\time 10/8
a'4 g'16 [fb] e8. [d16] d [c d e] c [d c8] bfc |
a16 [bfc a8] bfc c16 [d c8] d16 [e d8] e4 fb8 |
d4 a'8 a16 [g fb e] fb8 [g] a8. [b16] a16 [g] |
g4 g16 [fb] fb8. [e16] e [g fb e] e4 r8 |
```

Hüseyni Saz Semaisi

Lavtacı Andon
Adding a figured bass above or below the notes

When writing a figured bass, you can place the figures above or below the bass notes, by defining the \staffBassFigureAlignmentPositioning.direction property (exclusively in a \staff context). Choices are \#UP (or \#1), \#CENTER (or \#0) and \#DOWN (or \#-1).

This property can be changed as many times as you wish. Use \once \override if you don’t want the override to apply to the whole score.

```latex
\begin{verbatim}
\new Staff = bassStaff \bass
\context \Staff = bassStaff \continuo
\end{verbatim}
```

Adding an extra staff

An extra staff can be added (possibly temporarily) after the start of a piece.

```latex
\begin{verbatim}
\new \relative c' { \c1 | c | c | c | c }
\new \StaffGroup \relative c'' { \new \Staff { ...
\end{verbatim}
```
Adding an extra staff at a line break

When adding a new staff at a line break, some extra space is unfortunately added at the end of the line before the break (to fit in a key signature change, which will never be printed anyway). The workaround is to add a setting of Staff.explicitKeySignatureVisibility as is shown in the example.

```
\score { 
  \new StaffGroup \relative c'' { 
    \new Staff 
    \key f \major 
    c1 c-"Unwanted extra space" \break 
    << { c1 | c } 
    \new Staff { 
      \key f \major 
      \once \omit Staff.TimeSignature 
      c1 | c 
    } 
    \break 
    \new Staff { 
      \key f \major 
      \once \omit Staff.TimeSignature 
      \set Staff.explicitKeySignatureVisibility = #end-of-line-invisible 
      c1 | c 
    } 
    \break 
    \new Staff { 
      \key f \major 
      \once \omit Staff.TimeSignature 
      c1 | c 
    } 
  } 
} 
```
Cambiare automaticamente la direzione del gambo della nota centrale in base alla melodia

LilyPond può modificare la direzione del gambo della nota centrale di un rigo in modo che seguia la melodia: occorre aggiungere l’incisore Melody_engraver al contesto Voice.

La proprietà di contesto suspendMelodyDecisions può essere usata per disattivare questo comportamento localmente.

```lp
\relative c'' { 
  \time 3/4
  a8 b g f b g |
  \set suspendMelodyDecisions = ##t
  a b g f b g |
  \unset suspendMelodyDecisions
  c b d c b c |
}

\layout { 
  \context { 
    \Voice 
    \consists "Melody_engraver"
    \autoBeamOff
  } 
}```
Centered measure numbers

Scores of large ensemble works often have bar numbers placed beneath the system, centered horizontally on the measure’s extent. This snippet shows how the Measure_counter_engraver may be used to simulate this notational practice. Here, the engraver has been added to a Dynamics context.

This snippet presents a legacy method: starting from LilyPond 2.23.3, \set Score.centerBarNumbers = ##t is enough.

\layout {
  \context {
    Dynamics
    \consists #Measure_counter_engraver
    \override MeasureCounter.direction = #DOWN
    \override MeasureCounter.font-encoding = #'latin1
    \override MeasureCounter.font-shape = #'italic
    \% to control the distance of the Dynamics context from the staff:
    \override VerticalAxisGroup.nonstaff-relatedstaff-spacing.padding = #2
  }
  \context {
    \Score
    \remove "Bar_number_engraver"
  }
}

pattern = \repeat unfold 7 { c'4 d' e' f' }

\new StaffGroup <<
  \new Staff {
    \pattern
  }
  \new Staff {
    \pattern
  }
  \new Dynamics {
    \startMeasureCount s1*7
    \stopMeasureCount
  }
}>>
Impostare l’output MIDI su un canale per voce

Nella creazione del file di output MIDI, il comportamento predefinito prevede che ogni rigo sia assegnato a un canale MIDI, con tutte le voci del rigo amalgamate in un canale. Ciò diminuisce il rischio di esaurire i canali MIDI disponibili, dato che ce ne sono solo 16 per traccia.

Tuttavia, spostando Staff_performer nel contesto Voice, ogni voce in un rigo può avere il proprio canale MIDI, come è illustrato nell’esempio seguente: sebbene le voci siano sullo stesso rigo, vengono creati due canali MIDI, ciascuno con un diverso strumento MIDI (midiInstrument).

\score {
  \new Staff <<
    \new Voice \relative c'\prime { 
        \set midiInstrument = #'flute' 
        \voiceOne 
        \key g \major 
        \time 2/2 
        r2 g-'Flute' ~ 
        g fis ~ 
        fis4 g8 fis e2 ~ 
        e4 d8 cis d2 
    } 
    \new Voice \relative c' { 
        \set midiInstrument = #'clarinet' 
        \voiceTwo 
        b1-'Clarinet' 
        a2. b8 a 
        g2. fis8 e 
        fis2 r 
    } 
  >>
}

\context { 
  \Staff 
  \remove "Staff_performer" 
}
\context { 
  \Voice 
  \consists "Staff_performer" 
}
\tempo 2 = 72 
}
Changing time signatures inside a polyrhythmic section using \scaleDurations

The measureLength property, together with measurePosition, determines when a bar line is needed. However, when using \scaleDurations, the scaling of durations makes it difficult to change time signatures. In this case, measureLength should be set manually, using the ly:make-moment callback. The second argument must be the same as the second argument of \scaleDurations.

\layout {
  \context {
    \Score
    \remove "Timing_translator"
  }
  \context {
    \Staff
    \consists "Timing_translator"
  }
}

<<
  \new Staff {
    \scaleDurations 8/5 {
      \time 6/8
      \set Timing.measureLength = \(#(ly:make-moment 6/5)
      b8 b b b b b
      \time 2/4
      \set Timing.measureLength = \(#(ly:make-moment 4/5)
      b4 b
    }
  }
  \new Staff {
    \clef bass
    \time 2/4
    c2 d e f
  }
>>

Notazione per canti e salmi

Questa forma di notazione è utilizzata per i salmi, dove i versi non sono sempre della stessa lunghezza.

stemOff = \hide Staff.Stem
stemOn = \undo \stemOff

\score {
  \new Staff \with { \remove "Time_signature_engraver" }
Creating blank staves

To create blank staves, generate empty measures then remove the Bar_number_engraver from the Score context, and the Time_signature_engraver, Clef_engraver and Bar_engraver from the Staff context.

#(set-global-staff-size 20)

\score {
  \layout {
    indent = 0\in
    \context {
      \Staff
      \remove "Time_signature_engraver"
      \remove "Clef_engraver"
      \remove "Bar_engraver"
    }
    \context {
      \Score
      \remove "Bar_number_engraver"
    }
  }
}

% uncomment these lines for "letter" size
%
\paper {
  #(set-paper-size "letter")
  ragged-last-bottom = ##f
  line-width = 7.5\in
  left-margin = 0.5\in
  bottom-margin = 0.25\in
  top-margin = 0.25\in
}
%}

% Uncomment these lines for "A4" size
%
\paper {
  #(set-paper-size "a4")
  ragged-last-bottom = ##f
  line-width = 180
  left-margin = 15
  bottom-margin = 10
  top-margin = 10
}
%}
Creating custom key signatures

LilyPond supports custom key signatures. In this example, print for D minor with an extended range of printed flats.

\new Staff \with { 
\override StaffSymbol.line-count = #8 
\override KeySignature.flat-positions = #'((-7 . 6)) 
\override KeyCancellation.flat-positions = #'((-7 . 6))
% presumably sharps are also printed in both octaves
\override KeySignature.sharp-positions = #'((-6 . 7)) 
\override KeyCancellation.sharp-positions = #'((-6 . 7)) 
\override Clef.stencil = # (lambda (grob) (grob-interpret-markup grob 
#( \markup \combine 
\musicglyph "clefs.C"
\translate #'(-3 . -2) 
\musicglyph "clefs.F"
#))

clefPosition = #3
middleCPosition = #3
middleCClefPosition = #3 
}

{ 
\key d\minor 
f bes, f bes,
}

\Cross staff stems

This snippet shows the use of the Span\_stem\_engraver and \crossStaff to connect stems across staves automatically.

The stem length need not be specified, as the variable distance between noteheads and staves is calculated automatically.

\layout { 
\context { 
\PianoStaff 
\consists "Span\_stem\_engraver"
}
}

{ 
\new PianoStaff <<
\new Staff { 
<b d'>4 r d'16\> e'8. g8 r\!
e'8 f' g'4 e'2
}
}
Defining an engraver in Scheme: ambitus engraver

This example demonstrates how the ambitus engraver may be defined on the user side, with a Scheme engraver. This is basically a rewrite in Scheme of the code from lily/ambitus-engraver.cc.

#(use-modules (oop goops))

%%% Grob utilities
%%% These are literal rewrites of some C++ methods used by the ambitus engraver.

#(define (ly:separation-item::add-conditional-item grob grob-item)
  "Add @var{grob-item} to the array of conditional elements of @var{grob}.
  Rewrite of @code{Separation_item::add_conditional_item} from @file{lily/separation-item.cc}."
  (ly:pointer-group-interface::add-grob grob 'conditional-elements grob-item))

#(define (ly:accidental-placement::accidental-pitch accidental-grob)
  "Get the pitch from the grob cause of @var{accidental-grob}.
  Rewrite of @code{accidental_pitch} from @file{lily/accidental-placement.cc}.
  (ly:event-property
   (ly:grob-property (ly:grob-parent accidental-grob Y) 'cause)
   'pitch))

#(define (ly:accidental-placement::add-accidental grob accidental-grob)
  "Add @var{accidental-grob}, an @code{Accidental} grob, to the list of the accidental grobs of @var{grob}, an @code{AccidentalPlacement} grob.
  Rewrite of @code{Accidental_placement::add_accidental} from @file{lily/accidental-placement.cc}.
  (let ((pitch (ly:accidental-placement::accidental-pitch accidental-grob)))
    (set! (ly:grob-parent accidental-grob X) grob)
    (let* ((accidentals (ly:grob-object grob 'accidental-grobs))
            (handle (assq (ly:pitch-notename pitch) accidentals))
            (entry (if handle (cdr handle) '()))))

\new Staff { 
  \clef bass 
  \voiceOne 
  \autoBeamOff 
  \crossStaff { <e g>4 e, g16 a8. c8} d 
  \autoBeamOn 
  g8 f g4 c2 
}'
\begin{verbatim}
(set! (ly:grob-object grob 'accidental-grobs)
  (assq-set! accidentals
    (ly:pitch-notename pitch)
    (cons accidental-grob entry))))

%%% Ambitus data structure

%%% The \texttt{<ambitus>} class holds the various grobs that are created
%%% to print an ambitus:
%%% - \texttt{ambitus-group}: the grob that groups all the components of an ambitus
%%% (Ambitus grob);
%%% - \texttt{ambitus-line}: the vertical line between the upper and lower ambitus
%%% notes (AmbitusLine grob);
%%% - \texttt{ambitus-up-note} and \texttt{ambitus-down-note}: the note head and accidental
%%% for the lower and upper note of the ambitus (see \texttt{<ambitus-note>} class
%%% below).
%%% The other slots define the key and clef context of the engraver:
%%% - \texttt{start-c0}: position of middle c at the beginning of the piece. It
%%% is used to place the ambitus notes according to their pitch;
%%% - \texttt{start-key-sig}: the key signature at the beginning of the piece. It
%%% is used to determine if accidentals shall be printed next to ambitus
%%% notes.

#(define-class <ambitus> ()
  (ambitus-group #:accessor ambitus-group)
  (ambitus-line #:accessor ambitus-line)
  (ambitus-up-note #:getter ambitus-up-note
    #:init-form (make <ambitus-note>))
  (ambitus-down-note #:getter ambitus-down-note
    #:init-form (make <ambitus-note>))
  (start-c0 #:accessor ambitus-start-c0
    #:init-value #f)
  (start-key-sig #:accessor ambitus-start-key-sig
    #:init-value '()))

%%% Accessor for the lower and upper note data of an ambitus
#(define-method (ambitus-note (ambitus <ambitus>) direction)
  "If \texttt{direction} is \texttt{UP}, then return the upper ambitus note
  of \texttt{ambitus}, otherwise return the lower ambitus note."
  (if (= direction UP)
    (ambitus-up-note ambitus)
    (ambitus-down-note ambitus)))

%%% The \texttt{<ambitus-note>} class holds the grobs that are specific to ambitus
%%% (lower and upper) notes:
%%% - \texttt{head}: an AmbitusNoteHead grob;
%%% - \texttt{accidental}: an AmbitusAccidental grob, to be possibly printed next
%%% to the ambitus note head.
%%% Moreover:
%%% - \texttt{pitch} is the absolute pitch of the note
\end{verbatim}
Ambitus engraving logics

Rewrite of the code from @file{lily/ambitus-engraver.cc}.

(define (make-ambitus translator)
  "Build an ambitus object: initialize all the grobs and their relations.

  The Ambitus grob contain all other grobs:
  Ambitus
    |- AmbitusLine
    |- AmbitusNoteHead for upper note
    |- AmbitusAccidental for upper note
    |- AmbitusNoteHead for lower note
    |- AmbitusAccidental for lower note

  The parent of an accidental is the corresponding note head, and
  the accidental is set as the 'accidental-grob of the note head
  so that is printed by the function that prints notes."

  ;; make the ambitus object
  (let ((ambitus (make <ambitus>)))
    ;; build the Ambitus grob, which will contain all other grobs
    (set! (ambitus-group ambitus) (ly:engraver-make-grob translator 'Ambitus '()))
    ;; build the AmbitusLine grob (line between lower and upper note)
    (set! (ambitus-line ambitus) (ly:engraver-make-grob translator 'AmbitusLine '()))
    ;; build the upper and lower AmbitusNoteHead and AmbitusAccidental
    (for-each (lambda (direction)
        (let ((head (ly:engraver-make-grob translator 'AmbitusNoteHead '()))
            (accidental (ly:engraver-make-grob translator 'AmbitusAccidental '()))
            (group (ambitus-group ambitus)))
          ;; The parent of the AmbitusAccidental grob is the
          ;; AmbitusNoteHead grob
          (set! (ly:grob-parent accidental Y) head)
          ;; The AmbitusAccidental grob is set as the accidental-grob
          ;; object of the AmbitusNoteHead. This is later used by the
          ;; function that prints notes.
          (set! (ly:grob-object head 'accidental-grob) accidental)
        ;; both the note head and the accidental grobs are added
        ;; to the main ambitus grob.
(ly:axis-group-interface::add-element group head)
(ly:axis-group-interface::add-element group accidental)
;; the note head and the accidental grobs are added to the
;; ambitus object
(set! (ambitus-note-head (ambitus-note ambitus direction))
  head)
(set! (ambitus-note-accidental (ambitus-note ambitus direction))
  accidental))

(list DOWN UP))
;; The parent of the ambitus line is the lower ambitus note head
(set! (ly:grob-parent (ambitus-line ambitus) X)
  (ambitus-note-head (ambitus-note ambitus DOWN)))
;; the ambitus line is added to the ambitus main grob
(ly:axis-group-interface::add-element (ambitus-group ambitus) (ambitus-line ambitus))

#(define-method (initialize-ambitus-state (ambitus <ambitus>) translator)
  "Initialize the state of @var{ambitus}, by getting the starting
  position of middle C and key signature from @var{translator}'s context."
  (if (not (ambitus-start-c0 ambitus))
    (begin
      (set! (ambitus-start-c0 ambitus)
        (ly:context-property (ly:translator-context translator)
          'middleCPosition
          0))
      (set! (ambitus-start-key-sig ambitus)
        (ly:context-property (ly:translator-context translator)
          'keyAlterations))))

#(define-method (update-ambitus-notes (ambitus <ambitus>) note-grob)
  "Update the upper and lower ambitus pithes of @var{ambitus}, using
  @var{note-grob}."
  ;; Get the event that caused the note-grob creation
  ;; and check that it is a note-event.
  (let ((note-event (ly:grob-property note-grob 'cause)))
    (if (ly:in-event-class? note-event 'note-event)
      ;; get the pitch from the note event
      (let ((pitch (ly:event-property note-event 'pitch)))
        ;; if this pitch is lower than the current ambitus lower
        ;; note pitch (or it has not been initialized yet),
        ;; then this pitch is the new ambitus lower pitch,
        ;; and conversely for upper pitch.
        (for-each (lambda (direction pitch-compare)
          (if (or (not (ambitus-note-pitch (ambitus-note ambitus direction)))
            (pitch-compare pitch
              (ambitus-note-pitch (ambitus-note ambitus direction)))
            (begin
              (set! (ambitus-note-pitch (ambitus-note ambitus direction))
                pitch)
              (set! (ambitus-note-cause (ambitus-note ambitus direction))
                note-event))))
        (list DOWN UP)))
      (list DOWN UP)))

)
(list ly:pitch<? (lambda (p1 p2)
  (ly:pitch<? p2 p1)))))

#(define-method (typeset-ambitus (ambitus <ambitus>) translator)
  "Typeset the ambitus:
  - place the lower and upper ambitus notes according to their pitch and
    the position of the middle C;
  - typeset or delete the note accidentals, according to the key signature.
    An accidental, if it is to be printed, is added to an AccidentalPlacement
    grob (a grob dedicated to the placement of accidentals near a chord);
  - both note heads are added to the ambitus line grob, so that a line should
    be printed between them."
  ;; check if there are lower and upper pitches
  (if (and (ambitus-note-pitch (ambitus-note ambitus UP))
           (ambitus-note-pitch (ambitus-note ambitus DOWN)))
    ;; make an AccidentalPlacement grob, for placement of note accidentals
    (let ((accidental-placement (ly:engraver-make-grob
                                  translator
                                  'AccidentalPlacement
                                  (ambitus-note-accidental (ambitus-note ambitus DOWN))))))
    ;; For lower and upper ambitus notes:
    (for-each (lambda (direction)
               (let ((pitch (ambitus-note-pitch (ambitus-note ambitus direction))))
                 ;; set the cause and the staff position of the ambitus note
                 ;; according to the associated pitch
                 (set! (ly:grob-property (ambitus-note-head (ambitus-note ambitus direction))
                         'cause)
                       (ambitus-note-cause (ambitus-note ambitus direction)))
                 (set! (ly:grob-property (ambitus-note-head (ambitus-note ambitus direction))
                         'staff-position)
                       (+ (ambitus-start-c0 ambitus)
                          (ly:pitch-steps pitch))))
    ;; determine if an accidental shall be printed for this note,
    ;; according to the key signature
    (let* ((handle (or (assoc (cons (ly:pitch-octave pitch)
                                 (ly:pitch-notename pitch))
                         (ambitus-start-key-sig ambitus))
                       (assoc (ly:pitch-notename pitch)
                              (ambitus-start-key-sig ambitus))))
      (sig-alter (if handle (cdr handle) 0))
      (cond ((= (ly:pitch-alteration pitch) sig-alter)
             ;; the note alteration is in the key signature
             ;; => it does not have to be printed
             (ly:grob-suicide!
              (ambitus-note-accidental (ambitus-note ambitus direction))))
            (set! (ly:grob-object (ambitus-note-head (ambitus-note ambitus direction)))
                  'accidental-grob)
                  '()))
          (else
           ;; otherwise, the accidental shall be printed
           (set! (ly:grob-property (ambitus-note-accidental
                                   (ambitus-note ambitus direction)))
                 '())))
'alteration)
  (ly:pitch-alteration pitch))))

;; add the AccidentalPlacement grob to the conditional items of the AmbitusNoteHead
(ly:separation-item::add-conditional-item
  (ambitus-note-head (ambitus-note ambitus direction))
  accidental-placement)

;; add the AmbitusAccidental to the list of the ;; AccidentalPlacement grob accidentals
(ly:accidental-placement::add-accidental
  (ambitus-note-accidental (ambitus-note ambitus direction)))

;; add the AmbitusNoteHead to the list of thelist
 NOTE HEADS
 (ambitus-note-head (ambitus-note ambitus direction)))))

(ly:pointer-group-interface::add-grob
  (ambitus-line ambitus)
  'note-heads
  (ambitus-note-head (ambitus-note ambitus direction))))

(list DOWN UP))

;; add the AccidentalPlacement grob to the main Ambitus grob
(ly:axis-group-interface::add-element (ambitus-group ambitus) accidental-placement)

;; no notes ==> suicide the grobs
(begin
  (for-each (lambda (direction)
    (ly:grob-suicide! (ambitus-note-accidental (ambitus-note ambitus direction)))
    (ly:grob-suicide! (ambitus-note-head (ambitus-note ambitus direction))))
  (list DOWN UP))
  (ly:grob-suicide! ambitus-line))))

%%% Ambitus engraver definition
%%% #define ambitus-engraver

(lambda (context)
  (let ((ambitus #f))
    ;; when music is processed: make the ambitus object, if not already built
    (make-engraver
      ((process-music translator)
        (if (not ambitus)
          (set! ambitus (make-ambitus translator)))
          ;; set the ambitus clef and key signature state
        ((stop-translation-timestep translator)
          (if ambitus
            (initialize-ambitus-state ambitus translator)))
            ;; when a note-head grob is built, update the ambitus notes
        (acknowledgers
          ((note-head-interface engraver grob source-engraver)
            (if ambitus
              (update-ambitus-notes ambitus grob))))
              ;; finally, typeset the ambitus according to its upper and lower notes
            (if any).
        (finalize translator)
          (if ambitus
Displaying a whole GrandStaff system if only one of its staves is alive

In orchestral scores sometimes single or groups of instruments are silent for a while and their staves can be removed for that time (with \removeEmptyStaves).

When they play again it is often preferred to show the staves of all instruments of such a group. This can be done adding the \Keep_alive_together_engraver in the grouper (e.g., a GrandStaff or a StaffGroup).

In the example the violins are silent in the 2nd system and in the 3rd system. Only the first violin plays the last measure but the staff of the second violin is also displayed.
\relative c'' { 
  a1 \repeat unfold 7 { s1 } \repeat unfold 12 a16 a4 
} 
\new Staff = "StaffViolinII" \with { 
  instrumentName = "Violin II" 
  shortInstrumentName = "Vi II" 
} 
\relative c' { e1 \repeat unfold 8 { s1 } } 

\new Staff = "Staff_cello" \with { 
  instrumentName = "Cello" 
  shortInstrumentName = "Ce" 
} 
\relative c { \clef bass \repeat unfold 9 { c1 } }

\layout { 
  indent = 3.0\cm 
  short-indent = 1.5\cm 
  \context { 
    \GrandStaff 
    \consists Keep_alive_together_engraver 
  } 
  \context { 
    \Staff 
    \RemoveEmptyStaves 
  } 
}
Engravers one-by-one

The notation problem, creating a certain symbol, is handled by plugins. Each plugin is called an Engraver. In this example, engravers are switched on one by one, in the following order:

- note heads,
- staff symbol,
- clef,
- stem,
- beams, slurs, accents,
- accidentals, bar lines, time signature and key signature.

Engravers are grouped. For example, note heads, slurs, beams etc. form a Voice context. Engravers for key signature, accidentals, bar line, etc. form a Staff context.

%%% sample music

`topVoice = \relative c' {`
\key d \major
es8([ g] a[ fis])
b4
b16[- . b-. b-. cis-.]
d4->
}

`botVoice = \relative c' {`
\key d \major
c8([ f] b[ a])
es4
es16[- . es-. es-. fis-.]
b4->
}

`hoom = \relative c {
\key d \major
\clef bass
g8-. r
r4`
\relative c' {
  r8 b-.
  r4
  r8 g-.
  r16 g-. r8
  \clef treble
  fis'4->
}

%/ setup for Request->Element conversion. Guru-only%
%
MyStaff = \context {
  \type "Engraver_group"
  \name Staff

  \description "Handles clefs, bar lines, keys, accidentals. It can contain @code{Voice} contexts."

  \consists "Output_property_engraver"

  \consists "Font_size_engraver"

  \consists "Volta_engraver"
  \consists "Separating_line_group_engraver"
  \consists "Dot_column_engraver"

  \consists "Ottava_spanner_engraver"
  \consists "Rest_collision_engraver"
  \consists "Piano_pedal_engraver"
  \consists "Piano_pedal_align_engraver"
  \consists "Instrument_name_engraver"
  \consists "Grob_pq_engraver"
  \consists "Forbid_line_break_engraver"
  \consists "Axis_group_engraver"

  \consists "Pitch_squash_engraver"

  localAlterations = #() % explicitly set instrumentName, so we don't get weird effects when doing instrument names for piano staves

  instrumentName = #()
\shortInstrumentName = #'()

\accepts "Voice"
\defaultchild "Voice"
}

MyVoice = \context {
\type "Engraver_group"
\name Voice

\description "Corresponds to a voice on a staff. This context handles the conversion of dynamic signs, stems, beams, super- and subscripts, slurs, ties, and rests.

You have to instantiate this explicitly if you want to have multiple voices on the same staff."

\localAlterations = #'()
\consists "Font_size_engraver"

% must come before all
\consists "Output_property_engraver"
\consists "Arpeggio_engraver"
\consists "Multi_measure_rest_engraver"
\consists "Text_spanner_engraver"
\consists "Grob_pq_engraver"
\consists "Note_head_line_engraver"
\consists "Glissando_engraver"
\consists "Ligature_bracket_engraver"
\consists "Breathing_sign_engraver"
% \consists "Rest_engraver"
\consists "Grace_beam_engraver"
\consists "New_fingering_engraver"
\consists "Chord_tremolo_engraver"
\consists "Percent_repeat_engraver"
\consists "Slash_repeat_engraver"

%
% Must come before text_engraver, but after note_column engraver.
%
\consists "Text_engraver"
\consists "Dynamic_engraver"
\consists "Dynamic_align_engraver"
\consists "Fingering_engraver"

\consists "Script_column_engraver"
\consists "Rhythmic_column_engraver"
\consists "Cluster_spanner_engraver"
\consists "Tie_engraver"
\consists "Tie_engraver"
\texttt{MyStaff} = \texttt{context} { \
  \texttt{MyStaff} \n  \texttt{consists} "Staff\_symbol\_engraver" 
}\n
\texttt{MyStaff} = \texttt{context} { 
  \texttt{MyStaff} \n  \texttt{consists} "Clef\_engraver" 
  \texttt{remove} "Pitch\_squash\_engraver" 
}\n
\texttt{MyVoice} = \texttt{context} { 
  \texttt{MyVoice} \n  \texttt{consists} "Stem\_engraver" 
}
MyVoice = \context {
    \MyVoice
    \consists "Beam_engraver"
}

\score {
    \topVoice
    \layout {
        \context { \MyStaff }
        \context { \MyVoice }
    }
}

MyVoice = \context {
    \MyVoice
    \consists "Phrasing_slur_engraver"
    \consists "Slur_engraver"
    \consists "Script_engraver"
}

\score {
    \topVoice
    \layout {
        \context { \MyStaff }
        \context { \MyVoice }
    }
}

MyStaff = \context {
    \MyStaff
    \consists "Bar_engraver"
    \consists "Time_signature_engraver"
}

\score {
    \topVoice
    \layout {
        \context { \MyStaff }
        \context { \MyVoice }
    }
}

MyStaff = \context {
    \MyStaff
    \consists "Accidental_engraver"
    \consists "Key_engraver"
}
\score {
La formattazione mensurale, in cui le stanghette non appaiono sui righi ma nello spazio tra i righi, si può ottenere usando StaffGroup al posto di ChoirStaff. La stanghetta sui righi viene nascosta con \hide.

\layout {
  \context {
    \Staff
      \measureBarType = "-span|
  }
}

\music = \fixed { {c'} {c1}
Annidare i righi

Si può usare la proprietà systemStartDelimiterHierarchy per creare gruppi di righi annidati più complessi. Il comando \set StaffGroup.systemStartDelimiterHierarchy prende come argomento una lista alfabetica dell’insieme di righi prodotti. Prima di ogni rigo si può assegnare un delimitatore di inizio del sistema. Deve essere racchiuso tra parentesi e collega tutti i righi compresi tra le parentesi. Gli elementi nella lista possono essere omessi, ma la prima parentesi quadra collega sempre tutti i righi. Le possibilità sono SystemStartBar, SystemStartBracket, SystemStartBrace e SystemStartSquare.

\new StaffGroup
\relative c'' <<
 \override StaffGroup.SystemStartSquare.collapse-height = #4
 \set StaffGroup.systemStartDelimiterHierarchy
   = #'(SystemStartSquare (SystemStartBrace (SystemStartBracket a
   (SystemStartSquare b) ) c ) d)
 \new Staff { c1 }
 \new Staff { c1 }
 \new Staff { c1 }
 \new Staff { c1 }
 \new Staff { c1 }

>>
Numbering groups of measures

This snippet demonstrates the use of the `Measure_counter_engraver` to number groups of successive measures. Any stretch of measures may be numbered, whether consisting of repetitions or not.

The engraver must be added to the appropriate context. Here, a Staff context is used; another possibility is a Dynamics context.

The counter is begun with `\startMeasureCount` and ended with `\stopMeasureCount`. Numbering will start by default with 1, but this behavior may be modified by overriding the `count-from` property.

When a measure extends across a line break, the number will appear twice, the second time in parentheses.

```
\layout {
  \context {  
    \Staff  
    \consists #Measure_counter_engraver 
  }
}

\new Staff {  
  \startMeasureCount 
  \repeat unfold 7 { 
    c'4  d'  e'  f' 
  }  
  \stopMeasureCount  
  \bar "||"  
  g'4  f'  e'  d'  
  \override Staff.MeasureCounter.count-from = #2  
  \startMeasureCount  
  \repeat unfold 5 {  
    g'4  f'  e'  d' 
  }  
  g'4  f' 
```
Print ChordNames with same root and different bass as slash and bass-note

To print subsequent ChordNames only differing in its bass note as slash and bass note use the here defined engraver. The behaviour may be controlled in detail by the chordChanges context property.

```ly
#(define Bass_changes_equal_root_engraver
  (lambda (ctx)
    "For sequential @code{ChordNames} with same root, but different bass, the root markup is dropped: D D/C D/B -> D /C /B
    The behaviour may be controlled by setting the @code{chordChanges} context-property."
    (let ((chord-pitches '())
        (last-chord-pitches '())
        (bass-pitch #f))
      (make- engraver
        ((initialize this-engraver)
         (let ((chord-note-namer (ly:context-property ctx 'chordNoteNamer)))
          ;; Set 'chordNoteNamer, respect user setting if already done
          (ly:context-set-property! ctx 'chordNoteNamer
            (if (procedure? chord-note-namer)
                chord-note-namer
                note-name->markup))))
      (listeners
        ((note-event this-engraver event)
         (let* ((pitch (ly:event-property event 'pitch))
                (pitch-name (ly:pitch-notename pitch))
                (pitch-alt (ly:pitch-alteration pitch))
                (bass (ly:event-property event 'bass #f))
                (inversion (ly:event-property event 'inversion #f)))))
```
Collect notes of the chord
- to compare inversed chords we need to collect the bass note
  as usual member of the chord, whereas an added bass must be
  treated separate from the usual chord-notes
- notes are stored as pairs containing their
  pitch-name (an integer), i.e. disregarding their octave and
  their alteration
\begin{verbatim}
(cond (bass (set! bass-pitch pitch))
  (inversion
   (set! bass-pitch pitch)
   (set! chord-pitches
    (cons (cons pitch-name pitch-alt) chord-pitches)))
  (else
   (set! chord-pitches
    (cons (cons pitch-name pitch-alt) chord-pitches))))
\end{verbatim}

\begin{verbatim}
(acknowledgers
 (let ((chord-changes (ly:context-property ctx 'chordChanges #f)))
   ;; If subsequent chords are equal apart from their bass,
   ;; reset the 'text-property.
   ;; Equality is done by comparing the sorted lists of this chord's
   ;; elements and the previous chord. Sorting is needed because
   ;; inverted chords may have a different order of pitches.
   ;; 'chord-changes' needs to be true
   (if (and bass-pitch chord-changes
     (equal?
      (sort chord-pitches car<)
      (sort last-chord-pitches car<)))
     (ly:grob-set-property! grob 'text
      (make-line-markup
       (list
        (ly:context-property ctx 'slashChordSeparator)
        ((ly:context-property ctx 'chordNoteNamer) bass-pitch
         (ly:context-property ctx 'chordNameLowercaseMinor))))))
     (set! last-chord-pitches chord-pitches)
     (set! chord-pitches '())
     (set! bass-pitch #f)))))
(acknowledgers
 ( finalize this-engraver)
 (set! last-chord-pitches '()))))
\end{verbatim}

myChords = \chordmode {
  %GermanChords

  \set chordChanges = ##t
  d2:m d:m/cis
  d:m/c
  \set chordChanges = ##f
  d:m/b
Togliere i numeri di battuta da uno spartito

I numeri di battuta possono essere tolti rimuovendo l’incisore Bar_number_engraver dal contesto Score.

\layout{
\context{\Score%
\omit BarNumber
\% or:
\% remove "Bar_number_engraver"
}
}

\relative c' { c4 c c c \break
c4 c c c}
Usare una parentesi quadra all’inizio di un gruppo di righi

Si può usare il segno SystemStartSquare (uno dei segni che delimitano l’inizio del sistema) impostandolo esplicitamente in un contesto StaffGroup o ChoirStaff.

```latex
\score {
  \new StaffGroup { <<
    \set StaffGroup.systemStartDelimiter = #'SystemStartSquare
    \new Staff { c'4 d' e' f' }
    \new Staff { c'4 d' e' f' }
  >> }
}
```

Using marklines in a Frenched score

Using MarkLine contexts (such as in LSR1010 (https://lsr.di.unimi.it/LSR/Item?id=1010)) in a Frenched score can be problematic if all the staves between two MarkLines are removed in one system. The Keep_alive_together_engraver can be used within each StaffGroup to keep the MarkLine alive only as long as the other staves in the group stay alive.

```latex
bars = {
  \tempo "Allegro" 4=120
  s1*2
  \repeat unfold 5 { \mark \default s1*2 }
  \bar "||"
  \tempo "Adagio" 4=40
  s1*2
  \repeat unfold 8 { \mark \default s1*2 }
  \bar ".".
}
winds = \repeat unfold 120 { c''4 }
trumpet = { \repeat unfold 8 g'2 R1*16 \repeat unfold 4 g'2 R1*8 }
trombone = { \repeat unfold 4 c'1 R1*8 d'1 R1*17 }
strings = \repeat unfold 240 { c''8 }

#(set-global-staff-size 16)
\paper {
  systems-per-page = 5
  ragged-last-bottom = ##f
}
\layout {
```
indent = 15\ mm
short-indent = 5\ mm
\context {  
  \name MarkLine
  \type Engraver_group
  \consists Output_property_engraver
  \consists Axis_group_engraver
  \consists Mark_engraver
  \consists Metronome_mark_engraver
  \consists Staff_collecting_engraver
  \override VerticalAxisGroup.remove-empty = ##t
  \override VerticalAxisGroup.remove-layer = '#any
  \override VerticalAxisGroup.staff-affinity = #DOWN
  \override VerticalAxisGroup.nonstaff-relatedstaff-spacing.padding = 1
  keepAliveInterfaces = #'()
}
\context {  
  \Staff
  \override VerticalAxisGroup.remove-empty = ##t
  \override VerticalAxisGroup.remove-layer = ##f
}
\context {  
  \StaffGroup
  \accepts MarkLine
  \consists Keep_alive_together_engraver
}
\context {  
  \Score
  \remove Mark_engraver
  \remove Metronome_mark_engraver
  \remove Staff_collecting_engraver
}
}

\score {  
  <<  
  \new StaffGroup = "winds" \with {  
    instrumentName = "Winds"
    shortInstrumentName = "Winds"
  } <<
  \new MarkLine \bars
  \new Staff \winds
  >>  
  \new StaffGroup = "brass" <<  
  \new MarkLine \bars
  \new Staff = "trumpet" \with {  
    instrumentName = "Trumpet"
    shortInstrumentName = "Tpt"
  } \trumpet
  \new Staff = "trombone" \with {  
    instrumentName = "Trombone"
    shortInstrumentName = "Tbn"
  } \trombone
}
Modello per gruppo vocale con testo allineato sotto e sopra i righi

Questo modello è fondamentalmente analogo al semplice modello “Complesso vocale”, con l’unica differenza che qui tutti i versi del testo sono posizionati usando alignAboveContext e alignBelowContext.

global = {
    \key c \major
    \time 4/4
}

sopMusic = \relative c’ { c4 c c8[[ b]]} c4

sopWords = \lyricmode {
    hi hi hi hi
}

altoMusic = \relative c’ { e4 f d e }

altoWords = \lyricmode {
    ha ha ha ha
}

tenorMusic = \relative c’ { g4 a f g }

tenorWords = \lyricmode {
    hu hu hu hu
}

bassMusic = \relative c { c4 c g c }
Modello per gruppo vocale con strofa e ritornello

Questo modello crea una partitura che inizia con una sezione solistica e prosegue in un ritornello a due voci. Illustra anche l’uso delle pause spaziatrici all’interno della variabile \global per definire i cambi di tempo (e altri elementi comuni a tutte le parti) nel corso di tutta la partitura.

global = {
  \key g \major
}
\% verse
\time 3/4
s2.*2
\break

\% refrain
\time 2/4
s2*2
\bar "|.
}

SoloNotes = \relative g' { 
  \clef "treble"

  \% verse
  g^4 g g |
  b4 b b |

  \% refrain
  R2*2 |
}

SoloLyrics = \lyricmode { 
  One two three |
  four five six |
}

SopranoNotes = \relative c'' { 
  \clef "treble"

  \% verse
  R2.*2 |

  \% refrain
  c^4 c |
  g^4 g |
}

SopranoLyrics = \lyricmode { 
  la la |
  la la |
}

BassNotes = \relative c { 
  \clef "bass"

  \% verse
  R2.*2 |

  \% refrain
  c^4 e |

BassLyrics = \lyricmode {
    dum dum |
    dum dum |
}

\score {
    <<
        \new Voice = "SoloVoice" << \global \SoloNotes >>
        \new Lyrics \lyricsto "SoloVoice" \SoloLyrics
        \new ChoirStaff <<
            \new Voice = "SopranoVoice" << \global \SopranoNotes >>
            \new Lyrics \lyricsto "SopranoVoice" \SopranoLyrics
            \new Voice = "BassVoice" << \global \BassNotes >>
            \new Lyrics \lyricsto "BassVoice" \BassLyrics
        >>
    >>
    \layout {
        \ragged-right = ##t
        \context { \Staff
            \% these lines prevent empty staves from being printed
            \RemoveEmptyStaves
            \override VerticalAxisGroup.remove-first = ##t
        }
    }
}

\begin{music}
\new Staff
\relative c' { g^ \t \a^ \t \b^ \t 
\\c^ \t \d ^ \t \e^ \t 
\\f^ \t \g^ \t 
\\a^ \t \b^ \t \c^ \t 
\\d ^ \t \e^ \t 
\\f ^ \t \g ^ \t 
\\a ^ \t \b ^ \t 
\\c ^ \t \d ^ \t 
\\e ^ \t \f ^ \t 
\\g ^ \t \a ^ \t
\}
\new Staff
\relative c' { g \t \a \t \b 
\\c \t \d \t \e 
\\f \t \g 
\\a \t \b \t \c 
\\d \t \e \t 
\\f \t \g \t 
\\a \t \b \t 
\\c \t \d \t 
\\e \t \f \t 
\\g \t \a \t
\}
\new Staff
\relative c' { g \t \a \t \b 
\\c \t \d \t \e 
\\f \t \g 
\\a \t \b \t \c 
\\d \t \e \t 
\\f \t \g \t 
\\a \t \b \t 
\\c \t \d \t 
\\e \t \f \t 
\\g \t \a \t
\}
\new Staff
\relative c' { g \t \a \t \b 
\\c \t \d \t \e 
\\f \t \g 
\\a \t \b \t \c 
\\d \t \e \t 
\\f \t \g \t 
\\a \t \b \t 
\\c \t \d \t 
\\e \t \f \t 
\\g \t \a \t
\}
\end{music}

One two three four five six

la la la la

dum dum dum dum
Tweaks and overrides

Sezione “Changing defaults” in Guida alla Notazione
Sezione “Tweaking output” in Manuale di Apprendimento

Aggiungere un segno di ottava a una sola voce

Se il rigo ha più di una voce, l’ottavazione in una voce trasporrà la posizione delle note in tutte le voci per la durata della parentesi dell’ottava. Se si intende applicare l’ottavazione a una sola voce, occorre spostare l’incisore Ottava_spanner_engraver nel contesto Voice.

\layout {
  \context {  
    \Staff 
    \remove Ottava_spanner_engraver 
  }  
  \context { 
    \Voice 
    \consists Ottava_spanner_engraver 
  }  
}

{  
  \clef bass  
  << { <g d'>1~ q2 <c' e'> } 
  \\  
  { r2. \ottava -1  
    <b,,, b,,>4 ~ |  
    q2 \ottava 0  
    <c e>2  
  }  
  >>  
}

Adding links to objects

To add a link to a grob stencil you can use add-link as defined here. It works both with \override and \tweak.

Drawback: point-and-click is disturbed for the linked grobs.
Limitation: Works for PDF only.
The linked objects are colored with a separate command. Note that the links are not displayed and are not clickable from inside the LSR.

#(define (add-link url-strg)
(lambda (grob)
  (let* ((stil (ly:grob-property grob 'stencil))
           (if (ly:stencil? stil)
               (let* ((x-ext (ly:stencil-extent stil X))
                       (y-ext (ly:stencil-extent stil Y))
                       (url-expr `(url-link ,url-strg ,x-ext ,y-ext))
                       (new-stil
                         (ly:stencil-add
                          (ly:make-stencil url-expr x-ext y-ext)
                          stil))
                       (ly:grob-set-property! grob 'stencil new-stil))))

%%% test

%%% For easier maintenance of this snippet the URL is formatted to use the
%%% actually used LilyPond version.
%%% Of course a literal URL would work as well.

#(define major.minor-version
  (string-join (take (string-split (lilypond-version) #\.) 2) "."))

urlI =
#(format #f
  "http://lilypond.org/doc/v~a/Documentation/notation/writing-pitches"
  major.minor-version)

urlII =
#(format #f
  "http://lilypond.org/doc/v~a/Documentation/notation/rhythms"
  major.minor-version)

urlIII =
#(format #f
  "http://lilypond.org/doc/v~a/Documentation/notation/note-heads"
  major.minor-version)

urlIV =
#(format #f
  "http://lilypond.org/doc/v~a/Documentation/notation/beams"
  major.minor-version)

urlV =
#(format #f
  "http://lilypond.org/doc/v~a/Documentation/notation/note-head-styles"
  major.minor-version)

urlVI =
#(format #f
  "http://lilypond.org/doc/v~a/Documentation/notation/writing-pitches"
  major.minor-version)

\relative c' {
Adding markups in a tablature

By default markups does not show in a tablature. To make them appear, simply use the command \revert TabStaff.TextScript.stencil

```
\key cis \minor

\once \override Staff.Clef.color = #green
\once \override Staff.Clef.after-line-breaking =
  #(add-link urlI)

\once \override Staff.TimeSignature.color = #green
\once \override Staff.TimeSignature.after-line-breaking =
  #(add-link urlII)

\once \override NoteHead.color = #green
\once \override NoteHead.after-line-breaking =
  #(add-link urlIII)

cis'1
\once \override Beam.color = #green
\once \override Beam.after-line-breaking =
  #(add-link urlIV)

cis8 dis e fis gis2
<gis,
\tweak Accidental.color #green
\tweak Accidental.after-line-breaking #(add-link urlV)
\tweak color #green
\tweak after-line-breaking #(add-link urlVI)
\tweak style #'harmonic
bis
dis
fis
>1
<cis, cis' e>
```

```
\define StaffStyle
\define TimeSignatureStyle
\define NoteStyle
\define KeySignatureStyle
\define BeamStyle
\define AccidentalStyle
\define Color
\define AfterLineBreaking
\define TabStyle
\define TabColor
\define TabAfterLineBreaking
\define StaffStyles
\define TimeSignatureStyles
\define NoteStyles
\define KeySignatureStyles
\define BeamStyles
\define AccidentalStyles
\define Colors
\define AfterLineBreakings
\define TabStyles
\define TabColors
\define TabAfterLineBreakings

\score {
  \new TabStaff {
    \repeat unfold 2 << \high \low \pulse >>}
```
Aggiungere i segni di tempo per i glissandi lunghi

I battiti saltati nei glissandi molto lunghi vengono talvolta segnalati con delle indicazioni di tempo, che consistono solitamente in dei gambi privi di teste di nota. Questi gambi possono essere usati anche per contenere segni di espressione intermedi.

Se i gambi non si allineano bene al glissando, può essere necessario riposizionarli leggermente.

\texttt{glissandoSkipOn} = {
  \texttt{override NoteColumn.glissando-skip = ##t}
  \texttt{hide NoteHead}
  \texttt{override NoteHead.no-ledgers = ##t}
}

\texttt{glissandoSkipOff} = {
  \texttt{revert NoteColumn.glissando-skip}
  \texttt{undo hide NoteHead}
  \texttt{revert NoteHead.no-ledgers}
}

\texttt{relative c''} {
  r8 f8\glissando
  \texttt{glissandoSkipOn}
  f4 g a a8\noBeam
  \texttt{glissandoSkipOff}
  a8

  r8 f8\glissando
  \texttt{glissandoSkipOn}
  g4 a8
  \texttt{glissandoSkipOff}
a8 |
| r4 f   \glissando \<
\glissandoSkipOn
a4/\f \>
\glissandoSkipOff
b8\! r |

Adjusting grace note spacing

The space given to grace notes can be adjusted using the spacing-increment property of Score.GraceSpacing.

```latex
\graceNotes = {
\grace { c4 c8 c16 c32 }
c8
}
```

```latex
\relative c' {
\graceNotes
\override Score.GraceSpacing.spacing-increment = #2.0
\graceNotes
\revert Score.GraceSpacing.spacing-increment
\graceNotes
}
```

Adjusting lyrics vertical spacing

This snippet shows how to bring the lyrics line closer to the staff.

```latex
% Default layout:
<<
\new Staff \new Voice = melody \relative c' {
 c4 d e f
g4 f e d
c1
}
\new Lyrics \lyricsto melody { aa aa aa aa aa aa aa aa }
```

```latex
\new Staff {
\new Voice = melody \relative c' {
 c4 d e f
g4 f e d
```
Altering the length of beamed stems

Stem lengths on beamed notes can be varied by overriding the beamed-lengths property of the details of the Stem. If a single value is used as an argument, the length applies to all stems. When multiple arguments are used, the first applies to eighth notes, the second to sixteenth notes and so on. The final argument also applies to all notes shorter than the note length of the final argument. Non-integer arguments may also be used.

\relative c' \{
  \override Stem.details.beamed-lengths = #'(2)
  a8[ a] a16[ a] a32[ a]
  \override Stem.details.beamed-lengths = #'(8 10 12)
  a8[ a] a16[ a] a32[ a] r8
  \override Stem.details.beamed-lengths = #'(8)
  a8[ a]
  \override Stem.details.beamed-lengths = #'(8.5)
  a8[ a]
  \revert Stem.details.beamed-lengths
  a8[ a] a16[ a] a32[ a] r16
}\n
Numeri di battuta alternativi

Si possono impostare due metodi alternativi di numerazione della battuta, utili specialmente per le ripetizioni.

\relative c' \{
  \set Score.alternativeNumberingStyle = #'numbers
\repeat volta 3 { c4 d e f | }
\alternative {
{ c4 d e f | c2 d \break }
{ f4 g a b | f4 g a b | f2 a | \break }
{ c4 d e f | c2 d }
}
\set Score.alternativeNumberingStyle = #'numbers-with-letters
\repeat volta 3 { c,4 d e f | }
\alternative {
{ c4 d e f | c2 d \break }
{ f4 g a b | f4 g a b | f2 a | \break }
{ c4 d e f | c2 d }
}
\break
\set Score.alternativeNumberingStyle = #'numbers-with-letters

Parentesi analitiche sopra il rigo

Delle semplici parentesi analitiche orizzontali vengono aggiunte, per impostazione predefinita, sotto il rigo. L’esempio seguente mostra un modo per posizionarle sopra il rigo.

\layout {
\context {
\Voice
\consists "Horizontal_bracket_engraver"
Parentesi analitiche con etichette

Si può aggiungere del testo alle parentesi analitiche tramite la proprietà `text` del grob `HorizontalBracketText`. L’aggiunta di vari frammenti di testo alle parentesi che iniziano nello stesso momento musicale richiede l’uso del comando `\tweak`. Dopo un’interruzione di linea il testo viene messo tra parentesi.

```
paper { tagline = ##f }

layout {
  context {
    Voice
    consists "Horizontal_bracket_engraver"
    \override HorizontalBracket.direction = #UP
  }
}

\once \override HorizontalBracketText.text = "a"
c\"\startGroup d\"\stopGroup
\once \override HorizontalBracketText.text = "a'"
e\"\startGroup d\"\stopGroup |
c\"\tweak HorizontalBracketText.text
\markup \bold \huge "b" \startGroup
-\tweak HorizontalBracketText.text "a" \startGroup
d\"\stopGroup
e\"\tweak HorizontalBracketText.text "a'" \startGroup
d\"\stopGroup\stopGroup |
c\"\tweak HorizontalBracketText.text foo \startGroup
d\" e\" f\" | \break
g\" a\" b\" c\"\stopGroup
```
Asymmetric slurs

Slurs can be made asymmetric to match an asymmetric pattern of notes better.

```
\relative c' {  
  \stemDown  
  \slurUp  
  \slurNotes  
  \once \override Slur.eccentricity = #3.0  
  \slurNotes  
}
```

Caesura ("railtracks") with fermata

A caesura is sometimes denoted by a double “railtracks” breath mark with a fermata sign positioned above. This snippet shows an optically pleasing combination of railtracks and fermata.

```
\relative c'' {  
  c2.  
  \override BreathingSign.text = \markup {  
    \override #'(direction . 1)  
    \override #'(baseline-skip . 1.8)  
    \dir-column {  
      \translate #'(0.155 . 0)  
      \center-align \musicglyph "scripts.caesura.curved"  
      \center-align \musicglyph "scripts.ufermata"  
    }  
  }  
  \breathe c4  
  \override BreathingSign.text  
  c2.  \breathe c4  
  \bar 
}
```
Changing a single note’s size in a chord

Individual note heads in a chord can be modified with the `\tweak` command inside a chord, by altering the `font-size` property.

Inside the chord (within the brackets `< >`), before the note to be altered, place the `\tweak` command, followed by `font-size` and define the proper size like `-2` (a tiny note head).

```plaintext
relative c' {
  \tweak font-size +2 c e g c
  \tweak font-size -2 e>1
  \markup { A tiny e } \markup { A big c }
}
```

Changing beam thickness and spacing

To make beams thicker or thinner alter the `Beam.beam-thickness` property. To adjust the spacing between beams alter the `Beam.length-fraction` property.

```plaintext
relative f' {
  \time 1/8
  \override Beam.beam-thickness = #0.4
  \override Beam.length-fraction = #0.8
  c32 c c c
  \revert Beam.beam-thickness \% 0.48 is default thickness
  \revert Beam.length-fraction \% 1.0 is default spacing
  c32 c c c
  \override Beam.beam-thickness = #0.6
  \override Beam.length-fraction = #1.3
  c32 c c c
}
```

Modificare la forma delle pause multiple

Se la pausa multipla dura dieci misure o un numero inferiore a dieci, nel rigo apparirà una serie di pause di longa e di breve (chiamate in tedesco “Kirchenpausen” - pause ecclesiastiche); altrimenti apparirà una semplice linea. Il numero predefinito di dieci può essere cambiato sovrascrivendo la proprietà `expand-limit`.

```plaintext
\relative c'' {
  \compressMMRests {
    R1*2 | R1*5 | R1*9
    \override MultiMeasureRest.expand-limit = #3
    R1*2 | R1*5 | R1*9
  }
}
```
Changing properties for individual grobs

The \applyOutput command allows the tuning of any layout object, in any context. It requires a Scheme function with three arguments.

\[\texttt{#(define (mc-squared grob grob-origin context)}\]
\[\texttt{(let ((sp (ly:grob-property grob 'staff-position)))}\]
\[\texttt{(ly:grob-set-property! grob 'stencil}\]
\[\texttt{(grob-interpret-markup grob}\]
\[\texttt{#{ \markup \lower #0.5}}\]
\[\texttt{\case sp}\]
\[\texttt{((-5) "m")}\]
\[\texttt{((-3) "c ")}\]
\[\texttt{((-2) #\{ \markup \teny \bold 2 #\})}\]
\[\texttt{(else "bla")) #}})\]

\[\texttt{\relative c'} {\}
\[\texttt{<d f g b>2}\]
\[\texttt{\applyOutput Voice.NoteHead #mc-squared}\]
\[\texttt{<d f g b>2}\]

Nascondere la linea di estensione per le dinamiche testuali

Il testo usato per i crescendo e i decrescendo può essere cambiato modificando le proprietà di contesto crescendoText e decrescendoText.

Lo stile della linea dell’estensore può essere cambiato modificando la proprietà 'style di DynamicTextSpanner. Il valore predefinito è 'dashed-line; gli altri valori possibili sono 'line, 'dotted-line e 'none.

\[\texttt{\relative c'' {\}
\[\texttt{\set crescendoText = \markup { \italic { cresc. poco } } \}
\[\texttt{\set crescendoSpanner = #'text \}
\[\texttt{\override DynamicTextSpanner.style = #'dotted-line}\]
\[\texttt{a\< a}\]
\[\texttt{a2 a}\]
\[\texttt{a2 a\mf}\]

\[\texttt{cresc. poco \dots mf}\]
Changing the default text font family

The default font families for text can be overridden with make-pango-font-tree.

{%
You may have to install additional fonts.
%

Red Hat Fedora

dejavu-fonts-all

Debian GNU/Linux, Ubuntu

fonts-dejavu-core
fonts-dejavu-extra
%

\paper {
  % change for other default global staff size.
  myStaffSize = #20
  %{  
    run
      lilypond -dshow-available-fonts
      to show all fonts available in the process log.
    %}
    #(define fonts
      (make-pango-font-tree "DejaVu Serif"
      "DejaVu Sans"
      "DejaVu Sans Mono"
      (/ myStaffSize 20)))
  }
%

{ g''4\markup {  
  DejaVu Serif: \bold bold  
  \italic italic  
  \italic \bold { bold italic }  
}  
}{
  g''2\markup {  
    \override #'(font-family . sans) {  
      DejaVu Sans: \bold bold  
      \italic italic  
      \italic \bold { bold italic }  
    }  
  }
}{
  g''4\markup {  
    \override #'(font-family . typewriter) {  
      DejaVu Sans Mono: \bold bold  
      \italic italic  
      \italic \bold { bold italic }  
    }  
  }
}
Changing the staff size

Though the simplest way to resize staves is to use \( \text{(set-global-staff-size xx)} \), an individual staff’s size can be changed by scaling the properties ‘staff-space and fontSize.

\[
\begin{array}{l}
\text{\new Staff \{}
  \relative c' \{
    \text{\dynamicDown}
    \text{c8\ff c c c c c c c c c c c}
  \}
\}
\end{array}
\]

\[
\begin{array}{l}
\text{\new Staff \with \{}
  \text{\fontSize = \#-3}
  \text{\override StaffSymbol.staff-space = \#(magstep -3)}
\}
\end{array}
\]

\[
\begin{array}{l}
\text{\clef bass}
  \text{c8 c c c c\f c c c c c c c}
\}
\end{array}
\]
Tweaks and overrides

\mid

Changing the text for sustain markings

Staff.pedalSustainStrings can be used to set the text used for pedal down and up. Note that the only valid strings are those found in the list of pedal glyphs - the values used this snippet constitute an exhaustive list.

sustainNotes = { c4\sustainOn d e\sustainOff\sustainOn f\sustainOff }

\relative c' {

\sustainNotes
\set Staff.pedalSustainStrings = #'("P" "P-" "-"")
\sustainNotes
\set Staff.pedalSustainStrings = #'("d" "de" "e")
\sustainNotes
\set Staff.pedalSustainStrings = #'("M" "M-" "-"")
\sustainNotes
\set Staff.pedalSustainStrings = #'("Ped" "*Ped" "*")
\sustainNotes
}

Controlling spanner visibility after a line break

The visibility of spanners which end on the first note following a line break is controlled by the after-line-breaking callback ly:spanner::kill-zero-spanned-time.

For objects such as glissandos and hairpins, the default behaviour is to hide the spanner after a break; disabling the callback will allow the left-broken span to be shown.

Conversely, spanners which are usually visible, such as text spans, can be hidden by enabling the callback.

\paper { ragged-right = ##t }

\relative c'' {

\override Hairpin.to-barline = ##f
\override Glissando.breakable = ##t
% show hairpin
\override Hairpin.after-line-breaking = ##t
% hide text span
\override TextSpanner.after-line-breaking =
  #ly:spanner::kill-zero-spanned-time
e2\<\startTextSpan
Controllo dell’ordine verticale degli script

L’ordine verticale degli script è determinato dalla proprietà ’script-priority. Più il numero è piccolo, più sarà posto vicino alla nota. In questo esempio, il simbolo di diesis (oggetto TextScript) ha prima la priorità più bassa, dunque è posto più in basso nel primo esempio. Nel secondo, il trillo (oggetto Script) ha la priorità più bassa, quindi si trova all’interno. Quando due oggetti hanno la stessa priorità, l’ordine in cui sono inseriti determina quale viene prima.

Controllare la visibilità della parentesi del gruppo irregolare

Il comportamento predefinito relativo alla visibilità della parentesi quadra del gruppo irregolare è di mostrare una parentesi a meno che non ci sia una travatura della stessa lunghezza del gruppo.

Per controllare la visibilità di tale parentesi, si imposta la proprietà ’bracket-visibility su #t (mostra sempre la parentesi), #’if-no-beam (mostra la parentesi solo se non c’è una travatura, che è il comportamento predefinito) o #f (non mostrare mai la parentesi). L’ultima opzione equivale a omettere l’oggetto @code{TupletBracket} dall’output.
Creare un gruppetto ritardato

Creare un gruppetto ritardato, dove la nota più bassa del gruppetto usa l’alterazione, richiede vari \override. La proprietà outside-staff-priority deve essere impostata su #f, perché altrimenti questa avrebbe la precedenza sulla proprietà avoid-slur. Cambiando la frazione 2/3 si aggiusta la posizione orizzontale.

\relative c' {
\override Score.TextMark.non-musical = ##f
\textMark "default" \music
\override TupletBracket.bracket-visibility = #'if-no-beam
\textMark \markup \typewriter "'if-no-beam" \music
\override TupletBracket.bracket-visibility = #t
\textMark \markup \typewriter "#t" \music
\override TupletBracket.bracket-visibility = ##f
\textMark \markup \typewriter "#f" \music
\omit TupletBracket
\textMark \markup \typewriter "omit" \music
}

\relative c'' {
\after 2*2/3 \turn c2( d4) r |
\after 4 \turn c4.( d8)
\after 4
{
\once \set suggestAccidentals = ##t
\once \override AccidentalSuggestion.outside-staff-priority = ##f
\once \override AccidentalSuggestion.avoid-slur = #'inside
\once \override AccidentalSuggestion.font-size = -3
\once \override AccidentalSuggestion.script-priority = -1
\once \hideNotes
cis8\turn \noBeam
}
d4.( e8)
Creating custom key signatures

LilyPond supports custom key signatures. In this example, print for D minor with an extended range of printed flats.

\new \Staff \with { 
  \override \StaffSymbol.line-count = #8 
  \override \KeySignature.flat-positions = #'(\(-7 \ . \ 6\)) 
  \override \KeyCancellation.flat-positions = #'(\(-7 \ . \ 6\)) 
  \override \KeySignature.sharp-positions = #'(\(-6 \ . \ 7\)) 
  \override \KeyCancellation.sharp-positions = #'(\(-6 \ . \ 7\)) 
  \override \Clef.stencil = # (\lambda (grob)(grob-interpret-markup grob) 
    #(\{ 
      \musicglyph "clefs.C" 
      \translate #'(-3 \ . \ -2) 
      \musicglyph "clefs.F" 
    \})) 
  \clefPosition = #3 
  \middleCPosition = #3 
  \middleCClefPosition = #3 
} 

{ \key d\minor 
  f bes, f bes, 
}

Creating double-digit fingerings

Creating fingerings larger than 5 is possible.

\relative c' { 
  c1-10 
  c1-50 
  c1-36 
  c1-29 
}

Creating text spanners

The \startTextSpan and \stopTextSpan commands allow the creation of text spanners as easily as pedal indications or octavations. Override some properties of the TextSpanner object to modify its output.

\paper { \ragged-right = ##f }
Cross-staff chords - beaming problems workaround

Sometimes it is better to use stems from the upper staff for creating cross-staff chords, because no problems with automatic beam collision avoidance then arise. If the stems from the lower staff were used in the following example, it would be necessary to change the automatic beam collision avoidance settings so that it doesn’t detect collisions between staves using \override Staff.Beam.collision-voice-only = ##t

\new PianoStaff <<
\new Staff = up
\relative c' {
<<
{ r4
\override Stem.cross-staff = ##t
}
\override Stem.length = #19 \% this is in half-spaces, 
\% so it makes stems 9.5 staffspaces long 
\override Stem.Y-offset = #-6 \% stems are normally lengthened 
\% upwards, so here we must lower the stem by the amount 
\% equal to the lengthening - in this case (19 - 7) / 2 
\% (7 is default stem length)

e e e }
{s4
\change Staff = "bottom"
\override NoteColumn.ignore-collision = ##t
\c, \c \c
}

\new Staff = bottom 
\relative c' {
\clef bass
\voiceOne
g8 a g a g a g a}
}

Cross staff stems

This snippet shows the use of the Span\_stem\_engraver and \crossStaff to connect stems across staves automatically.

The stem length need not be specified, as the variable distance between noteheads and staves is calculated automatically.

\layout {
\context {
\PianoStaff
\consists "Span\_stem\_engraver"
}
}

{\new PianoStaff <<
\new Staff {
\<b d'\>4 r d'16\> e'8. g8 r\!
e'8 f' g'4 e'2
}
\new Staff {
\clef bass
\voiceOne
\autoBeamOff
**Custodes**

Custodes may be engraved in various styles.

```latex
\layout { ragged-right = \#t }

\new Staff \with { \consists "Custos_engraver" } \relative c' { 
  \override Staff.Custos.neutral-position = #4

  \override Staff.Custos.style = #'hufnagel
  c1^"hufnagel" \break
  <d a' f'>1

  \override Staff.Custos.style = #'medicaea
  c1^"medicaea" \break
  <d a' f'>1

  \override Staff.Custos.style = #'vaticana
  c1^"vaticana" \break
  <d a' f'>1

  \override Staff.Custos.style = #'mensural
  c1^"mensural" \break
  <d a' f'>1
}
```

![Hufnagel](image1)

![Medicaea](image2)

![Vaticana](image3)
Personalizzare la tastiera del diagramma dei tasti


```
\include "predefined-guitar-fretboards.ly"
\storePredefinedDiagram #default-fret-table \chordmode { c' }
  #guitar-tuning
  ":x;1-1-;3-2;3-3;3-4;1-1-;"
```

```
\% shorthand
oo = #'(define-music-function
  (grob-path value)
  (list? scheme?)
  #{ \once \override $grob-path = #value #})

<<
\new ChordNames {
  \chordmode { c1 | c | c | d }
}
\new FretBoards {
  \% Set global properties of fret diagram
  \override FretBoards.FretBoard.size = #'1.2
  \override FretBoard.fret-diagram-details.finger-code = #'in-dot
  \override FretBoard.fret-diagram-details.dot-color = #'white
  \chordmode { c
    \oo FretBoard.size #'1.0
    \oo FretBoard.fret-diagram-details.barre-type #'straight
    \oo FretBoard.fret-diagram-details.dot-color #'black
    \oo FretBoard.fret-diagram-details.finger-code #'below-string
    c
    \oo FretBoard.fret-diagram-details.barre-type #'none
    \oo FretBoard.fret-diagram-details.number-type #'arabic
    \oo FretBoard.fret-diagram-details.orientation #'landscape
    \oo FretBoard.fret-diagram-details.mute-string #'M'
    \oo FretBoard.fret-diagram-details.label-dir #LEFT
    \oo FretBoard.fret-diagram-details.dot-color #'black
    c
    \oo FretBoard.fret-diagram-details.finger-code #'below-string
    \oo FretBoard.fret-diagram-details.dot-radius #0.35
    \oo FretBoard.fret-diagram-details.dot-position #0.5
```
Personalizzare il diagramma dei tasti di tipo markup

Le proprietà del diagramma dei tasti si possono impostare tramite 'fret-diagram-details. Per diagrammi di tipo markup, gli override possono essere applicati all’oggetto Voice.TextScript o direttamente al markup.

```music
\new Voice = \"mel\" {
    \textLengthOn
    % Set global properties of fret diagram
    \override TextScript.size = #\'1.2
    \override TextScript.fret-diagram-details.finger-code = #\'in-dot
    \override TextScript.fret-diagram-details.dot-color = #\'white

    %% C major for guitar, no barre, using defaults
    % terse style
    c'1 \markup { \fret-diagram-terse \"x;3-3;2-2;o;1-1;o;\" } }

    %% C major for guitar, barred on third fret
    % verbose style
    % size 1.0
    % roman fret label, finger labels below string, straight barre
    c'1 \markup { \fret-diagram-verbose \"x;3-3;2-2;o;1-1;o;\" }
```
\begin{verbatim}
(place-fret 1 3 1)
(barre 5 1 3))

\end{verbatim}

%% C major for guitar, barred on third fret
%% verbose style
%% landscape orientation, arabic numbers, M for mute string
%% no barre, fret label down or left, small mute label font
\markup {
  \override #'(fret-diagram-details . (finger-code . below-string)
                   (number-type . arabic)
                   (label-dir . -1)
                   (mute-string . "M")
                   (orientation . landscape)
                   (barre-type . none)
                   (xo-font-magnification . 0.4)
                   (xo-padding . 0.3))) {
    \fret-diagram-verbose #'(mute 6)
    (place-fret 5 3 1)
    (place-fret 4 5 2)
    (place-fret 3 5 3)
    (place-fret 2 5 4)
    (place-fret 1 3 1)
    (barre 5 1 3))
}

%% simple D chord
%% terse style
%% larger dots, centered dots, fewer frets
%% label below string
\markup {
  \override #'(fret-diagram-details . (finger-code . below-string)
                   (dot-radius . 0.35)
                   (dot-position . 0.5)
                   (fret-count . 3))) {
    \fret-diagram-terse "x;x;o;2-1;3-2;2-3;"
}
}

>>
Mostrare la parentesi anche se c’è un solo rigo nel sistema

Se c’è un solo rigo in uno dei tipi di rigo ChoirStaff o StaffGroup, la parentesi e la stanghetta iniziale non appaiono. Si può modificare questo comportamento predefinito sovrascrivendo collapse-height e impostando un valore inferiore al numero di linee del rigo.

Nei contesti PianoStaff e GrandStaff, dove i sistemi iniziano con una parentesi graffa invece di una parentesi quadra, occorre impostare un’altra proprietà, come si vede nel secondo sistema dell’esempio.

```latex
\score {
  \new StaffGroup <<
    \override StaffGroup.SystemStartBracket.collapse-height = #4
    \override Score.SystemStartBar.collapse-height = #4
  \new Staff {
    c'1
  }
} >>
}
\score {
  \new PianoStaff <<
    \override PianoStaff.SystemStartBrace.collapse-height = #4
    \override Score.SystemStartBar.collapse-height = #4
  \new Staff {
    c'1
  }
} >>
}
```

Displaying grob ancestry

When working with grob callbacks, it can be helpful to understand a grob’s ancestry. Most grobs have parents which influence the positioning of the grob. X- and Y-parents influence the horizontal and vertical positions for the grob, respectively. Additionally, each parent may have parents of its own.

Unfortunately, there are several aspects of a grob’s ancestry that can lead to confusion:

- The types of parents a grob has may depend on context.
• For some grobs, the X- and Y-parents are the same.
• A particular ancestor may be related to a grob in multiple ways.
• The concept of generations is misleading.

For example, the System grob can be both parent (on the Y-side) and grandparent (twice on the X-side) to a VerticalAlignment grob.

This macro prints (to the console) a textual representation of a grob’s ancestry.

When called this way:
{ \once \override NoteHead.before-line-breaking = #display-ancestry c }

The following output is generated:

NoteHead
X,Y: NoteColumn
  X: PaperColumn
  X,Y: System
  Y: VerticalAxisGroup
    X: NonMusicalPaperColumn
      X,Y: System
      Y: VerticalAlignment
        X: NonMusicalPaperColumn
          X,Y: System
          Y: System

%% http://lsr.di.unimi.it/LSR/Item?id=622
%% see also http://www.lilypond.org/doc/v2.18/Documentation/snippets/tweaks-and-overrides#tweaks-and-overrides-displaying-grob-ancestry

%% Remark:
%% grob::name is in the source since 2.19.x could be deleted during next LSR-upgrade
#(define (grob::name grob)
  (assq-ref (ly:grob-property grob 'meta) 'name))

#(define (get-ancestry grob)
  (if (not (null? (ly:grob-parent grob X)))
    (list (grob::name grob)
      (get-ancestry (ly:grob-parent grob X))
      (get-ancestry (ly:grob-parent grob Y)))
    (grob::name grob))))

#(define (format-ancestry lst padding)
  (string-append
    (symbol->string (car lst))
    "\n"
    (let ((X-ancestry
      (if (list? (cadr lst))
        (format-ancestry (cadr lst) (+ padding 3))
        (symbol->string (cadr lst))))
    (Y-ancestry
      (if (list? (caddr lst))
        (format-ancestry (caddr lst) (+ padding 3))
        (symbol->string (caddr lst))))))
    (if (equal? X-ancestry Y-ancestry)
      (string-append
        "\n")
    
      "\n""
Tweaks and overrides

```lisp
(format #f "~&")
(make-string padding #\space)
"X,Y: "
(if (list? (cadr lst))
  (format-ancestry (cadr lst) (+ padding 5))
  (symbol->string (cadr lst))))
(string-append
 (format #f "~&")
(make-string padding #\space)
"X: " X-ancestry
"\n"
(make-string padding #\space)
"Y: " Y-ancestry
(format #f "~&"))")

#(define (display-ancestry grob)
  (format (current-error-port)
"~-3&a~-2%-a-&"  
(make-string 36 #\-)
(if (ly:grob? grob)
  (format-ancestry (get-ancestry grob) 0)
  (format #f "~-a is not a grob" grob))))

\relative c' {\once \override NoteHead.before-line-breaking = #display-ancestry f4
\once \override Accidental.before-line-breaking = #display-ancestry
\once \override Arpeggio.before-line-breaking = #display-ancestry
<f as c>4\arpeggio }

\relative c''' {\time 3/4
\key f \major
\set harmonicDots = ##t
<bes f'\harmonic>2. ~
<bes f'\harmonic>4. <a e'\harmonic>8( <gis dis'\harmonic> <g d'\harmonic>)
<fis cis'\harmonic>2.
<bes f'\harmonic>2. }
```

Dotted harmonics

Artificial harmonics using \harmonic do not show dots. To override this behavior, set the context property harmonicDots.

\relative c'''' {\time 3/4
\key f \major
\set harmonicDots = ##t
<bes f'\harmonic>2. ~
<bes f'\harmonic>4. <a e'\harmonic>8( <gis dis'\harmonic> <g d'\harmonic>)
<fis cis'\harmonic>2.
<bes f'\harmonic>2. }
Drawing boxes around grobs

The `print-function` can be overridden to draw a box around an arbitrary grob.

```latex
relative c' { 
  \override TextScript.stencil = 
    #(make-stencil-boxer 0.1 0.3 ly:text-interface::print) 
  c'4"foo"

\override Stem.stencil = 
  #(make-stencil-boxer 0.05 0.25 ly:stem::print) 
\override Score.RehearsalMark.stencil = 
  #(make-stencil-boxer 0.15 0.3 ly:text-interface::print) 

\revert Stem.stencil 
\revert Flag.stencil 
\mark \default 
\}
```

Drawing circles around various objects

The `\circle` markup command draws circles around various objects, for example fingering indications. For other objects, specific tweaks may be required: this example demonstrates two strategies for rehearsal marks and measure numbers.

```latex
relative c' { 
  c1 
  \set Score.rehearsalMarkFormatter = 
    #(lambda (mark context) 
      (make-circle-markup (format-mark-numbers mark context)))) 
  \mark \default 

  c2 d"\markup { 
    \override #'(thickness . 3) { 
      circle \finger 2 
    } 
  }

\override Score.BarNumber.break-visibility = #all-visible 
\override Score.BarNumber.stencil = 
  #(make-stencil-circler 0.1 0.25 ly:text-interface::print)
```
Estensore testuale della dinamica personalizzato

Funzioni postfix per estensori testuali personalizzati del crescendo. Gli estensori devono iniziare sulla prima nota della misura; e bisogna usare -\mycresc, altrimenti l’inizio dell’estensore viene assegnato alla nota successiva.

% Two functions for (de)crescendo spanners where you can explicitly give the spanner text.

\mycresc =
#(define-music-function (mymarkup) (markup?)
  (make-music 'CrescendoEvent
    'span-direction START
    'span-type 'text
    'span-text mymarkup))

\mydecresc =
#(define-music-function (mymarkup) (markup?)
  (make-music 'DecrescendoEvent
    'span-direction START
    'span-type 'text
    'span-text mymarkup))

\relative c' {
  c4-\mycresc "custom cresc" c4 c4 c4 |
  c4 c4 c4 c4 |
  c4-\mydecresc "custom decresc" c4 c4 c4 |
  c4 c4\! c4 c4
}

Estensore testuale della dinamica personalizzato

Si possono definire estensori testuali personalizzati che fanno uso delle forcine e dei crescendo testuali. \(< \) \(> \) generano le forcine, \(\cresc \) etc. generano gli estensori testuali.

% Some sample text dynamic spanners, to be used as postfix operators

\crpoco =
#(make-music 'CrescendoEvent
  'span-direction START
  'span-type 'text
  'span-text "cresc. poco a poco")

\relative c' {
  c4\cresc d4 e4 f4 |
Extending a TrillSpanner

For TrillSpanner, the minimum-length property becomes effective only if the set-spacing-rods procedure is called explicitly.

To do this, the springs-and-rods property should be set to ly:spanner::set-spacing-rods.

```
(relative c' { 
(key c\minor 
(time 2/4 
c16( as'\ ) c,-\ des-. 
(once)\override TrillSpanner.minimum-length = #15 
(once)\override TrillSpanner.springs-and-rods = #ly:spanner::set-spacing-rods 
(afterGrace es4 
(startTrillSpan { d16([{ \stopTrillSpan es}) ] } 
c( c' g es c g' es d 
(hideNotes 
c8) 
)
```

Estendere i glissandi sulle volte delle ripetizioni

Un glissando che si estende in vari blocchi \alternative può essere simulato aggiungendo all’inizio di ogni blocco \alternative una nota di abbellimento nascosta da cui inizia un glissando. La nota di abbellimento deve avere la stessa altezza della nota da cui parte il glissando iniziale. In questo frammento si usa una funzione musicale che prende come argomento l’altezza della nota di abbellimento.

Attenzione: nella musica polifonica la nota di abbellimento deve avere una nota di abbellimento corrispondente in tutte le altre voci.

```
repeatGliss = #define-music-function (grace) 
(ly:pitch?)#
  (\once \override Glissando.springs-and-rods 
    = #ly:spanner::set-spacing-rods
```

\once \override Glissando.minimum-length = #3.5
\once \hideNotes
\grace $\text{grace} \text{glissando}
#
}

\score {
\relative c' {
\repeat volta 3 { c4 d e f\text{glissando} }
\alternative {
  { g2 d }
  { \repeatGliss f g2 e }
  { \repeatGliss f e2 d }
}
}
}

music = \relative c' {
\voiceOne
\repeat volta 2 {
  g a b c\text{glissando}
}
\alternative {
  { d1 }
  { \repeatGliss c \once \omit StringNumber e1\text{2} }
}
}

\score {
\new StaffGroup <<
\new Staff <<
  \new Voice { \clef "G_8" \music }
>>
\new TabStaff <<
  \new TabVoice { \clef "moderntab" \music }
>>
}

\begin{music}
\input /dots.dot
\end{music}
Fine-tuning pedal brackets

The appearance of pedal brackets may be altered in different ways.

\begin{verbatim}
paper { ragged-right = ##f }
relative c'' {
  c2\sostenutoOn c
  c2\sostenutoOff c
  \once \override Staff.PianoPedalBracket.shorten-pair = #'(-7 . -2)
  c2\sostenutoOn c
  c2\sostenutoOff c
  \once \override Staff.PianoPedalBracket.edge-height = #'(0 . 3)
  c2\sostenutoOn c
  c2\sostenutoOff c
}
\end{verbatim}

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{example.pdf}
\caption{Example of fine-tuned pedal brackets}
\end{figure}

Flat Ties

The function takes the default Tie.stencil as an argument, calculating the result relying on the extents of this default.

Further tweaking is possible by overriding Tie.details.height-limit or with \shape. It’s also possible to change the custom-definition on the fly.

%% http://lsr.di.unimi.it/LSR/Item?id=1031

\begin{verbatim}
#(define ((flared-tie coords) grob)
  (define (pair-to-list pair)
    (list (car pair) (cdr pair)))
  (define (normalize-coords goods x y dir)
    (map (lambda (coord)
           (coord-scale coord (cons x (* y dir)))
           (cons (* x (car coord)) (* y dir (cdr coord)))
        goods))
  (define (my-c-p-s points thick)
    (make-connected-path-stencil
      points
      thick
      1.0
      1.0
      #f
      #f))
  ;; outer let to trigger suicide
  (let ((sten (ly:tie::print grob))
        (if (grob::is-live? grob)

\end{verbatim}
(let* ((layout (ly:grob-layout grob))
   (line-thickness (ly:output-def-lookup layout 'line-thickness))
   (thickness (ly:grobp-property grob 'thickness 0.1))
   (used-thick (* line-thickness thickness))
   (dir (ly:grobp-property grob 'direction))
   (xex (ly:stencil-extent sten X))
   (yex (ly:stencil-extent sten Y))
   (lenx (interval-length xex))
   (leny (interval-length yex))
   (xtrans (car xex))
   (ytrans (if (> dir 0) (car yex) (cdr yex)))
   (uplist
    (map pair-to-list
      (normalize-coords coords lenx (* leny 2) dir)))))

(ly:stencil-translate
  (my-c-p-s uplist used-thick)
  (cons xtrans ytrans)))

#(define flare-tie
  (flared-tie '((0 . 0)(0.1 . 0.2) (0.9 . 0.2) (1.0 . 0.0))))

\layout {
  \context {
    \Voice
    \override Tie.stencil = #flare-tie
  }
}

\paper { ragged-right = ##f }

\relative c' {
  a4-a
  \override Tie.height-limit = 4
  a'4-a
  a'4-a
  <a,, c e a c e a c e>- q

  \break

  a'4-a
  \once \override Tie.details.height-limit = 14
  a4-a

  \break

  a4-a
  \once \override Tie.details.height-limit = 0.5
  a4-a

  \break

Force a cancellation natural before accidentals

The following example shows how to force a natural sign before an accidental.

\relative c' { 
  \key es \major 
  bes c des 
  \tweak Accidental.restore-first ##t 
  eis 
}
Forzare lo spostamento orizzontale delle note

Quando il motore tipografico non riesce a risolvere una situazione, si può usare la sintassi che sovrascrive le decisioni tipografiche. L’unità di misura usata è lo spazio del rigo.

\relative c' <<
{ 
  <d g>2 <d g>
}
\
{ 
  <b f'>2
  \once \override NoteColumn.force-hshift = #1.7
  <b f'>2
}
>>

Fret diagrams explained and developed

This snippet shows many possibilities for obtaining and tweaking fret diagrams.

<<
\chords {
  a2 a
  \repeat unfold 3 {
    c c c d d
  }
}

\new Voice = "mel" {
  \textLengthOn
  % Set global properties of fret diagram
  \override TextScript.size = #1.2
  \override TextScript.fret-diagram-details.finger-code = #'below-string
  \override TextScript.fret-diagram-details.dot-color = #'black

  %% A chord for ukulele
  a'2\markup {
    \override #'(fret-diagram-details . ( 
      (string-count . 4)
      (dot-color . white)
      (finger-code . in-dot))) { 
      \fret-diagram "4-2-2;3-1-1;2-o;1-o;" 
    }
  }
}
%% A chord for ukulele, with formatting defined in definition string
% 1.2 * size, 4 strings, 4 frets, fingerings below string
% dot radius .35 of fret spacing, dot position 0.55 of fret spacing
\markup {\override #'(fret-diagram-details . (dot-color . white) (open-string . "o")) {\fret-diagram "s:1.2;w:4;h:3;f:2;d:0.35;p:0.55;4-2-2;3-1-1;2-o;1-o;"}}
}

%% These chords will be in normal orientation

%% C major for guitar, barred on third fret
% verbose style
% roman fret label, finger labels below string, straight barre
\markup {\override #'(size . 1.1) {\override #'(fret-diagram-details . (number-type . roman-lower) (finger-code . below-string) (barre-type . straight))) {\fret-diagram-verbose #'(mute 6) (place-fret 5 3 1) (place-fret 4 5 2) (place-fret 3 5 3) (place-fret 2 5 4) (place-fret 1 3 1) (barre 5 1 3)}}}

%% C major for guitar, barred on third fret
% Double barre used to test barre function
% verbose style
\markup {\override #'(size . 1.1) {\override #'(fret-diagram-details . (number-type . arabic) (dot-label-font-mag . 0.9) (finger-code . in-dot) (fret-label-font-mag . 0.6) (fret-label-vertical-offset . 0) (label-dir . -1) (mute-string . "M") (xo-font-magnification . 0.4) (xo-padding . 0.3))) {
\fret-diagram-verbose #'(mute 6)
  (place-fret 5 3 1)
  (place-fret 4 5 2)
  (place-fret 3 5 3)
  (place-fret 2 5 4)
  (place-fret 1 3 1)
  (barre 4 2 5)
  (barre 5 1 3))

%%%% C major for guitar, with capo on third fret
%%%% verbose style
c'2''\markup {
  % 110% of default size
  \override #'(size . 1.1) {
    \override #'(fret-diagram-details . (number-type . roman-upper)
      (dot-label-font-mag . 0.9)
      (finger-code . none)
      (fret-label-vertical-offset . 0.5)
      (xo-font-magnification . 0.4)
      (xo-padding . 0.3))) {
      \fret-diagram-verbose #'(mute 6)
        (capo 3)
        (open 5)
        (place-fret 4 5 1)
        (place-fret 3 5 2)
        (place-fret 2 5 3)
        (open 1))
  }
}

%%%% simple D chord
d'2''\markup {
  \override #'(fret-diagram-details . (finger-code . below-string)
    (dot-radius . 0.35)
    (string-thickness-factor . 0.3)
    (dot-position . 0.5)
    (fret-count . 3))) {
    \fret-diagram-terse "x;x;o;2-1;3-2;2-3;"
  }
}

%%%% simple D chord, large top fret thickness
d'2''\markup {
  \override #'(fret-diagram-details . (finger-code . below-string)
    (dot-radius . 0.35)
    (dot-position . 0.5)
    (fret-count . 3))) {

(dot-position . 0.5)
(top-fret-thickness . 7)
(fret-count . 3))) {
  \fret-diagram-terse "x;x;o;2-1;3-2;2-3;"
}
}

% These chords will be in landscape orientation
\override TextScript.fret-diagram-details.orientation = #'landscape

%% C major for guitar, barred on third fret
% verbose style
% roman fret label, finger labels below string, straight barre
\markup {
  \percent 110\% of default size
  \override #'(size . 1.1) {
    \override #'(fret-diagram-details . (num
      (number-type . roman-lower)
      (fret-label-font-mag . 0.6)
      (fret-label-vertical-offset . 0)
      (label-dir . -1)
      (mute-string . "M")
      (xo-font-magnification . 0.4)
      (xo-padding . 0.3))) {
      \fret-diagram-verbose #'(mute 6)
        (place-fret 5 3 1)
        (place-fret 4 5 2)
        (place-fret 3 5 3)
        (place-fret 2 5 4)
        (place-fret 1 3 1)
        (barre 5 1 3))
    }
  }
}

%% C major for guitar, barred on third fret
%% Double barre used to test barre function
% verbose style
\markup {
  \percent 110\% of default size
  \override #'(size . 1.1) {
    \override #'(fret-diagram-details . (num
      (number-type . arabic)
      (dot-label-font-mag . 0.9)
      (fret-label-font-mag . 0.6)
      (fret-label-vertical-offset . 0)
      (label-dir . -1)
      (mute-string . "M")
      (xo-font-magnification . 0.4)
      (xo-padding . 0.3))) {
      \fret-diagram-verbose #'(mute 6)
        (place-fret 5 3 1)
        (place-fret 4 5 2)
        (place-fret 3 5 3)
        (place-fret 2 5 4)
(place-fret 1 3 1)
(barre 4 2 5)
(barre 5 1 3))
}
}

%%% C major for guitar, with capo on third fret
%%% verbose style
c'2``\markup {
  \% 110% of default size
  \override #'(size . 1.1) {
    \override #'(fret-diagram-details . ()
      (number-type . roman-upper)
      (dot-label-font-mag . 0.9)
      (finger-code . none)
      (fret-label-vertical-offset . 0.5)
      (xo-font-magnification . 0.4)
      (xo-padding . 0.3))) {
      \fret-diagram-verbose #'(mute 6)
        (capo 3)
        (open 5)
        (place-fret 4 5 1)
        (place-fret 3 5 2)
        (place-fret 2 5 3)
        (open 1))
    }
  }
}

%%% simple D chord
d'2``\markup {
  \override #'(fret-diagram-details . ()
    (finger-code . below-string)
    (dot-radius . 0.35)
    (dot-position . 0.5)
    (fret-count . 3))) {
      \fret-diagram-terse "x;x;o;2-1;3-2;2-3;"
    }
}

%%% simple D chord, large top fret thickness
d'2``\markup {
  \override #'(fret-diagram-details . ()
    (finger-code . below-string)
    (dot-radius . 0.35)
    (dot-position . 0.5)
    (top-fret-thickness . 7)
    (fret-count . 3))) {
      \fret-diagram-terse "x;x;o;2-1;3-2;2-3;"
% These chords will be in opposing-landscape orientation
\override TextScript.fret-diagram-details.orientation = #'opposing-landscape

%% C major for guitar, barred on third fret
%% verbose style
%% roman fret label, finger labels below string, straight barre
c\text{"markup}
\begin{verbatim}
\override #\'(size . 1.1) {
  \override #\'(fret-diagram-details . (
    (number-type . roman-lower)
    (finger-code . below-string)
    (barre-type . straight))) {
    \fret-diagram-verbose #\'(mute 6)
    (place-fret 5 3 1)
    (place-fret 4 5 2)
    (place-fret 3 5 3)
    (place-fret 2 5 4)
    (place-fret 1 3 1)
    (barre 5 1 3)
  }
}
\end{verbatim}

%% C major for guitar, barred on third fret
%% Double barre used to test barre function
%% verbose style
c\text{"markup}
\begin{verbatim}
\override #\'(size . 1.1) {
  \override #\'(fret-diagram-details . (
    (number-type . arabic)
    (dot-label-font-mag . 0.9)
    (fret-label-font-mag . 0.6)
    (fret-label-vertical-offset . 0)
    (label-dir . -1)
    (mute-string . "M")
    (xo-font-magnification . 0.4)
    (xo-padding . 0.3))) {
    \fret-diagram-verbose #\'(mute 6)
    (place-fret 5 3 1)
    (place-fret 4 5 2)
    (place-fret 3 5 3)
    (place-fret 2 5 4)
    (place-fret 1 3 1)
    (barre 4 2 5)
    (barre 5 1 3)
  }
}
\end{verbatim}
%% C major for guitar, with capo on third fret
%% verbose style
c'2^\markup {
  \override #'(size . 1.1) {
    \override \fret-diagram-details . (number-type . roman-upper)
    \override \fret-diagram-details . (dot-label-font-mag . 0.9)
    \override \fret-diagram-details . (finger-code . none)
    \override \fret-diagram-details . (fret-label-vertical-offset . 0.5)
    \override \fret-diagram-details . (xo-font-magnification . 0.4)
    \override \fret-diagram-details . (xo-padding . 0.3))) {
      \fret-diagram-verbose 
        (mute 6)
        (capo 3)
        (open 5)
        (place-fret 4 5 1)
        (place-fret 3 5 2)
        (place-fret 2 5 3)
        (open 1))
  }
}

%% simple D chord
d'2^\markup {
  \override \fret-diagram-details . (finger-code . below-string)
  \override \fret-diagram-details . (dot-radius . 0.35)
  \override \fret-diagram-details . (dot-position . 0.5)
  \override \fret-diagram-details . (fret-count . 3))) {
    \fret-diagram-terse "x;x;o;2-1;3-2;2-3;"
  }
}

%% simple D chord, large top fret thickness
d'2^\markup {
  \override \fret-diagram-details . (finger-code . below-string)
  \override \fret-diagram-details . (dot-radius . 0.35)
  \override \fret-diagram-details . (dot-position . 0.5)
  \override \fret-diagram-details . (top-fret-thickness . 7)
  \override \fret-diagram-details . (fret-count . 3))) {
    \fret-diagram-terse "x;x;o;2-1;3-2;2-3;"
  }
}
Generating custom flags

The stencil property of the Flag grob can be set to a custom scheme function to generate the glyph for the flag.

```lilypond
#(define-public (weight-flag grob)
  (let* ((stem-grob (ly:grob-parent grob X))
         (log (~ (ly:grob-property stem-grob 'duration-log) 2))
         (is-up? (eqv? (ly:grob-property stem-grob 'direction) UP))
         (yext (if is-up? (cons (* log -0.8) 0) (cons 0 (* log 0.8))))
         (flag-stencil (make-filled-box-stencil '(-0.4 . 0.4) yext))
         (stroke-style (ly:grob-property grob 'stroke-style))
         (stroke-stencil (if (equal? stroke-style "grace")
                           (make-line-stencil 0.2 -0.9 -0.4 0.9 -0.4)
                           (empty-stencil)))
   (ly:stencil-add flag-stencil stroke-stencil)))

% Create a flag stencil by looking up the glyph from the font
#(define (inverted-flag grob)
  (let* ((stem-grob (ly:grob-parent grob X))
         (dir (if (eqv? (ly:grob-property stem-grob 'direction) UP) "d" "u"))
         (flag (retrieve-glyph-flag "" dir "" grob))
         (line-thickness (ly:staff-symbol-line-thickness grob))
         (stem-thickness (ly:grob-property stem-grob 'thickness))
         (stem-width (* line-thickness stem-thickness))
         (stroke-style (ly:grob-property grob 'stroke-style))
         (stencil (if (null? stroke-style)
                    flag
                    (add-stroke-glyph flag stem-grob dir stroke-style "")))
         (rotated-flag (ly:stencil-rotate-absolute stencil 180 0 0))
         (ly:stencil-translate rotated-flag (cons (- (/ stem-width 2)) 0))))

snippetexamplenotes =
{
  \autoBeamOff c'8 d'16 c'32 d'64 \acciaccatura {c'8} d'64
}
```
Tweaks and overrides

\{  
  \time 1/4  
  \textMark "Normal flags"  
  \snippetexemplenotes  
  \textMark "Custom flag: inverted"  
  \override Flag.stencil = #inverted-flag  
  \snippetexemplenotes  
  \textMark "Custom flag: weight"  
  \override Flag.stencil = #weight-flag  
  \snippetexemplenotes  
  \textMark "Revert to normal"  
  \revert Flag.stencil  
  \snippetexemplenotes  
  \}

\begin{music}
\begin{music}
\normalflags \customflag {\inverted} \revertnormal
\end{music}
\end{music}

\Glissandicankisskips\grobs

NoteColumn grobs can be skipped over by glissandi.
\relative c' {  
a2 \glissando  
\once \override NoteColumn.glissando-skip = ##t  
f''4 d,  
}

\Hairpins\with\different\linestyles

Hairpins can take any style from line-interface - dashed-line, dotted-line, line, trill or zigzag.
\relative c' {  
c2\< c\< c\< c\!
\override Hairpin.style = #'dashed-line  
c2\< c\< c\!
\override Hairpin.style = #'dotted-line  
c2\< c\< c\!
\override Hairpin.style = #'line  
c2\< c\< c\!
\override Hairpin.style = #'trill  
c2\< c\< c\!
\override Hairpin.style = #'zigzag  
c2\< c\< c\!
}
Horizontally aligning custom dynamics (e.g. "sempre pp", "piu f", "subito p")

Some dynamic expressions involve additional text, like “sempre pp”. Since dynamics are usually centered under the note, the \pp would be displayed way after the note it applies to.

To correctly align the “sempre pp” horizontally so that it is aligned as if it were only the \pp, there are several approaches:

- Simply use \once\override DynamicText.X-offset = #\(-9.2\) before the note with the dynamics to manually shift it to the correct position. Drawback: This has to be done manually each time you use that dynamic markup...
- Add some padding (#:\hspace 7.1) into the definition of your custom dynamic mark so that after LilyPond center-aligns it, it is already correctly aligned. Drawback: The padding really takes up that space and does not allow any other markup or dynamics to be shown in that position.
- Shift the dynamic script \once\override ... X-offset = .... Drawback: \once\override is needed for every invocation!
- Set the dimensions of the additional text to 0 (using #:with-dimensions `(0 . 0)` `(0 . 0)`). Drawback: For LilyPond, “sempre” has no extent now. This means it might put other stuff there, causing collisions (which are not detected by LilyPond’s collision detection algorithm!). There also seems to be some spacing, so it is not exactly the same alignment as without the additional text.
- Add an explicit shift directly inside the scheme function for the dynamic script.
- Set an explicit alignment inside the dynamic script. By default, this won’t have any effect, only if one sets X-offset! Drawback: One needs to set DynamicText.X-offset, which will apply to all dynamic texts! Also, it is aligned at the right edge of the additional text, not at the center of \pp.

\paper {
  \ragged-right = ##f
  indent = 2.5\cm
}

% Solution 1: Using a simple markup with a particular halign value
% Drawback: It's a markup, not a dynamic command, so \dynamicDown
% etc. will have no effect
semppMarkup = \markup { \halign #1.4 \italic "sempre" \dynamic "pp" }

% Solution 2: Using a dynamic script & shifting with
% Drawback: \once \override ...X-offset = ...
% \once \override needed for every invocation
semppK = #(make-dynamic-script

\revert Hairpin.style
\c2< c\!}
(markup #:line
  (#:normal-text
   #:italic "sempre"
   #:dynamic "pp")))

% Solution 3: Padding the dynamic script so the center-alignment
% puts it at the correct position
% Drawback: the padding really reserves the space, nothing else can be there
semppT =
#(make-dynamic-script
  (markup #:line
    (#:normal-text
     #:italic "sempre"
     #:dynamic "pp"
     #:hspace 7.1)))

% Solution 4: Dynamic, setting the dimensions of the additional text to 0
% Drawback: To lilypond "sempre" has no extent, so it might put
% other stuff there => collisions
% Drawback: Also, there seems to be some spacing, so it’s not exactly the
% same alignment as without the additional text
semppM =
#(make-dynamic-script
  (markup #:line
    (#:with-dimensions '(0 . 0) '(0 . 0)
     #:right-align
     #:normal-text
     #:italic "sempre"
     #:dynamic "pp")))

% Solution 5: Dynamic with explicit shifting inside the scheme function
semppG =
#(make-dynamic-script
  (markup #:hspace 0
    #:translate '(-18.85 . 0)
    #:line (#:normal-text
           #:italic "sempre"
           #:dynamic "pp")))

% Solution 6: Dynamic with explicit alignment. This has only effect
% if one sets X-offset!
% Drawback: One needs to set DynamicText.X-offset!
% Drawback: Aligned at the right edge of the additional text,
% not at the center of pp
semppMII =
#(make-dynamic-script
  (markup #:line (#:right-align
               #:normal-text
               #:italic "sempre"
               #:dynamic "pp")))
\new StaffGroup <<
\new Staff = "s" \with { instrumentName = \markup \column { Normal } }
<<
  \relative c'' {
    \key es \major
    c4\pp c\p c c | c\ff c c\pp c
  }
>>
\new Staff = "sMarkup" \with {
  instrumentName = \markup \column { Normal markup } }
<<
  \relative c'' {
    \key es \major
    c4-\semppMarkup c\p c c | c\ff c c-\semppMarkup c
  }
>>
\new Staff = "sK" \with {
  instrumentName = \markup \column { Explicit shifting } }
<<
  \relative c'' {
    \key es \major
    \once \override DynamicText.X-offset = #-9.2
    c4\semppK c\p c c
    c4\ff c
c7-\semppK c
    \once \override DynamicText.X-offset = #-9.2
    c4\semppK c
  }
>>
\new Staff = "sT" \with {
  instrumentName = \markup \column { Right padding } }
<<
  \relative c'' {
    \key es \major
    c4\semppT c\p c c | c\ff c c\semppT c
  }
>>
\new Staff = "sM" \with {
  instrumentName = \markup \column { Set dimension "to zero" } }
<<
  \relative c'' {
    \key es \major
    c4\semppM c\p c c | c\ff c c\semppM c
  }
>>
\new Staff = "sG" \with {
  instrumentName = \markup \column { Shift inside dynamics } }
<<
  \relative c'' {

Tweaks and overrides

\key es \major
c4\semppG \p \p | \p \pp c \semppG c

>>
\new Staff = "sMII" \with {
instrumentName = \markup \column { Alignment inside dynamics }
}
<<
\relative c'' {
\key es \major
% Setting to ##f (false) gives the same result
\override DynamicText.X-offset = #0
c4\semppMII \p \p | \p \pp \semppMII c
}
>>
>>

\layout { \override Staff.InstrumentName.self-alignment-X = #LEFT }

How to change fret diagram position

If you want to move the position of a fret diagram, for example, to avoid collision, or to place it between two notes, you have various possibilities:

1) modify \#padding or \#extra-offset values (as shown in the first snippet)

2) you can add an invisible voice and attach the fret diagrams to the invisible notes in that voice (as shown in the second example).
If you need to move the fret according with a rhythmic position inside the bar (in the example, the third beat of the measure) the second example is better, because the fret is aligned with the third beat itself.

```
harmonies = \chordmode
{
  a8:13
  % THE FOLLOWING IS THE COMMAND TO MOVE THE CHORD NAME
  \once \override ChordNames.ChordName.extra-offset = #'(10 . 0)
  b8:13 s2.
  % THIS LINE IS THE SECOND METHOD
  s4 s4 b4:13
}

\score
{
  \new ChordNames \harmonies
  \new Staff
  \{a8\markup { \fret-diagram "6-x;5-0;4-2;3-0;2-0;1-2;" }\}
  % THE FOLLOWING IS THE COMMAND TO MOVE THE FRET DIAGRAM
  \once \override TextScript.extra-offset = #'(10 . 0)
  b4.~\markup { \fret-diagram "6-x;5-2;4-4;3-2;2-2;1-4;" } b4. a8\break
  % HERE IS THE SECOND METHOD
  \{ a8 b4.~ b4. a8\}
  \{ s4 s4 a4\markup { \fret-diagram "6-x;5-2;4-4;3-2;2-2;1-4;" }\}
  \}
}
```

Inserire una cesura

I segni di cesura possono essere creati sovrascrivendo la proprietà 'text dell’oggetto BreathingSign. È disponibile anche un segno di cesura curvo.

```
\relative c'' {
```
Keep change clefs full sized

When a clef is changed, the clef sign displayed is smaller than the initial clef. This can be overridden with \full-size-change.

```tex
relative c' {
  \clef "treble"
  c1
  \clef "bass"
  c1
  \clef "treble"
  c1
  \override Staff.Clef.full-size-change = ##t
  \clef "bass"
  c1
  \clef "treble"
  c1
  \revert Staff.Clef.full-size-change
  \clef "bass"
  c1
  \clef "treble"
  c1
}
```

Line arrows

Arrows can be applied to text-spanners and line-spanners (such as the Glissando).

```tex
relative c'' {
  \override TextSpanner.bound-padding = #1.0
  \override TextSpanner.style = #'line
  \override TextSpanner.bound-details.right.arrow = ##t
  \override TextSpanner.bound-details.left.text = #"fof"
  \override TextSpanner.bound-details.right.text = #"gag"
  \override TextSpanner.bound-details.right.padding = #0.6
```
Making an object invisible with the 'transparent property

Setting the transparent property will cause an object to be printed in “invisible ink”: the object is not printed, but all its other behavior is retained. The object still takes up space, it takes part in collisions, and slurs, ties and beams can be attached to it.

This snippet demonstrates how to connect different voices using ties. Normally, ties only connect two notes in the same voice. By introducing a tie in a different voice, and blanking the first up-stem in that voice, the tie appears to cross voices.

\relative {
  \time 2/4
  <<
  { \once \hide Stem
    \once \override Stem.length = #8
    b'8 \noBeam
    \once \hide Stem
    \once \override Stem.length = #8
    g8 \noBeam
  }
  \\
  { b8 g g e }
  >>
}

Lasciare che i glissandi vadano a capo

Per permettere a un glissando di andare a capo se capita su un’interruzione di riga, si impostano le proprietà breakable e after-line-breaking su #t:

glissandoSkipOn = {
\override NoteColumn.glissando-skip = ##t
\hide NoteHead
\override NoteHead.no-ledgers = ##t
}

\relative c'' {\override Glissando.breakable = ##t
\override Glissando.after-line-breaking = ##t
f\glissando |
\break
a4 r2. |
f\glissando
\once \glissandoSkipOn
\break
a2 a4 r4 |
}

Manually controlling beam positions

Beam positions may be controlled manually, by overriding the positions setting of the Beam grob.

\relative c' {\time 2/4
% from upper staff-line (position 2) to center (position 0)
\override Beam.positions = #'(2 . 0)
c8 c
% from center to one above center (position 1)
\override Beam.positions = #'(0 . 1)
c8 c
}

Measure-centered bar numbers

For film scores, a common convention is to center bar numbers within their measure. This is achieved through setting the centerBarNumbers context property to true. When this is used, the type of the bar number grobs is CenteredBarNumber rather than BarNumber.
This example demonstrates a number of settings: the centered bar numbers are boxed and placed below the staves.

```\layout {
\context {
  \Score
  centerBarNumbers = ##t
  barNumberVisibility = #all-bar-numbers-visible
  \override CenteredBarNumber stencil
    = #(make-stencil-boxer 0.1 0.25 ly:text-interface::print)
  \override CenteredBarNumberLineSpanner.direction = #DOWN
}
}

\new StaffGroup <<
\new Staff \relative c' {
  d4-. f8( e d4) bes'-> |
  d, .- f8( e d4) cis'-> |
  g-. f8( d e4) g-> |
  a,1-> |
}
\new Staff \relative c {
  \clef bass
  d4 f8 e d2~ |
  4 f8 e d2~ |
  4 4 2 |
  a1 |
}>>
```

Formattazione mensurale (stanghette tra i righi)
La formattazione mensurale, in cui le stanghette non appaiono sui righi ma nello spazio tra i righi, si può ottenere usando StaffGroup al posto di ChoirStaff. La stanghetta sui righi viene nascosta con \hide.

```\layout {
\context {
  \Staff
    measureBarType = "-span|"
}
}
```

music = \fixed c' {
  c1
Tweaks and overrides

Modificare l’inclinazione dell’estensore dell’ottava
È possibile cambiare l’inclinazione dell’estensore dell’ottava.

```latex
relative c'' {
  \override Staff.OttavaBracket.stencil = \ly:line-spanner::print
  \override Staff.OttavaBracket.bound-details = 
    \ly:horizontal-line-spanner::calc-left-bound-info-and-text
  \override Staff.OttavaBracket.right-bound-info = 
    \ly:horizontal-line-spanner::calc-right-bound-info

  \ottava #1
  c1
  c''''1
}
```

Spostare le note puntate in polifonia
Quando una nota puntata della voce più alta viene spostata per evitare una collisione con una nota di un’altra voce, il comportamento predefinito è spostare la nota più alta a destra. Tale comportamento può essere modificato tramite la proprietà prefer-dotted-right di NoteCollision.
Moving slur positions vertically

The vertical position of a slur can be adjusted using the positions property of Slur. The property has 2 parameters, the first referring to the left end of the slur and the second to the right. The values of the parameters are not used by LilyPond to make an exact movement of the slur - instead it selects what placement of the slur looks best, taking into account the parameter values. Positive values move the slur up, and are appropriate for notes with stems down. Negative values move downward slurs further down.

\relative c' {
  \stemDown
  e4( a)
  \override Slur.positions = #'(1 . 1)
  e4( a)
  \override Slur.positions = #'(2 . 2)
  e4( a)
  \override Slur.positions = #'(3 . 3)
  e4( a)
  \override Slur.positions = #'(4 . 4)
  e4( a)
  \override Slur.positions = #'(5 . 5)
  e4( a)
  \override Slur.positions = #'(0 . 5)
  e4( a)
  \override Slur.positions = #'(5 . 0)
  e4( a)
  \stemUp
  \override Slur.positions = #'(-5 . -5)
  e4( a)
  \stemDown
  \revert Slur.positions
  e4( a)
}
Annidare i righi

Si può usare la proprietà systemStartDelimiterHierarchy per creare gruppi di righi annidati più complessi. Il comando \set StaffGroup.systemStartDelimiterHierarchy prende come argomento una lista alfabetica dell’insieme di righi prodotti. Prima di ogni rigo si può assegnare un delimitatore di inizio del sistema. Deve essere racchiuso tra parentesi e collega tutti i righi compresi tra le parentesi. Gli elementi nella lista possono essere omessi, ma la prima parentesi quadra collega sempre tutti i righi. Le possibilità sono SystemStartBar, SystemStartBracket, SystemStartBrace e SystemStartSquare.

\new StaffGroup
\relative c'' <<
\override StaffGroup.SystemStartSquare.collapse-height = #4
\set StaffGroup.systemStartDelimiterHierarchy
  = #'(SystemStartSquare (SystemStartBrace (SystemStartBracket a
                        (SystemStartSquare b) ) c ) d)
\new Staff { c1 }
\new Staff { c1 }
\new Staff { c1 }
\new Staff { c1 }
\new Staff { c1 }
\new Staff { c1 }
>>

Overriding articulations by type

Sometimes you may want to affect a single articulation-type. Although it is always possible to use \tweak, it might become tedious to do so for every single sign of a whole score. The following shows how to tweak articulations with a list of custom settings. One use-case might be to create a style sheet.
With 2.16.2 and above it is possible to put the proposed function, `\customScripts`, into a `\layout`-block.

% Code by David Nalesnik and Thomas Morley

#(define (custom-script-tweaks ls)
  (lambda (grob)
    (let* ((type (ly:event-property
                   (ly:grob-property grob 'cause)
                   'articulation-type))
           (tweaks (assoc-ref ls type)))
      (if tweaks
          (for-each
            (lambda (x) (ly:grob-set-property!
                          grob (car x) (cdr x)))
            tweaks))))

customScripts =
#(define-music-function (settings) (list?)
 #{
  \override Script.before-line-breaking =
   #(custom-script-tweaks settings)
 #})

revertCustomScripts = \revert Script.before-line-breaking

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Example:
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Predefine a list of desired tweaks.
#(define my-settings-1
  '(
    (staccato . ((color . (1 0 0))
                  (padding . 0.5)))
    (accent . ((font-size . 0)
               (color . (1 0 0))))
    (tenuto . ((rotation . (45 0 0))
               (padding . 2)
               (font-size . 10)))
    (staccatissimo . ((padding . 1)
                      (color . (1 0 0))))
    (segno . ((font-size . 0)
              (color . (1 0 0))))
  ))

#(define my-settings-2
  '(
    (staccato . ((color . (0 1 0))))
    (accent . ((font-size . 4)
               (color . (0 1 0)))
               (padding . 1.5))
    (tenuto . ((font-size . 10))))
(staccatissimo . ((padding . 2)
  (color . (0 1 0))))
(coda . ((color . (0 1 0))
  (padding . 1)))
)

one = \relative c'' {
  f1--
  \customScripts #my-settings-1
  f-. f-! f--> f--! \segno
  \revertCustomScripts
  f--> f-. }
}

two = \relative c' {
  f1--
  \customScripts #my-settings-2
  f-. f-! f--> f---> f-!
  f--> f-. \coda
}
\new Staff <<
  \new Voice { \voiceOne \one }
  \new Voice { \voiceTwo \two }
  >>

Overriding articulations of distinct type

Sometimes you may want to affect a single articulation type. Although it is always possible to use \tweak, it might become tedious to do so for every single sign of a whole score. The following shows how to tweak articulations with a list of custom settings. One use case might be to create a style sheet.

With 2.16.2 it is possible to put the proposed function, \customScripts, into a \layout block.

% Code by David Nalesnik and Thomas Morley

#(define (custom-script-tweaks ls)
 (lambda (grob)
   (let* ((type (ly:prob-property
     (ly:grob-property grob 'cause)
     'articulation-type))
     (tweaks (assoc-ref ls type)))
     (if tweaks
(for-each
  (lambda (x) (ly:grob-set-property! grob (car x) (cdr x))
    tweaks)))))

customScripts =
#(define-music-function (settings)(list?)
 #{\override Script.before-line-breaking =
  #(custom-script-tweaks settings)
  #})
revertCustomScripts = { \revert Script.before-line-breaking }

%%%%%%%%%%%%%%%%%%
% Example:
%%%%%%%%%%%%%%%%%%

% Predefine a list of desired tweaks.
#(define my-settings-1
  '(  
    ("staccato" . ((color . (1 0 0))(padding . 0.5)))
    ("accent" . ((font-size . 0)(color . (1 0 0))))
    ("tenuto" . ((rotation . (45 0 0)) (padding . 2)(font-size . 10)))
    ("staccatissimo" . ((padding . 1) (color . (1 0 0))))
    ("segno" . ((font-size . 0)(color . (1 0 0))))
  ))

#(define my-settings-2
  '(  
    ("staccato" . ((color . (0 1 0))))
    ("accent" . ((font-size . 4)(color . (0 1 0))(padding . 1.5)))
    ("tenuto" . ((font-size . 10))))
    ("staccatissimo" . ((padding . 2) (color . (0 1 0))))
    ("coda" . ((color . (0 1 0)) (padding . 1)))
  ))

one = \relative c'' {  
  f1--  
  \customScripts #my-settings-1  
  f-. f-! f--> f--! f-->\segno  
  \revertCustomScripts  
  f--> f-.}
}

two = \relative c' {  
  f1--  
  \customScripts #my-settings-2  
  f-. f-! f--> f----> f-!  
  f--> f-.\coda  
}
Visibilità del conto della ripetizione con segno percentuale

I contatori della ripetizione con segno percentuale possono essere mostrati a intervalli regolari impostando la proprietà di contesto repeatCountVisibility.

Positioning arpeggios

If you need to extend or shorten an arpeggio, you can modify the upper and lower start positions independently.
Posizionamento delle pause multiple

Diversamente dalle pause normali, non esiste un comando predefinito per cambiare la posizione sul rigo di un simbolo di pausa multipla di qualsiasi tipo connettendolo a una nota. Tuttavia, nella musica polifonica le pause multiple nelle voci dispari e pari sono separate verticalmente. Il posizionamento delle pause multiple si controlla nel modo seguente:

\relative c' { 
  % Multi-measure rests by default are set under the fourth line
  R1
  \override MultiMeasureRest.staff-position = #-2
  R1
  \override MultiMeasureRest.staff-position = #0
  R1
  \override MultiMeasureRest.staff-position = #2
  R1
  \override MultiMeasureRest.staff-position = #3
  R1
  \override MultiMeasureRest.staff-position = #6
  R1
  \revert MultiMeasureRest.staff-position
\break

  % In two Voices, odd-numbered voices are under the top line
  << { R1 } \ \ { a1 } >>
  % Even-numbered voices are under the bottom line
  << { a1 } \ \ { R1 } >>
  % Multi-measure rests in both voices remain separate
  << { R1 } \ \ { R1 } >>

  % Separating multi-measure rests in more than two voices
  % requires an override
  << { R1 } \ \ { R1 } \ \ 
    \once \override MultiMeasureRest.staff-position = #0
    { R1 }
  >>

  % Using compressed bars in multiple voices requires another override
  % in all voices to avoid multiple instances being printed
  \compressMMRests
  <<
    \revert MultiMeasureRest.direction
    { R1*3 }
    \ \
    \revert MultiMeasureRest.direction
    { R1*3 }
  >>
}
Tweaks and overrides

Posizionare il testo a margine dentro le legature di portamento

I testi a margine devono avere la proprietà outside-staff-priority impostata su false per poter apparire dentro le legature di portamento.

\relative c' {
  \override TextScript.avoid-slur = #'inside
  \override TextScript.outside-staff-priority = ##f
  c2(\markup { \halign #-10 \natural } d4.) c8
}\n
Numeri di battuta racchiusi in rettangoli o cerchi

I numeri di battuta possono apparire anche all’interno di rettangoli o cerchi.

\relative c' {
  % Prevent bar numbers at the end of a line and permit them elsewhere
  \override Score.BarNumber.break-visibility = #end-of-line-invisible
  \set Score.barNumberVisibility = #(every-nth-bar-number-visible 4)

  % Increase the size of the bar number by 2
  \override Score.BarNumber.font-size = #2

  % Draw a box round the following bar number(s)
  \override Score.BarNumber.stencil
    = #(make-stencil-boxer 0.1 0.25 ly:text-interface::print)
  \repeat unfold 5 { c1 }

  % Draw a circle round the following bar number(s)
  \override Score.BarNumber.stencil
    = #(make-stencil-circler 0.1 0.25 ly:text-interface::print)
  \repeat unfold 4 { c1 } \bar "|."\n}\n
Posizionare il metronomo e i numeri di chiamata sotto il rigo

Di norma, il metronomo e i numeri di chiamata vengono posizionati sopra il rigo. Per metterli sotto il rigo basta impostare correttamente la proprietà direction di MetronomeMark o RehearsalMark.

\layout {
Tweaks and overrides

```plaintext
indent = 0
ragged-right = ##f
}

{
  % Metronome marks below the staff
  \override Score.MetronomeMark.direction = #DOWN
  \tempo 8. = 120
c''1

  % Rehearsal marks below the staff
  \override Score.RehearsalMark.direction = #DOWN
  \mark \default
c''1
}

\(\text{\textguitar} A = 120\)

Print the note names with and without an octave marker

The `NoteNames` context can be used to print the text value of notes. The `printOctaveNames` property turns on or off the representation of the octave of the note.

```plaintext
scale = \relative c' { 
a4 b c d
e4 f g a
}

\new Staff { 
  <<
    \scale
    \context NoteNames { 
      \set printOctaveNames = ##f
      \scale
    }
  >>
  R1
  <<
    \scale
    \context NoteNames { 
      \set printOctaveNames = ##t
      \scale
    }
  >>
}
```
Printing tuplet brackets on the note head side

Whichever option you choose for controlling the tuplet bracket visibility, it will show or hide the tuplet bracket irrespectively of tuplet bracket placement (stem side or note head side). However, when placing the tuplet bracket on the note head side some authors recommend always printing the tuplet bracket. The option visible-over-note-heads can be used to achieve this.

```latex
\texttt{\textbackslash relative c'' \{\\}
\texttt{\textbackslash tupletNeutral \textbackslash tuplet 3/2 \{ c16[ d e } f8\]\\}
\texttt{\textbackslash tupletUp \textbackslash tuplet 3/2 \{ c8 d e }\}\}
```

Proportional strict notespacing

If strict-note-spacing is set spacing of notes is not influenced by bars or clefs within a system. Rather, they are placed just before the note that occurs at the same time. This may cause collisions.

```latex
\texttt{\relative c' \inner}{\\}
\texttt{\override \Score.Staff.\strict-note-spacing = ##t\\}
\texttt{\set \Score.proportionalNotationDuration = #\textbackslash (\textup{ly:make-moment 1/16})\\}
\texttt{new \Staff \{\\}
\texttt{c8[ c \textbackslash clef alto c c \textbackslash grace \{ d16 } c8 c] c4\\}
\texttt{c2 \textbackslash grace \{ c16[ c16] } c2\\}
\texttt{\}}\\
\texttt{new \Staff \{\\}
\texttt{c2 \textbackslash tuplet 3/2 \{ c8 \textbackslash clef bass cis,, c } c4\\}
\texttt{c1\\}
\texttt{>>}
```
Removing brace on first line of piano score

This snippet removes the first brace from a PianoStaff or a GrandStaff.

It may be useful when cutting and pasting the engraved image into existing music.

It uses `\alterBroken`.

```latex
\begin{verbatim}
someMusic = {
    \once \override Staff.Clef.stencil = \Bnatural
    \once \override Staff.TimeSignature.stencil = \Bnatural
    \repeat unfold 3 c1 \break
    \repeat unfold 5 c1 \break
    \repeat unfold 5 c1
}
\score {
    \new PianoStaff
    \new Staff = "right" \relative c' \someMusic
    \new Staff = "left" \relative c' { \clef F \someMusic }
    \layout {
        indent=75
        \context {
            \PianoStaff
            \alterBroken transparent #'(*t) SystemStartBrace
        }
    }
}
\end{verbatim}
```

This snippet removes the first brace from a PianoStaff or a GrandStaff. It may be useful when cutting and pasting the engraved image into existing music. It uses `\alterBroken`. someMusic = { \once \override Staff.Clef.stencil = \Bnatural \once \override Staff.TimeSignature.stencil = \Bnatural \repeat unfold 3 c1 \break \repeat unfold 5 c1 \break \repeat unfold 5 c1 }
\score {
    \new PianoStaff
    \new Staff = "right" \relative c' \someMusic
    \new Staff = "left" \relative c' { \clef F \someMusic }
    \layout {
        indent=75
        \context {
            \PianoStaff
            \alterBroken transparent #'(*t) SystemStartBrace
        }
    }
}
Removing connecting bar lines on StaffGroup, PianoStaff, or GrandStaff

By default, bar lines in StaffGroup, PianoStaff, or GrandStaff groups are connected between the staves, i.e. a SpanBar is printed. This behaviour can be overridden on a staff-by-staff basis.

```
\relative c' {
  \new StaffGroup <<
    \new Staff {
      \once \override Staff.BarLine.allow-span-bar = ##f
      e1 | e | e
    }
    \new Staff {
      \once \override Staff.BarLine.allow-span-bar = ##f
      c1 | c | c
    }
    \new Staff {
      a1 | a | a | a | a
    }
  >>
}
```

Eliminare la prima linea vuota

Il primo rigo vuoto si può togliere dalla partitura impostando la proprietà remove-first di VerticalAxisGroup. Questa impostazione agisce a livello globale se posta nel blocco \layout, a livello locale se posta nel rigo specifico che deve essere tolto. Nel secondo caso, si deve specificare il contesto (Staff si applica solo al rigo corrente) prima della proprietà.

Il rigo inferiore del secondo gruppo di righi non viene rimosso, perché l'impostazione ha effetto solo sul rigo in cui si trova.

```
\layout {
  \context {
```

\Staff \RemoveEmptyStaves
% To use the setting globally, uncomment the following line:
% \override VerticalAxisGroup.remove-first = ##t
}

\new StaffGroup <<
\new Staff \relative c' {
  e4 f g a \break
  c1
}
\new Staff {
  % To use the setting globally, comment this line,
  % uncomment the line in the \layout block above
  \override Staff.VerticalAxisGroup.remove-first = ##t
  R1 \break
  R
}
>>

\new StaffGroup <<
\new Staff \relative c' {
  e4 f g a \break
  c1
}
\new Staff {
  R1 \break
  R
}
>>

\timesig C\text{44}
Stili di pausa
Esistono vari stili di pausa.

\new Staff \relative c {
  \omit Score.TimeSignature
  \cadenzaOn

  \override Staff.Rest.style = #'mensural
  r\longa r\breve r1 r2 r4 r8 r16 s32 s64 s128 s128
  \bar ""
  \break

  \override Staff.Rest.style = #'neomensural
  r\longa r\breve r1 r2 r4 r8 r16 s32 s64 s128 s128
  \bar ""
  \break

  \override Staff.Rest.style = #'classical
  r\longa r\breve r1 r2 r4 r8 r16 r32 r64 r128 s128
  \bar ""
  \break

  \override Staff.Rest.style = #'z
  r\longa r\breve r1 r2 r4 r8 r16 r32 r64 r128 s128
  \bar ""
  \break

  \override Staff.Rest.style = #'default
  r\longa r\breve r1 r2 r4 r8 r16 r32 r64 r128 s128
}

mensural

neomensural

classical

z-style
Rhythmic slashes

In “simple” lead-sheets, sometimes no actual notes are written, instead only “rhythmic patterns” and chords above the measures are notated giving the structure of a song. Such a feature is for example useful while creating/transcribing the structure of a song and also when sharing lead sheets with guitarists or jazz musicians.

The standard support for this using \repeat percent is unsuitable here since the first beat has to be an ordinary note or rest.

This example shows two solutions to this problem, by redefining ordinary rests to be printed as slashes. (If the duration of each beat is not a quarter note, replace the r4 in the definitions with a rest of the appropriate duration).

\% Macro to print single slash

\% Function to print a specified number of slashes

Separating key cancellations from key signature changes

By default, the accidentals used for key cancellations are placed adjacent to those for key signature changes. This behavior can be changed by overriding the 'break-align-orders property of the BreakAlignment grob.
The value of ‘break-align-orders’ is a vector of length 3, with quoted lists of breakable items as elements. This example only modifies the second list, moving key-cancellation before staff-bar; by modifying the second list, break alignment behavior only changes in the middle of a system, not at the beginning or the end.

\new Staff {
\override Score.BreakAlignment.break-align-orders = 
##((left-edge ambitus breathing-sign clef staff-bar
key-cancellation key-signature time-signature custos)

(left-edge ambitus breathing-sign clef key-cancellation
staff-bar key-signature time-signature custos)

(left-edge ambitus breathing-sign clef key-cancellation
key-signature staff-bar time-signature custos))

\key des \major
c'1
\bar "||"
\key bes \major
c'1
}

Impostare il comportamento delle forcelle sulle stanghette
Se la nota che termina una forcella si trova sul primo battito di una battuta, la forcella si ferma prima della stanghetta che precede la nota. Si può controllare questo comportamento modificando la proprietà ‘to-barline.’

\relative c' { 
e4< e2.
e1!
\override Hairpin.to-barline = ##f
e4< e2.
e1!
}

Setting system separators
System separators can be inserted between systems. Any markup can be used, but \slashSeparator has been provided as a sensible default.

\paper {
  system-separator-markup = \slashSeparator
  line-width = 120
}
notes = \relative c' { 
  c1 | c \break 
  c1 | c \break 
  c1 | c 
}

\book { 
  \score { 
    \new GrandStaff << 
    \new Staff notes 
    \new Staff notes 
    >> 
  } 
}
Tweaks and overrides
Showing the same articulation above and below a note or chord

By default, LilyPond does not allow the same articulation (e.g., an accent, a fermata, a flageolet, etc.) to be displayed above and below a note. For example, \c4\fermata\fermata only shows a fermata below. The fermata above gets simply ignored.

However, one can stick scripts (just like fingerings) inside a chord, which means it is possible to have as many articulations as desired. This approach has the advantage that it ignores the stem and positions the articulation relative to the note head. This can be seen in the case of the flageolets in the snippet. To mimic the behaviour of scripts outside a chord, ’add-stem-support would be required.

The solution is thus to write the note as a chord and add the articulations inside of <...>, using the direction modifiers ^ and _ as appropriate.

```
\relative c' {
  <>^"Wrong"
  c2\fermata\fermata % The second fermata is ignored!
  <e d'>2^\flageolet\flageolet
}
```

```
\relative c' {
  <>^"Wrong"
  c2\fermata\fermata % The second fermata is ignored!
  <e d'>2^\flageolet\flageolet
  \stopStaff s1 \startStaff
  <>^"Works if written inside a chord"
  <e \flageolet d'\flageolet>2
  <e \flageolet d'\flageolet>2
  <e \flageolet\flageolet>2
  <e \fermata\fermata>2
}
```

String number extender lines

Make an extender line for string number indications, showing that a series of notes is supposed to be played all on the same string.

```
stringNumberSpanner =
  #(define-music-function (StringNumber) (string?)
  #{
    \override TextSpanner.style = #'solid
    \override TextSpanner.font-size = #-5
    \override TextSpanner.bound-details.left.stencil-align-dir-y = #CENTER
    \override TextSpanner.bound-details.left.text =
    \markup { \circle \number $StringNumber }
  #})
```

```
\relative c {
  \clef "treble_8"
  \stringNumberSpanner "5"
  \textSpannerDown
  a8\startTextSpan
```
Suppressing warnings for clashing note columns

If notes from two voices with stems in the same direction are placed at the same position, and both voices have no shift or the same shift specified, the error message ‘warning: ignoring too many clashing note columns’ will appear when compiling the LilyPond file. This message can be suppressed by setting the ‘ignore-collision’ property of the NoteColumn object to #t. Please note that this does not just suppress warnings but stops LilyPond trying to resolve collisions at all and so may have unintended results unless used with care.

\override NoteColumn.ignore-collision = #t

Time signature in parentheses

The time signature can be enclosed within parentheses.

\relative c' { 
  \override Staff.TimeSignature.stencil = #(\lambda (grob) 
    \bracketify-stencil (ly:time-signature::print grob) Y 0.1 0.2 0.1)) 
  \time 2/4 
  a4 b8 c 
}

Time signature in parentheses - method 3

Another way to put the time signature in parenthesis

\relative c' { 
  \override Staff.TimeSignature.stencil = #(\lambda (grob)
Indicazione di tempo che mostra solo il numeratore (invece della frazione)

Talvolta un’indicazione di tempo non deve mostrare la frazione intera (ad esempio 7/4), ma solo il numeratore (numero 7 in questo caso). Si può ottenere facilmente con \override Staff.TimeSignature.style = #'single-digit, che cambia lo stile in modo permanente. Con \revert Staff.TimeSignature.style, questa impostazione può essere annullata. Per applicare lo stile a cifra singola (single-digit) a una sola indicazione di tempo, si usa il comando \override preceduto da \once.

\relative c'' {  
\time 3/4  
c4 c c  
% Change the style permanently  
\override Staff.TimeSignature.style = #'single-digit  
\time 2/4  
c4 c  
\time 3/4  
c4 c c  
% Revert to default style:  
\revert Staff.TimeSignature.style  
\time 2/4  
c4 c  
% single-digit style only for the next time signature  
\once \override Staff.TimeSignature.style = #'single-digit  
\time 5/4  
c4 c c c c  
\time 2/4  
c4 c  
}

Tuplet bracket and change staff

This snippet shows how to set a tuplet starting in a lower staff and finishing in the upper one.

aigues = \relative c' {  
\time 6/8  
s4.  
\stemDown  
c16[ bes' e]  
}
\stemUp
\g c e
\stemDown
\g8

\basses = \relative c {  
  \time 3/4  
  \clef F  
  \tweak positions #'(4.5 . 9.5)  
  \tweak edge-height #'(1 . -1)  
  \tuplet 7/6 {  
    \c16[ bes' e]  
    \change Staff = \md  
    \stemUp  
    \g[ c e g]  
  }  
  \s4.s8
}

\new PianoStaff  
\with { \omit TimeSignature } <<  
  \new Staff = \md \aigues  
  \new Staff = \mg \basses
>>

\newStaff\makeatletter\newStaff\mkern@-22mu  
\makeatother\makeatletter\makeatother\makeatother

\Modifiche manuali della proprietà della chiave

Cambiando il glifo della chiave, la sua posizione o l’ottavazione non cambia la posizione delle note successive nel rigo. Per far sì che le armature di chiave si trovino sulle linee del rigo corrette, bisogna specificare anche middleCPosition, con valori positivi o negativi che spostano il Do centrale rispettivamente su o giù in senso relativo alla linea centrale del rigo.

Per esempio, \clef "treble_8" equivale a impostare clefGlyph, clefPosition (che regola la posizione verticale della chiave), middleCPosition e clefTransposition. Viene stampata una chiave quando cambia una di queste proprietà, eccetto middleCPosition.

Gli esempi seguenti mostrano le possibilità date dall’impostazione manuale di tali proprietà. Sulla prima linea le modifiche manuali preservano il posizionamento relativo standard di chiavi e note, mentre sulla seconda linea non lo fanno.

{  
  \% The default treble clef  
  \key f \major  
  \c'1
% The standard bass clef
\set Staff.clefGlyph = "clefs.F"
\set Staff.clefPosition = #2
\set Staff.middleCPosition = #6
\set Staff.middleCClefPosition = #6
\key g \major
c'1
% The baritone clef
\set Staff.clefGlyph = "clefs.C"
\set Staff.clefPosition = #4
\set Staff.middleCPosition = #4
\set Staff.middleCClefPosition = #4
\key f \major
c'1
% The standard choral tenor clef
\set Staff.clefGlyph = "clefs.G"
\set Staff.clefPosition = #-2
\set Staff.clefTransposition = #-7
\set Staff.middleCPosition = #1
\set Staff.middleCClefPosition = #1
\key f \major
c'1
% A non-standard clef
\set Staff.clefPosition = #0
\set Staff.clefTransposition = #0
\set Staff.middleCPosition = #-4
\set Staff.middleCClefPosition = #-4
\key g \major
c'1 \break
% The following clef changes do not preserve
% the normal relationship between notes, key signatures
% and clefs:
\set Staff.clefGlyph = "clefs.F"
\set Staff.clefPosition = #2
c'1
\set Staff.clefGlyph = "clefs.G"
c'1
\set Staff.clefGlyph = "clefs.C"
c'1
\set Staff.clefTransposition = #7
c'1
\set Staff.clefTransposition = #0
c'1
\set Staff.clefPosition = #0
c'1
% Return to the normal clef:
\set Staff.middleCPosition = #0
c'1}
Modificare l'aspetto degli abbellimenti di un intero brano

L'aspetto di tutte le espressioni contenute nei blocchi `\grace` di un brano può essere modificato con le funzioni `add-grace-property` e `remove-grace-property`. L'esempio seguente toglie la definizione della direzione di `Stem` nell'abbellimento, in modo che gli abbellimenti non siano sempre rivolti in su, e barra le teste di nota.

```latex
\relative c'' {
\new Staff {
  $(\text{remove-grace-property \ 'Voice \ 'Stem \ 'direction})$
  $(\text{add-grace-property \ 'Voice \ 'NoteHead \ 'style \ 'cross})$
  \new Voice {
    \acciaccatura { f16 } g4
    \grace { d16 e } f4
    \appoggiatura { f,32 g a } e2
  }
}
}
```

Using alternative flag styles

Alternative styles of flag on eighth and shorter notes can be displayed by overriding the `stencil` property of `Flag`. Valid values are `modern-straight-flag`, `old-straight-flag` and `flat-flag`.

```latex
testnotes = {
  \autoBeamOff
  c8 d16 c32 d64 \acciaccatura { c8 } d64 r4
}
\score {
  \relative c' {
    \time 2/4
    \testnotes
    \override \Flag.stencil = \#modern-straight-flag
    \testnotes
    \override \Flag.stencil = \#old-straight-flag
    \testnotes
    \override \Flag.stencil = \#flat-flag
```

```
Using \ly:grob-object to access grobs with \tweak

Some grobs can be accessed “laterally” from within another grob’s callback. These are usually listed as “layout objects” in the “Internal properties” section of a grob-interface. The function \ly:grob-object is used to access these grobs.

Demonstrated below are some ways of accessing grobs from within a NoteHead callback, but the technique is not limited to NoteHeads. However, the NoteHead callback is particularly important, since it is the implicit callback used by the \tweak command.

The example function defined below ("display-grobs") is probably not that useful, but it demonstrates that the grobs are indeed being accessed.

Example console output:

#Grob Accidental () #Grob Stem

#(define (notehead-get-accidental notehead)
  ;; notehead is grob
  (ly:grob-object notehead 'accidental-grob))

#(define (notehead-get-arpeggio notehead)
  ;; notehead is grob
  (let ((notecolumn (notehead-get-notecolumn notehead)))
    (ly:grob-object notecolumn 'arpeggio)))

#(define (notehead-get-notecolumn notehead)
  ;; notehead is grob
  (ly:grob-parent notehead X))

#(define (notehead-get-stem notehead)
  ;; notehead is grob
  (let ((notecolumn (notehead-get-notecolumn notehead)))
    (ly:grob-object notecolumn 'stem)))

#(define (display-grobs notehead)
  ;; notehead is grob
  (let ((accidental (notehead-get-accidental notehead)))
(arpeggio (notehead-get-arpeggio notehead))
(stem (notehead-get-stem notehead)))
(format (current-error-port) "-2k-a\n" (make-string 20="#-))
(for-each
(lambda (x) (format (current-error-port) "-a\n" x))
(list accidental arpeggio stem))))

\relative c' {
  \override NoteHead.before-line-breaking = #display-grobs
  \override NoteHead.before-line-breaking = #display-grobs
    \override NoteHead.before-line-breaking = #display-grobs
  \override NoteHead.before-line-breaking = #display-grobs
  \override NoteHead.before-line-breaking = #display-grobs
  \override NoteHead.before-line-breaking = #display-grobs

Using PostScript to generate special note head shapes

When a note head with a special shape cannot easily be generated with graphic markup, PostScript code can be used to generate the shape. This example shows how a parallelogram-shaped note head is generated.

parallelogram =
#(ly:make-stencil (list 'embedded-ps
"gsave
  currentpoint translate
  newpath
  0 0.25 moveto
  1.3125 0.75 lineto
  1.3125 -0.25 lineto
  0 -0.75 lineto
  closepath
  fill
  gsave"
  (cons 0 1.3125)
  (cons -.75 .75))

myNoteHeads = \override NoteHead.stencil = \parallelogram
normalNoteHeads = \revert NoteHead.stencil

\relative c'' {
  \myNoteHeads
g4 d'
  \normalNoteHeads
  <f, \tweak stencil \parallelogram b e>4 d
}
Using the \tweak command to tweak individual grobs

With the \tweak command, every grob can be tuned directly. Here are some examples of available tweaks.

\relative c' { 
    \time 2/4 
    \set fingeringOrientations = #'(right) 
    < 
        \tweak font-size #3 c 
        \tweak color #red d \tweak font-size #8 -4 
        \tweak style #'cross g 
        \tweak duration-log #2 a 
    >2
}

Dinamiche e segni testuali allineati verticalmente

Tutti gli oggetti DynamicLineSpanner (forcelle e testi di dinamica) sono posti a una distanza minima dal rigo determinata da 'staff-padding. Se si imposta 'staff-padding su un valore abbastanza grande, le dinamiche saranno allineate.

\music = \relative c' { 
    a'2\p b\f 
    e4\p f\f> g, b\p 
    c2\markup { \huge gorgeous } c^-\markup { \huge fantastic } 
}

{
    \music 
    \break 
    \override DynamicLineSpanner.staff-padding = #3 
    \textLengthOn 
    \override TextScript.staff-padding = #1 
    \music 
}
Allineare verticalmente gli ossia e il testo vocale

Questo frammento mostra come usare le proprietà di contesto `alignBelowContext` e `alignAboveContext` per controllare il posizionamento del testo vocale e degli ossia.

```plaintext
\paper {
  \ragged-right = ##t
}
\relative c' <<
  \new Staff = "1" { c4 c s2 }
  \new Staff = "2" { c4 c s2 }
  \new Staff = "3" { c4 c s2 }
  { \skip 2 <<
    \lyrics {
      \set alignBelowContext = #"1"
      lyrics4 below
    }
    \new Staff \with {
      alignAboveContext = #"3"
      fontSize = #-2
      \override StaffSymbol.staff-space = #(magstep -2)
      \remove "Time_signature_engraver"
    }
    \tuplet 6/4 {
      \override TextScript.padding = #3
      c8[^"ossia above" d d e f]
    }
  }
  >>
}

\relative c' 
\new Staff = "4" { c4 c s2 }
\new Staff = "5" { c4 c s2 }
\new Staff = "6" { c4 c s2 }
\relative c' 
\new Staff = "7" { c4 c s2 }
```

Allineare verticalmente gli ossia e il testo vocale
Vertically centering paired figured bass extenders

Where figured bass extender lines are being used by setting `useBassFigureExtenders` to true, pairs of congruent figured bass extender lines are vertically centered if `figuredBassCenterContinuations` is set to true.

```
\relative c' {  
  c8 c b b a a c16 c b b  
  c8 c b b a a c16 c b b  
  c8 c b b a a c c b b  
}
\figures {  
  \set useBassFigureExtenders = ##t  
  <6+ 4 3>4 <6 4 3>8 r  
  <6+ 4 3>4 <6 4 3>8 <4 3>16 r  
  \set figuredBassCenterContinuations = ##t  
  <6+ 4 3>4 <6 4 3>8 r  
  <6+ 4 3>4 <6 4 3>8 <4 3>16 r  
  \set figuredBassCenterContinuations = ##f  
  <6+ 4 3>4 <6 4 3>8 r  
  <6+ 4 3>4 <6 4 3>8 <4 3>8  
}
```

![Diagram of vertically centered figured bass extenders]
Paper and layout

Sezione “Spacing issues” in Guida alla Notazione

Aligning and centering instrument names

The horizontal alignment of instrument names is tweaked by changing the \Staff.InstrumentName.self-alignment-X property. The \layout variables indent and short-indent define the space in which the instrument names are aligned before the first and the following systems, respectively.

\paper { left-margin = 3\cm }

\score {
\new StaffGroup <<

\new Staff \with {
\override InstrumentName.self-alignment-X = #LEFT
instrumentName = \markup \left-column {
"Left aligned"
"instrument name"
}
shortInstrumentName = "Left"
}
{ c''1 \break c''1 }

\new Staff \with {
\override InstrumentName.self-alignment-X = #CENTER
instrumentName = \markup \center-column {
Centered
"instrument name"
}
shortInstrumentName = "Centered"
}
{ g'1 g'1}

\new Staff \with {
\override InstrumentName.self-alignment-X = #RIGHT
instrumentName = \markup \right-column {
"Right aligned"
"instrument name"
}
shortInstrumentName = "Right"
}
{ e'1 e'1 }
>>

\layout {
ragged-right = ##t
Arranging separate lyrics on a single line

Sometimes you may want to put lyrics for different performers on a single line: where there is rapidly alternating text, for example. This snippet shows how this can be done with `\override VerticalAxisGroup.nonstaff-nonstaff-spacing.minimum-distance = ##f`.

```
layout {
  context {
    \Lyrics
      \override VerticalAxisGroup.nonstaff-nonstaff-spacing.minimum-distance = ##f
  }
}

aliceSings = \markup { \smallCaps "Alice" }
eveSings = \markup { \smallCaps "Eve" }
```

```
<<
  \new Staff <<
  \new Voice = "alice" {
    f'4'\aliceSings g' r2 |
    s1 |
    f'4'\aliceSings g' r2 |
    s1 | \break
    \%
    ...
  }

  \voiceOne
```
s2 a'8\aliceSings a' b'4 |
\oneVoice
g''1
}
\new Voice = "eve" {
s1 |
a'2\eveSings g' |
s1 |
a'2\eveSings g'
%
...
\voiceTwo
f'4\eveSings a'8 g' f'4 e' |
\oneVoice
s1
}
>>
\new Lyrics \lyricsto "alice" {
may -- be
sec -- ond
%
Shut up, you fool!
}
\new Lyrics \lyricsto "eve" {
that the
words are
%
... and then I was like--
}
>>

\set-default-paper-size "a6"

\book {
  \%
  book paper, which is inherited by all children bookparts
}
\paper{
  ragged-last-bottom = ##t
  \%\% Page footer: add a different part-tagline at part last page
  oddFooterMarkup = \markup{
    \column{
      \fill-line{
        \%\% Copyright header field only on book first page.
        \if\on-first-page\fromproperty#'header:copyright
      }
      \fill-line{
        \%\% Part tagline header field only on each part last page.
        \if\on-last-page-of-part\fromproperty#'header:parttagline
      }
      \fill-line{
        \%\% Tagline header field only on book last page.
        \if\on-last-page\fromproperty#'header:tagline
      }
    }
  }
  \%\% book header, which is inherited by the first bookpart
  \header{
    title = "Book title"
    copyright = "Copyright line on book first page"
    parttagline = "Part tagline"
    tagline = "Book tagline"
  }
  \bookpart{
    \%\% a different page breaking function may be used on each part
    \paper{ page-breaking = #ly:minimal-breaking }
    \header{ subtitle = "First part" }
    \markup{ The first book part }
    \markup{ a page break }
    \pageBreak
    \markup{ first part last page }
    \markup{ wordwrap with \texttt{ragged-last-bottom} (see the space below this text) }
  }
  \bookpart{ subtitle = "Second part" }
  { c'4 }
}
Book title
First part

The first book part

a page break

Copyright line on book first page
first part last page
with ragged-last-bottom (see the space below this text)
Changing the staff size

Though the simplest way to resize staves is to use \#(set-global-staff-size xx), an individual staff's size can be changed by scaling the properties 'staff-space and fontSize.

<<
\new Staff {
  \relative c'' {
    \dynamicDown
    c8\ff c c c c c c c
  }
}
\new Staff \with {
  fontSize = #-3
  \override StaffSymbol.staff-space = #(magstep -3)
} {
  \clef bass
  c8 c c c c\f c c c
}
>>
Clip systems

This code shows how to clip (extract) snippets from a full score.

This file needs to be run separately with `-dclip-systems`: the snippets page may not adequately show the results. The result will be files named ‘base-from-start-to-end[-count].eps’.

If system starts and ends are included, they include extents of the System grob, e.g., instrument names.

Grace notes at the end point of the region are not included.

Regions can span multiple systems. In this case, multiple EPS files are generated.

```latex
#(ly:set-option 'clip-systems)
#(define output-suffix "1")

origScore = \score {
  \relative c' {
    \new Staff \with { instrumentName = "Instrument" }
    \c1
    \d1
    \grace \c16 \e1
    \key d \major
    \f1 \break
    \clef bass
    \g,\break
    \fis\break
  }
}

\book {
  \score {
    \origScore
    \layout {
      \% Each clip-region is a (START . END) pair
      \% where both are rhythmic-locations.
      \%
      \% (make-rhythmic-locations BAR-NUMBER NUM DEN)
      \%
      \% means NUM/DEN whole-notes into bar numbered BAR-NUMBER

      clip-regions = #(list
                      (cons
                        (make-rhythmic-location 2 0 1)
                        (make-rhythmic-location 4 0 1))
                      (cons
                        (make-rhythmic-location 0 0 1))
      }
  }
}
(make-rhythmic-location 4 0 1))

(cons
 (make-rhythmic-location 0 0 1)
 (make-rhythmic-location 6 0 1)))

\book {
  \score { \origScore }
  \markup { \bold \fontsize #6 clips }
  \score {
    \lyrics {
      \markup { from-2.0.1-to-4.0.1-clip.eps }
      \markup {
        \epsfile #X #30.0 #\{(format #f "-a-1-from-2.0.1-to-4.0.1-clip.eps"
                            (ly:parser-output-name)) \}
    }
  }
}
}
clips

from-2.0.1-to-4.0.1-clip.eps
Creating blank staves

To create blank staves, generate empty measures then remove the Bar_number_engraver from the Score context, and the Time_signature_engraver, Clef_engraver and Bar_engraver from the Staff context.

```latex
\score {
  \repeat unfold 12 { s1 \break }
}\layout {
  indent = 0\in
  \context {
    \Staff
    \remove "Time_signature_engraver"
    \remove "Clef_engraver"
    \remove "Bar_engraver"
  }
  \context {
    \Score
    \remove "Bar_number_engraver"
  }
}

\% uncomment these lines for "letter" size
\%
\paper {
  #(set-paper-size "letter")
  ragged-last-bottom = ##f
  line-width = 7.5\in
  left-margin = 0.5\in
  bottom-margin = 0.25\in
  top-margin = 0.25\in
}
\%

\% uncomment these lines for "A4" size
\%
\paper {
  #(set-paper-size "a4")
  ragged-last-bottom = ##f
  line-width = 180
  left-margin = 15
  bottom-margin = 10
  top-margin = 10
}
\%}
Demonstrating all headers

All header fields with special meanings.

\header {
  \copyright = "copyright"
  \title = "title"
  \subtitle = "subtitle"
  \composer = "composer"
  \arranger = "arranger"
  \instrument = "instrument"
  \meter = "meter"
  \opus = "opus"
  \piece = "piece"
  \poet = "poet"
  \texidoc = "All header fields with special meanings."
  \copyright = "public domain"
  \enteredby = "jcn"
  \source = "urtext"
}

\layout {


```latex
ragged-right = \###f 

\score { 
  \relative c'' \{ c1 | c | c | c \} 
}

\score { 
  \relative c'' \{ c1 | c | c | c \} 
  \header { 
    title = "localtitle" 
    subtitle = "localsubtitle" 
    composer = "localcomposer" 
    arranger = "localarranger" 
    instrument = "localinstrument" 
    metre = "localmetre" 
    opus = "localopus" 
    piece = "localpiece" 
    poet = "localpoet" 
    copyright = "localcopyright" 
  }
}
```

---

### Setting system separators

System separators can be inserted between systems. Any markup can be used, but `\slashSeparator` has been provided as a sensible default.

```latex
\paper { 
  system-separator-markup = \slashSeparator 
  line-width = 120 
}
```

```latex
notes = \relative c' { 
```
\book {
  \score {
    \new GrandStaff <<
    \new Staff \notes
    \new Staff \notes
  }>
}
}
Table of contents

A table of contents is included using \markuptlist\table-of-contents. The TOC items are added with the \tocItem command.

#(set-default-paper-size "a6")

\book {
\markuptlist\table-of-contents
\pageBreak
\tocItem \markup { The first score }
\score {
  {c'1 \pageBreak
   \mark \default \tocItem \markup { Mark A }
   d'1
  }
}
\pageBreak
\tocItem \markup { The second score }
\score {
  { e'1 }
  \header { piece = "Second score" }
}
}

Table of Contents

The first score 2
Mark A 3
The second score 4

\timesig.C44/noteheads.s0/clefs.G
\noteheads.s0
\clefs.G
\A
Vertical aligned StaffGroups without connecting SystemStartBar

This snippet shows how to achieve vertically aligned StaffGroups with a SystemStartBar for each StaffGroup, but without connecting them.

\bash
(set-global-staff-size 18)

\lilypond
\paper {
    indent = 0
    ragged-right = ##f
    print-all-headers = ##t
}
#'(SystemStartBrace (SystemStartBracket a b))

\context {
  \Score
  \remove Text_mark_engraver
  \remove Staff_collecting_engraver
  \override SystemStartBrace.style = #'bar-line
  \omit SystemStartBar
  \override SystemStartBrace.padding = #-0.1
  \override SystemStartBrace.thickness = #1.6
  \override StaffGrouper.staffgroup-staff-spacing.basic-distance = #15
}

%%% EXAMPLE

txt = \lyricmode {
  Wer4 nur den lie -- ben Gott läßt wal2 -- ten4
  und4 hof -- fet auf ihn al -- le Zeit2.
}

% First StaffGroup "exercise"

eI = \relative c' {
  \textMark \markup {
    \bold Teacher:
    This is a simple setting of the choral. Please improve it.
  }
  \key a \minor
  \time 4/4
  \voiceOne

  \partial 4
  e4
  a b c b
  a b gis2
  e4\fermata g! g f
  e a a gis
  a2.\fermata
  \bar ":|."\n}

eII = \relative c' {
  \key a \minor
  \time 4/4
  \voiceTwo
  \partial 4
  c4
}

e e e gis
a f e2
b4 b d d
c c d d
c2.
\bar ":.|
}
eIII =
\relative c' {
  \key a \minor
  \time 4/4
  \clef bass
  \voiceOne

  \partial 4
  a4
c b a b
c d b2
gis4 g g b
c a f e
e2.
}
eIV =
\relative c' {
  \key a \minor
  \time 4/4
  \clef bass
  \voiceTwo

  \partial 4
  a,4
  a' gis a e
  a, d e2
e,4\fermata e' b g
c f d e
  a,2.\fermata
  \bar ":.|
}
exercise =
\new StaffGroup = "exercise"
<<

\new Staff
<<
  \new Voice \eI
  \new Voice \eII
>>

\new Lyrics \txt
\new Staff
<<
\new Voice \eIII
\new Voice \eIV
>>

% Second StaffGroup "simple Bach"

sbI =
\relative c' {
  \textMark \markup \bold "Pupil:" Here's my version! }
\key a \minor
\time 4/4
\voiceOne

  \partial 4
e4
  a b c b
  a b gis2
e4\fermata g! g f
e a a gis
  a2.\fermata
  \bar ".:|."}

sbII =
\relative c' {
  \key a \minor
  \time 4/4
  \voiceTwo
  \partial 4
c8 d
e4 e e8 f g4
  f f e2
  b4 b8 c d4 d
e8 d c4 b8 c d4
c2.
  \bar ".:|."}

sbIII =
\relative c' {
  \key a \minor
  \time 4/4
  \clef bass
  \voiceOne

  \partial 4
  a8 b
  c4 b a b8 c
\relative c' {
  \key a \minor
  \time 4/4
  \clef bass
  \voiceTwo

  \partial 4
  a,4
  a' gis a e
  f8 e d4 e2
  e,4 \fermata e' b a8 g
  c4 f8 e d4 e
  a,2 \fermata
  \bar "|:"}
}

\new StaffGroup = "simple Bach"

\new Staff
\new Voice \sbI
\new Voice \sbII

\new Lyrics \txt

\new Staff
\new Voice \sbIII
\new Voice \sbIV

% Third StaffGroup "chromatic Bach"

cbI = \relative c' {
  \textMark \markup {
    \bold "Teacher:"
    \column {
      "Well, you simply copied and transposed a version of J.S.Bach."
      "Do you know this one?"
    }
  }
}
\key a \minor
\time 4/4
\voiceOne

\partial 4
e4
a \ b \ c \ b
a \ b \ gis4. \ fis8
e4\fermata g' g f
e a \ a8 \ b \ gis4
a2.\fermata
\bar ":\|." 

\cbII =
\relative c' { \key a \minor \time 4/4 \voiceTwo \partial 4
c8 \ d
e4 \ e \ e8 \ fis \ gis4
a8 \ g' f14 \ e2
b4 \ e \ e \ d
d8[ cis] \ d \ dis \ e \ fis \ e4
e2.
\bar ":\|." 

\cbIII =
\relative c' { \key a \minor \time 4/4 \clef bass \voiceOne

\partial 4
a8 \ b
c[ b] \ a \ gis8 \ a4 \ d, 
e8[ e'] \ d \ c \ b4. \ a8
gis4 \ b \ c \ d8 \ c
b[ a] \ a \ b \ c \ b \ c16 \ d
\c2. 
}

\cbIV =
\relative c' { \key a \minor \time 4/4 \clef bass \voiceTwo
\partial 4
a^4
c, e a, b
c d e2
e4/fermata e a b8 c
gis[ g] fis f e dis e4
a,2./fermata
\bar "|.|"

chromaticBach =
new StaffGroup = "chromatic Bach"
<<
new Staff
<<
new Voice cbI
new Voice cbII
>>

new Lyrics txt

new Staff
<<
new Voice cbIII
new Voice cbIV
>>
>>

% Score

\score {
<<
exercise
simpleBach
chromaticBach
>>
header {
title = \markup
\column {
\combine \null \vspace #1
"Exercise: Improve the given choral"
" "
}
}
layout {
context {
Lyrics
\override LyricText.X-offset = #-1
}
Exercise: Improve the given choral

Teacher: This is a simple setting of the choral. Please improve it.

Pupil: Here's my version!

Teacher: Well, you simply copied and transposed a version of J.S.Bach. Do you know this one?
ten und hoffet auf ihn alle Zeit

ten und hoffet auf ihn alle Zeit

ten und hoffet auf ihn alle Zeit
Adding the current date to a score

With a little Scheme code, the current date can easily be added to a score.

```
% first, define a variable to hold the formatted date:
date = $(strftime "%d-%m-%Y" (localtime (current-time)))

% use it in the title block:
header {
  title = "Including the date!"
  subtitle = \date
}

\score {
  \relative c' {
    c4 c c c
  }
}

% and use it in a \markup block:
\markup {
  \date
}
```

Including the date!
18-11-2023

18-11-2023

Aligning and centering instrument names

The horizontal alignment of instrument names is tweaked by changing the `Staff.InstrumentName.self-alignment-X` property. The `\layout` variables `indent` and `short-indent` define the space in which the instrument names are aligned before the first and the following systems, respectively.

```
\paper { left-margin = 3cm }

\score {
  \new StaffGroup <<
    \new Staff \with {
      \override InstrumentName.self-alignment-X = #LEFT
      instrumentName = \markup \left-column {
        "Left aligned"
        "instrument name"
    }
  }
```

\new StaffGroup <<
    \new Staff \with {
      \override InstrumentName.self-alignment-X = #LEFT
      instrumentName = \markup \left-column {
        "Left aligned"
        "instrument name"
    }
  }
```
}\}
shortInstrumentName = "Left"
}\}
\new Staff \with {\new Staff \with {
\override InstrumentName.self-alignment-X = #CENTER
instrumentName = \markup \center-column {Centered
"instrument name"
}
shortInstrumentName = "Centered"
}
{ g'1 g'1} \new Staff \with {
\override InstrumentName.self-alignment-X = #RIGHT
instrumentName = \markup \right-column {"Right aligned"
"instrument name"
}
shortInstrumentName = "Right"
}
{ e'1 e'1 }
>>\layout {ragged-right = ##t
indent = 4\cm
short-indent = 2\cm
}
Demonstrating all headers

All header fields with special meanings.

\header {
  \copyright = "copyright"
  \title = "title"
  \subtitle = "subtitle"
  \composer = "composer"
  \arranger = "arranger"
  \instrument = "instrument"
  \meter = "meter"
  \opus = "opus"
  \piece = "piece"
  \poet = "poet"
  \texidoc = "All header fields with special meanings."
  \copyright = "public domain"
  \enteredby = "jcn"
  \source = "urtext"
}

\layout {
  \ragged-right = ##f
}

\score {
  \relative c' \{ c1 | c | c | c \}
}

\score {
  \relative c' \{ c1 | c | c | c \}
  \header {
    \title = "localtitle"
    \subtitle = "localsubtitle"
    \composer = "localcomposer"
    \arranger = "localarranger"
    \instrument = "localinstrument"
    \metre = "localmetre"
    \opus = "localopus"
    \piece = "localpiece"
    \poet = "localpoet"
  }
}
Outputting the version number

It is possible to print the version number of LilyPond in markup.
\markup { Processed with LilyPond version #(lilypond-version) }

Processed with LilyPond version 2.24.3
Spacing

Sezione “Spacing issues” in Guida alla Notazione

Adjusting lyrics vertical spacing

This snippet shows how to bring the lyrics line closer to the staff.

% Default layout:
<<
\new Staff \new Voice = melody \relative c' {
  c4 d e f
  g4 f e d
  c1
}
\new Lyrics \lyricstoforward { aa aa aa aa aa aa aa aa }

\new Staff {
  \new Voice = melody \relative c' {
    c4 d e f
    g4 f e d
    c1
  }
}
% Reducing the minimum space below the staff and above the lyrics:
\new Lyrics \with {
  \override VerticalAxisGroup.nonstaff-relatedstaff-spacing = #'((basic-distance . 1))
}
\lyricstoforward { aa aa aa aa aa aa aa aa }
>>

Far sì che la diteggiatura appaia dentro il rigo

Per impostazione predefinita, le diteggiature orientate verticalmente sono poste fuori dal rigo; questo comportamento tuttavia può essere disabilitato. Occorre fare attenzione alle situazioni in cui le diteggiature e i gambi sono rivolti nella stessa direzione: normalmente le diteggiature evitano soltanto i gambi con travature. Questa impostazione predefinita può essere cambiata in modo da evitare tutti i gambi oppure nessuno. L’esempio seguente mostra queste due opzioni, così come tornare al comportamento predefinito.

relative c' {
  <c-1 e-2 g-3 b-5>2
  \override Fingering.staff-padding = #'()
  <c-1 e-2 g-3 b-5>4 g'-0
  a8[-1 b]-2 g'-0 r
Page label

Page labels may be placed inside music or at top-level, and referred to in markups.

```latex
\label license
\markup "This snippet is available under the Creative Commons Public Domain Dedication license."

{\
  \repeat volta 2 {\
    \label startRepeat\
    \repeat unfold 20 { c'2 2 }\
    \pageBreak\
    2 2
  }
}\textEndMark \markup {\
  \with-link #'startRepeat \line {\
    To page \page-ref #'startRepeat "0" "?"
  }\
}
```

See page \page-ref #'license "0" "?" for licensing information.

This snippet is available under the Creative Commons Public Domain Dedication license.
Proportional strict notespacing

If strict-note-spacing is set spacing of notes is not influenced by bars or clefs within a system. Rather, they are placed just before the note that occurs at the same time. This may cause collisions.

```lilypond
\relative c' {
\override Score.SpacingSpanner.strict-note-spacing = ##t
\set Score.proportionalNotationDuration = #((ly:make-moment 1/16)
\new Staff {
  c8[ c \clef alto c c \grace { d16 } c8 c] c4
  c2 \grace { c16[ c16] } c2
}
\new Staff {
  c2 \tuplet 3/2 { c8 \clef bass cis,, c } c4
  c1
}
}
```

Dinamiche e segni testuali allineati verticalmente

Tutti gli oggetti DynamicLineSpanner (forcelle e testi di dinamica) sono posti a una distanza minima dal rigo determinata da 'staff-padding. Se si imposta 'staff-padding su un valore abbastanza grande, le dinamiche saranno allineate.

```lilypond
music = \relative c' {
  a'2 \p b\f
  e4 \p f\f> g, b\p
  c2^-\markup { \huge gorgeous } c^-\markup { \huge fantastic }
}
```
Allineare verticalmente gli ossia e il testo vocale

Questo frammento mostra come usare le proprietà di contesto alignBelowContext e alignAboveContext per controllare il posizionamento del testo vocale e degli ossia.

```
\relative c' <<
\new Staff = "1" { c4 c s2 }
\new Staff = "2" { c4 c s2 }
\new Staff = "3" { c4 c s2 }
{ \skip 2 }<<
\lyrics {
  \set alignBelowContext = "1"
  lyrics4 below
}
\new Staff \with {
  alignAboveContext = "3"
  fontSize = -2
  \override StaffSymbol.staff-space = #(magstep -2)
  \remove "Time_signature_engraver"
} \
\tuplet 6/4 {
  \override TextScript.padding = #3
  c8 ["ossia above" d e d e f]
}
```

Allineare verticalmente gli ossia e il testo vocale
>>

lyrics below

ossia above
**MIDI**

Sezione “Creating MIDI output” in *Guida alla Notazione*

**Impostare l’output MIDI su un canale per voce**

Nella creazione del file di output MIDI, il comportamento predefinito prevede che ogni rigo sia assegnato a un canale MIDI, con tutte le voci del rigo amalgamate in un canale. Ciò diminuisce il rischio di esaurire i canali MIDI disponibili, dato che ce ne sono solo 16 per traccia.

Tuttavia, spostando `Staff_performer` nel contesto `Voice`, ogni voce in un rigo può avere il proprio canale MIDI, come è illustrato nell’esempio seguente: sebbene le voci siano sullo stesso rigo, vengono creati due canali MIDI, ciascuno con un diverso strumento MIDI (`midiInstrument`).

```plaintext
\score { \new Staff << \new Voice \relative c' \{ \set midiInstrument = #"flute" \voiceOne \
    \key g \major \time 2/2 
    r2 g-"Flute" ~ 
    g fis ~ 
    fis4 g8 fis e2 ~ 
    e4 d8 cis d2 
}\new Voice \relative c' \{ 
    \set midiInstrument = #"clarinet" 
    \voiceTwo 
    b1-"Clarinet" 
    a2. b8 a 
    g2. fis8 e 
    fis2 r 
}\layout { } \midi { \context { \Staff \remove "Staff_performer" } \context { \Voice \consists "Staff_performer" } \tempo 2 = 72 } }
```

![MIDI Example](image-url)
Modificare il tempo senza mostrare l’indicazione metronomica

Per cambiare il tempo del file MIDI senza che appaia l’indicazione metronomica, basta renderla invisibile.

```
\score {
    \new Staff \relative c' {
        \tempo 4 = 160
        c4 e g b
        c4 b d c
        \set Score.tempoHideNote = ##t
        \tempo 4 = 96
        d,4 fis a cis
        d4 cis e d
    }
    \layout {}
    \midi {}
}
```

![Scheda musicale](image)

Creare dinamiche personalizzate nell’output MIDI

L’esempio seguente mostra come creare un segno di dinamica, non incluso nell’elenco predefinito, e assegnargli un valore specifico così che possa essere usato per cambiare l’output MIDI.

```
#(define (myDynamics dynamic)
    (if (equal? dynamic "rfz")
        0.9
        (default-dynamic-absolute-volume dynamic)))
```

```
\score {
    \new Staff {
        \set Staff.midiInstrument = "cello"
        \set Score.dynamicAbsoluteVolumeFunction = #myDynamics
        \new Voice {
            \relative {
                a'4\pp b c-\rfz
            }
        }
    }
    \layout {}
    \midi {}
}
```

![Scheda musicale](image)
Customized drum notation in printed and MIDI output

Customized drum “pitch” names (suitable for a custom drum style, for example) may be used both in printed and MIDI output by defining such variables as drumPitchNames, drumStyleTable and midiDrumPitches, as demonstrated here. In short, this snippet:

- defines some "pitch" names,
- defines how they will be rendered,
- tells LilyPond to use them for layout,
- assigns pitches to the names,
- tells LilyPond to use them for MIDI output.

%% This snippet tries to amend
%% NR 2.5.1 Common notation for percussion - Custom percussion staves
%% http://lilypond.org/doc/v2.18/Documentation/notation/common-notation-for-percussion#custom-percussion-staves

%%% This snippet tries to amend
%%% NR 2.5.1 Common notation for percussion - Custom percussion staves
%%% http://lilypond.org/doc/v2.18/Documentation/notation/common-notation-for-percussion#custom-percussion-staves

%%% To use custom drum pitch names for your score and midi you need to follow
%%% this route:

%%%% LAYOUT:
%%%%
%%
%%% (1) Define a name and put it in 'drumPitchNames'
%%% This can be done at toplevel with
%%% drumPitchNames.my-name = #'my-name
%%% It's possible to add an alias as well.
%%
%%% (2) Define how it should be printed
%%% Therefore put them into a top-level list, where each entry should
%%% be of the form:
%%
%%%   (my-name
%%%    note-head-style-or-default
%%%    articulation-type-or-#f
%%%    staff-position)

%%% Example:
%%% #define my-style
%%% '(
%%%   (my-name default tenuto -1)
%%%   ; ...
%%% ))

%%% (3) Tell LilyPond to use these custom definitions, with
%%% drumStyleTable = #(alist->hash-table my-style)
%%% in a \layout or \with block

%%% Now we're done for layout. Here is a short but complete example:
%%% new DrumStaff
%%% \with { drumStyleTable = #(alist->hash-table my-style) }
%%% \drummode { my-name }


%% MIDI:
%%%%%%%%%%%%%%%%%%
%%
%% (1) Again at top-level, assign a pitch to your custom note name
%% midiDrumPitches.my-name = ges
%% Note that you have to use the name, which is in drumPitchNames, no alias
%% (2) Tell LilyPond to use this pitch(es), with
%% drumPitchTable = #(alist->hash-table midiDrumPitches)
%%
%% Example:
%% 
%% \score {
%% \new DrumStaff
%% \with {
%%   drumStyleTable = #(alist->hash-table my-style)
%%   drumPitchTable = #(alist->hash-table midiDrumPitches)
%% }
%% \drummode { my-name4 }
%% \layout {}
%% \midi {}
%% }
%%

%%%%%%%%%%%%%%%%%%
%% TESTING
%%%%%%%%%%%%%%%%%%
%%
%% To test whether all is fine, run the following sequence in terminal:
%% lilypond my-file.ly
%% midi2ly my-file.midi
%% gedit my-file-midi.ly
%%
%% This will do the following:
%% 1. create pdf and midi
%% 2. transform the midi back to a .ly-file
%%   (note: midi2ly is not always good in correctly identifying enharmonic pitches)
%% 3. open this file in gedit (or use another editor)
%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% FULL EXAMPLE
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

drumPitchNames.dbass   = #'dbass
drumPitchNames.db     = #'dbass % 'db is in use already
drumPitchNames.dbassmute = #'dbassmute
drumPitchNames.dbm    = #'dbassmute
drumPitchNames.do     = #'dopen
drumPitchNames.dopenmute = #'dopenmute
drumPitchNames.dom    = #'dopenmate
drumPitchNames.dslap  = #'dslap
drumPitchNames.ds     = #'dslap
drumPitchNames.dslapmute = #'dslapmute
drumPitchNames.dsm = #'dslapmute

#(define djembe
  '((dbass  default  f  -2)
     (dbassmute default stopped -2)
     (dopen   default  f   0)
     (dopenmute default stopped 0)
     (dslap   default  f   2)
     (dslapmute default stopped 2)))

midiDrumPitches.dbass = g
midiDrumPitches.dbassmute = fis
midiDrumPitches.dopen = a
midiDrumPitches.dopenmute = gis
midiDrumPitches.dslap = b
midiDrumPitches.dslapmute = ais

one = \drummode { r4 dba4 do ds r dbm dom dsm }

\score {
\new DrumStaff
  \with {
    \override StaffSymbol.line-count = #3
    instrumentName = #'Djembe
    drumStyleTable = #(alist->hash-table djembe)
    drumPitchTable = #(alist->hash-table midiDrumPitches)
  }
  \one
  \layout {}
  \midi {}
}

Djembe

Demo MidiInstruments

Problem: How to know which midiInstrument would be best for your composition?

Solution: A LilyPond demo file.

\header {
  \title = "Demo of all midi sounds"
  \arranger = "Myself "
}

baseMelody = \relative c' { c4.mf g c16 b' c d
  e16 d e f g4 g'4 r
  R1}

melody = {
\September 150
\setBaseMelody
\
\score {\
  \newStaff\
  \newVoice {\
  \layout { }\
}
}
\setStaff.midiInstrument = #"acoustic grand" \melody
\setStaff.midiInstrument = #"bright acoustic" \melody
\setStaff.midiInstrument = #"electric grand" \melody
\setStaff.midiInstrument = #"honky-tonk" \melody
\setStaff.midiInstrument = #"electric piano 1" \melody
\setStaff.midiInstrument = #"electric piano 2" \melody
\setStaff.midiInstrument = #"harpsichord" \melody
\setStaff.midiInstrument = #"clav" \melody
\setStaff.midiInstrument = #"celesta" \melody
\setStaff.midiInstrument = #"glockenspiel" \melody
\setStaff.midiInstrument = #"music box" \melody
\setStaff.midiInstrument = #"vibraphone" \melody
\setStaff.midiInstrument = #"marimba" \melody
\setStaff.midiInstrument = #"xylophone" \melody
\setStaff.midiInstrument = #"tubular bells" \melody
\setStaff.midiInstrument = #"dulcimer" \melody
\setStaff.midiInstrument = #"drawbar organ" \melody
\setStaff.midiInstrument = #"percussive organ" \melody
\setStaff.midiInstrument = #"rock organ" \melody
\setStaff.midiInstrument = #"church organ" \melody
\setStaff.midiInstrument = #"reed organ" \melody
\setStaff.midiInstrument = #"accordion" \melody
\setStaff.midiInstrument = #"harmonica" \melody
\setStaff.midiInstrument = #"concertina" \melody
\setStaff.midiInstrument = #"acoustic guitar (nylon)" \melody
\setStaff.midiInstrument = #"acoustic guitar (steel)" \melody
\setStaff.midiInstrument = #"electric guitar (jazz)" \melody
\setStaff.midiInstrument = #"electric guitar (clean)" \melody
\setStaff.midiInstrument = #"electric guitar (muted)" \melody
\setStaff.midiInstrument = #"overdriven guitar" \melody
\setStaff.midiInstrument = #"distorted guitar" \melody
\setStaff.midiInstrument = #"acoustic bass" \melody
\setStaff.midiInstrument = #"electric bass (finger)" \melody
\setStaff.midiInstrument = #"electric bass (pick)" \melody
\setStaff.midiInstrument = #"fretless bass" \melody
\setStaff.midiInstrument = #"slap bass 1" \melody
\setStaff.midiInstrument = #"slap bass 2" \melody
\set Staff.midiInstrument = #"synth bass 1" \melody
\set Staff.midiInstrument = #"synth bass 2" \melody
\set Staff.midiInstrument = #"violin" \melody
\set Staff.midiInstrument = #"viola" \melody
\set Staff.midiInstrument = #"cello" \melody
\set Staff.midiInstrument = #"contrabass" \melody
\set Staff.midiInstrument = #"tremolo strings" \melody
\set Staff.midiInstrument = #"pizzicato strings" \melody
\set Staff.midiInstrument = #"orchestral harp" \melody
\set Staff.midiInstrument = #"timpani" \melody
\set Staff.midiInstrument = #"string ensemble 1" \melody
\set Staff.midiInstrument = #"string ensemble 2" \melody
\set Staff.midiInstrument = #"synthstrings 1" \melody
\set Staff.midiInstrument = #"synthstrings 2" \melody
\set Staff.midiInstrument = #"choir aahs" \melody
\set Staff.midiInstrument = #"voice oohs" \melody
\set Staff.midiInstrument = #"synth voice" \melody
\set Staff.midiInstrument = #"orchestra hit" \melody
\set Staff.midiInstrument = #"trumpet" \melody
\set Staff.midiInstrument = #"trombone" \melody
\set Staff.midiInstrument = #"tuba" \melody
\set Staff.midiInstrument = #"muted trumpet" \melody
\set Staff.midiInstrument = #"french horn" \melody
\set Staff.midiInstrument = #"brass section" \melody
\set Staff.midiInstrument = #"synthbrass 1" \melody
\set Staff.midiInstrument = #"synthbrass 2" \melody
\set Staff.midiInstrument = #"soprano sax" \melody
\set Staff.midiInstrument = #"alto sax" \melody
\set Staff.midiInstrument = #"tenor sax" \melody
\set Staff.midiInstrument = #"baritone sax" \melody
\set Staff.midiInstrument = #"oboe" \melody
\set Staff.midiInstrument = #"english horn" \melody
\set Staff.midiInstrument = #"bassoon" \melody
\set Staff.midiInstrument = #"clarinet" \melody
\set Staff.midiInstrument = #"piccolo" \melody
\set Staff.midiInstrument = #"flute" \melody
\set Staff.midiInstrument = #"recorder" \melody
\set Staff.midiInstrument = #"pan flute" \melody
\set Staff.midiInstrument = #"blown bottle" \melody
\set Staff.midiInstrument = #"shakuhachi" \melody
\set Staff.midiInstrument = #"whistle" \melody
\set Staff.midiInstrument = #"ocarina" \melody
\set Staff.midiInstrument = #"lead 1 (square)" \melody
\set Staff.midiInstrument = #"lead 2 (sawtooth)" \melody
\set Staff.midiInstrument = #"lead 3 (calliope)" \melody
\set Staff.midiInstrument = #"lead 4 (chiff)" \melody
\set Staff.midiInstrument = #"lead 5 (charang)" \melody
\set Staff.midiInstrument = #"lead 6 (voice)" \melody
\set Staff.midiInstrument = #"lead 7 (fifths)" \melody
\set Staff.midiInstrument = #"lead 8 (bass+lead)" \melody
\set Staff.midiInstrument = #"pad 1 (new age)" \melody
\set Staff.midiInstrument = #"pad 2 (warm)" \melody
\set Staff.midiInstrument = \"pad 3 (polysynth)\" \melody
\set Staff.midiInstrument = \"pad 4 (choir)\" \melody
\set Staff.midiInstrument = \"pad 5 (bowed)\" \melody
\set Staff.midiInstrument = \"pad 6 (metallic)\" \melody
\set Staff.midiInstrument = \"pad 7 (halo)\" \melody
\set Staff.midiInstrument = \"pad 8 (sweep)\" \melody
\set Staff.midiInstrument = \"fx 1 (rain)\" \melody
\set Staff.midiInstrument = \"fx 2 (soundtrack)\" \melody
\set Staff.midiInstrument = \"fx 3 (crystal)\" \melody
\set Staff.midiInstrument = \"fx 4 (atmosphere)\" \melody
\set Staff.midiInstrument = \"fx 5 (brightness)\" \melody
\set Staff.midiInstrument = \"fx 6 (goblins)\" \melody
\set Staff.midiInstrument = \"fx 7 (echoes)\" \melody
\set Staff.midiInstrument = \"fx 8 (sci-fi)\" \melody
\set Staff.midiInstrument = \"sitar\" \melody
\set Staff.midiInstrument = \"banjo\" \melody
\set Staff.midiInstrument = \"shamisen\" \melody
\set Staff.midiInstrument = \"koto\" \melody
\set Staff.midiInstrument = \"kalimba\" \melody
\set Staff.midiInstrument = \"bagpipe\" \melody
\set Staff.midiInstrument = \"fiddle\" \melody
\set Staff.midiInstrument = \"shanai\" \melody
\set Staff.midiInstrument = \"tinkle bell\" \melody
\set Staff.midiInstrument = \"agogo\" \melody
\set Staff.midiInstrument = \"steel drums\" \melody
\set Staff.midiInstrument = \"woodblock\" \melody
\set Staff.midiInstrument = \"taiko drum\" \melody
\set Staff.midiInstrument = \"melodic tom\" \melody
\set Staff.midiInstrument = \"synth drum\" \melody
\set Staff.midiInstrument = \"reverse cymbal\" \melody
\set Staff.midiInstrument = \"guitar fret noise\" \melody
\set Staff.midiInstrument = \"breath noise\" \melody
\set Staff.midiInstrument = \"seashore\" \melody
\set Staff.midiInstrument = \"bird tweet\" \melody
\set Staff.midiInstrument = \"telephone ring\" \melody
\set Staff.midiInstrument = \"helicopter\" \melody
\set Staff.midiInstrument = \"applause\" \melody
\set Staff.midiInstrument = \"gunshot\" \melody

}

Demo of all midi sounds
Modificare l’equalizzazione predefinita degli strumenti MIDI

L’equalizzatore predefinito degli strumenti MIDI può essere modificato impostando la proprietà instrumentEqualizer nel contesto Score come una procedura Scheme definita dall’utente che usi il nome dello strumento MIDI come argomento insieme a una coppia di frazioni indicanti i volumi minimi e massimi da applicare a quello specifico strumento.

L’esempio seguente imposta i volumi massimo e minimo per il flauto e per il clarinetto.

```scheme
#(define my-instrument-equalizer-alist '())

#(set! my-instrument-equalizer-alist
 (append
  '("flute" . (0.7 . 0.9))
  ("clarinet" . (0.3 . 0.6)))
 my-instrument-equalizer-alist))

#(define (my-instrument-equalizer s)
 (let ((entry (assoc s my-instrument-equalizer-alist)))
  (if entry
   (cdr entry))))

\score {
  <<
    \new Staff {
      \key g \major
      \time 2/2
      \set Score.instrumentEqualizer = #my-instrument-equalizer
      \set Staff.midiInstrument = "flute"
      \new Voice \relative { r2 g'' \mp g fis~ 4 g8 fis e2~ 4 d8 cis d2 }
    }
    \new Staff {
      \key g \major
      \set Staff.midiInstrument = "clarinet"
      \new Voice \relative { b'1 \p a2. b8 a g2. fis8 e fis2 r }
    }
  >>
  \layout { } 
  \midi { } 
  }
}
Templates

Modello per notazione antica – trascrizione moderna di musica gregoriana

Questo esempio mostra come realizzare una trascrizione moderna di musica gregoriana. La musica gregoriana non presenta la suddivisione in misure né gambi; impiega soltanto le teste della minima e della semiminima, e dei segni appositi che indicano pause di diversa lunghezza.

\include "gregorian.ly"

\set Score.timing = ##f  
\relative c' {  
  f4 a2 \divisioMinima  
  g4 b a2 f2 \divisioMaior  
  g4( f) f( g) a2 \finalis  
}

\lyricmode {  
  Lo -- rem ip -- sum do -- lor sit a -- met  
}

\new GregorianTranscriptionStaff <<  
\new GregorianTranscriptionVoice = "melody" \chant  
\new GregorianTranscriptionLyrics = "one" \lyricsto melody \verba  
>>  

Lorem ipsum dolor sit amet

Modello per salmo anglicano

Questo modello presenta un modo per impostare un salmo anglicano. Mostra anche come le strofe possano essere aggiunte come testo separato al di sotto della musica. Le due strofe sono scritte con stili diversi per illustrare le varie possibilità.

\relative g' {  
  g1 | c2 b | a1 | \bar "||"  
  a1 | d2 c | c b | c1 | \bar "||"  
}

\relative c' {  
  e1 | g2 g | f1 |  
  f1 | f2 e | d d | e1 |  
}

\relative a {  
  c1 | c2 c | c1 |  
  d1 | g,2 g | g g | g1 |  
}
BassMusic = \relative c { 
|c\ | e\ | f\ |
|d\ | b\ | g'\ | c,1 |
}

global = { 
\time 2/2 
}

dot = \markup { 
\raise #0.7 \musicglyph "dots.dot" 
}

tick = \markup { 
\raise #1 \fontsize #-5 \musicglyph "scripts.rvarcomma" 
}

% Use markup to center the chant on the page
\markup {
\fill-line {
\score { % centered
<< 
\new ChoirStaff <<
\new Staff <<
\global
\clef "treble"
\new Voice = "Soprano" <<
\voiceOne
\SopranoMusic
>>
\new Voice = "Alto" <<
\voiceTwo
\AltoMusic
>>

\new Staff <<
\clef "bass"
\global
\new Voice = "Tenor" <<
\voiceOne
\TenorMusic
>>
\new Voice = "Bass" <<
\voiceTwo
\BassMusic
>>

>>

\layout {
\context { }
\Score
\override SpacingSpanner.base-shortest-duration = #(ly:make-moment 1/2)
}
\context {
\Staff
\remove "Time_signature_engraver"
}
}% End score
}
}% End markup

\markup {
\fill-line {
\column {
\left-align {
\null \null \null
\line {
\fontsize #5 0
\fontsize #3 come
\let us \bold sing | unto \dot the | Lord : let
}
\line {
\us heartily
\concat { re \bold joice }
\in the | strength of | our
}
\line {
\sal | vation.
}
\null
\line {
\hspace #2.5 8. Today if ye will hear his voice *
}
\line {
\concat { \bold hard en }
\tick not your \tick hearts : as in the pro-
}
\line {
\vocation * and as in the \bold day of tempt- \tick
}
\line {
-ation \tick in the \tick wilderness.
}
}
}
}
O come let us sing | unto the | Lord: let us heartily rejoice in the | strength of | our salvation.

8. Today if ye will hear his voice * harden not your hearts: as in the provocation * and as in the day of temptation * and in the wilderness.

Modello per inno

Il codice seguente presenta un modo di impostare un inno in cui ogni verso inizia e finisce con una misura parziale. Mostra anche come aggiungere delle strofe come testo separato sotto la musica.

```
Timeline = {
    \time 4/4
    \tempo \tempo 4=96
    \partial 2
    s2 | s1 | s2 \breathe s2 | s1 | s2 \caesura \break s2 | s1 | s2 \breathe s2 | s1 | s2 \fine
}
```

```plaintext
SopranoMusic = \relative g' {
g4 g | g g g g | g g g g | g g g g | g2
g4 g | g g g g | g g g g | g g g g | g2
}
```

```plaintext
AltoMusic = \relative c' {
d4 d | d d d d | d d d d | d d d d | d2
d4 d | d d d d | d d d d | d d d d | d2
}
```

```plaintext
TenorMusic = \relative a {
b4 b | b b b b | b b b b | b b b b | b2
b4 b | b b b b | b b b b | b b b b | b2
}
```

```plaintext
BassMusic = \relative g {
g4 g | g g g g | g g g g | g g g g | g2
g4 g | g g g g | g g g g | g g g g | g2
```
global = {
  \key g \major
}

\score {  % Start score
  \new PianoStaff << % Start pianostaff
  \new Staff << % Start Staff = RH
    \global
    \clef "treble"
    \new Voice = "Soprano" << % Start Voice = "Soprano"
        \Timeline
        \voiceOne
        \SopranoMusic
    >> % End Voice = "Soprano"
    \new Voice = "Alto" << % Start Voice = "Alto"
        \Timeline
        \voiceTwo
        \AltoMusic
    >> % End Voice = "Alto"
    % End Staff = RH
    \new Staff << % Start Staff = LH
    \global
    \clef "bass"
    \new Voice = "Tenor" << % Start Voice = "Tenor"
        \Timeline
        \voiceOne
        \TenorMusic
    >> % End Voice = "Tenor"
    \new Voice = "Bass" << % Start Voice = "Bass"
        \Timeline
        \voiceTwo
        \BassMusic
    >> % End Voice = "Bass"
    >> % End Staff = LH
  >> % End pianostaff
}

\markup {
  \fill-line {  
    ""
    {
        \column {
            \left-align {
                "This is line one of the first verse"
                "This is line two of the same"
                "And here's line three of the first verse"
                "And the last line of the same"
            }
        }
    }
}
This is line one of the first verse
This is line two of the same
And here's line three of the first verse
And the last line of the same

Modello per combo jazz

Ecco un modello piuttosto complesso, per un gruppo jazz. Si noti che tutti gli strumenti sono in key c major. Si tratta della tonalità reale; sarà trasposta automaticamente includendo la musica all'interno di una sezione \transpose.

\header {
  title = "Song"
  subtitle = "(tune)"
  composer = "Me"
  meter = "moderato"
  piece = "Swing"
  tagline = \markup {
    \column {
      \begin{music}
      \relative c' {
      \clef treble
      \key c \major
      \time 4/4
      \note c'2 \rest \note c'2 \note d'2 \note e'2 \note f'2
      }\end{music}
    \end{music}
  \}}
"LilyPond example file by Amelie Zapf,"
"Berlin 07/07/2003"

\%
To make the example display in the documentation
\paper { 
  \paper-width = 130
}\%
(set-global-staff-size 16)
\include "english.ly"

%%%%%%%%%%%%%%%%%% Some macros %%%%%%%%%%%%%%%%%%%

sl = { 
  \override NoteHead.style = #'slash
  \hide Stem
}

nsl = { 
  \revert NoteHead.style
  \undo \hide Stem
}

crOn = \override NoteHead.style = #'cross

crOff = \revert NoteHead.style

%%% insert chord name style stuff here.

jazzChords = { }

%%%%%%%%%%%%%%%%%% Keys’n’tang %%%%%%%%%%%%%%%%%%%

global = { \time 4/4 }

Key = { \key c \major }

% #------- Trumpet #-------

trpt = \transpose c d \relative c'' { 
  \Key
  c1 | c | c |
}

trpHarmony = \transpose c' d { 
  \jazzChords
}

trumpet = {
  \global
  \clef treble
  <<
    \trpt
  >>
}
%% ------ Alto Saxophone ------
alto = \transpose c a \relative c' {
   \Key
   c | c | c |
}
altoHarmony = \transpose c' a {
   \jazzChords
}
altoSax = {
   \global
   \clef treble
   <<
   \alto
   >>
}

%% ------ Baritone Saxophone ------
bari = \transpose c a' \relative c {
   \Key
c1
c1
   \sl
d4-"Solo" d d d
   \ns1
}
bariHarmony = \transpose c' a \chordmode {
   \jazzChords s1 s d2:maj e:m7
}
bariSax = {
   \global
   \clef treble
   <<
   \bari
   >>
}

%% ------ Trombone ------
tbone = \relative c {
   \Key
c1 | c | c |
}
tboneHarmony = \chordmode {
   \jazzChords
}
trombone = {
   \global
   \clef bass
   <<
   \tbone
   >>
}
% # Rhythm Section

% ------ Guitar ------

gtr = \relative c'' { 
  \Key 
  c1 \\
  \sl 
  b4 b b b \\
  \ns1 
  c1 
}
gtrHarmony = \chordmode { 
  \jazzChords
  s1 c2:min7+ d2:maj9 
}
guitar = { 
  \global 
  \clef treble 
  << 
  \gtr 
  >> 
}

% ------ Piano ------

rhUpper = \relative c'' { 
  \voiceOne 
  \Key 
  c1 | c | c 
}
rhLower = \relative c' { 
  \voiceTwo 
  \Key 
  e1 | e | e 
}
lhUpper = \relative c' { 
  \voiceOne 
  \Key 
  g1 | g | g 
}
lhLower = \relative c { 
  \voiceTwo 
  \Key 
  c1 | c | c 
}

PianoRH = { 
  \clef treble 
  \global 
  << 
    \new Voice = "one" \rhUpper
\new Voice = "two" \rhLower
>>
}
PianoLH = {
  \clef bass
  \global
  <<
    \new Voice = "one" \lhUpper
    \new Voice = "two" \lhLower
  >>
}
piano = {
  <<
    \new Staff = "upper" \PianoRH
    \new Staff = "lower" \PianoLH
  >>
}

% ------ Bass Guitar ------
Bass = \relative c {
  \Key
c1 | c | c
}
bass = {
  \global
  \clef bass
  <<
    \Bass
  >>
}

% ------ Drums ------
up = \drummode {
  \voiceOne
  hh4 <hh sn> hh <hh sn>
  hh4 <hh sn> hh <hh sn>
  hh4 <hh sn> hh <hh sn>
}
down = \drummode {
  \voiceTwo
  bd4 s bd s
  bd4 s bd s
  bd4 s bd s
}
drumContents = {
  \global
  <<
    \new DrumVoice \up
    \new DrumVoice \down
  >>
\score {
<<
\new StaffGroup = "horns" <<
\new Staff = "trumpet" \with { instrumentName = "Trumpet" }
\trumpet
\new Staff = "altosax" \with { instrumentName = "Alto Sax" }
\altosax
\new ChordNames = "barichords" \with { instrumentName = "Trumpet" }
\bariHarmony
\new Staff = "barisax" \with { instrumentName = "Bari Sax" }
\bariSax
\new Staff = "trombone" \with { instrumentName = "Trombone" }
\trombone
>>

\new StaffGroup = "rhythm" <<
\new ChordNames = "chords" \gtrHarmony
\new Staff = "guitar" \with { instrumentName = "Guitar" }
\guitar
\new PianoStaff = "piano" \with {
   instrumentName = "Piano"
   midiInstrument = "acoustic grand"
}
\piano
\new Staff = "bass" \with { instrumentName = "Bass" }
\bass
\new DrumStaff \with { instrumentName = "Drums" }
\drumContents
>>

>>

\layout {
\context { \Staff \RemoveEmptyStaves }
\context {
   \Score
   \override BarNumber.padding = #3
   \override RehearsalMark.padding = #2
   skipBars = ##t
}
}
\midi { }
}

Song
(tune)

Me
moderato
Modello per orchestra, coro e pianoforte

Questo modello mostra come usare i contesti annidati StaffGroup e GrandStaff per creare sottogruppi degli strumenti dello stesso tipo. Mostra anche come usare \transpose in modo che le variabili mantengano la musica per gli strumenti traspositori nell’intonazione reale.

\set-global-staff-size 17
\paper {
    indent = 3.0\cm  % add space for instrumentName
    short-indent = 1.5\cm  % add less space for shortInstrumentName
}

fluteMusic = \relative c' { \key g \major g'1 b }

% Pitches as written on a manuscript for Clarinet in A
% are transposed to concert pitch.
clarinetMusic = \transpose c' a
  \relative c'' { \key bes \major bes1 d }

trumpetMusic = \relative c { \key g \major g''1 b }

% Key signature is often omitted for horns

hornMusic = \transpose c' f
  \relative c { d'1 fis }

percussionMusic = \relative c { \key g \major g1 b }

sopranoMusic = \relative c'' { \key g \major g'1 b }

sopranoLyrics = \lyricmode { Lyr -- ics }

altoIMusic = \relative c' { \key g \major g'1 b }

altoIIMusic = \relative c' { \key g \major g'1 b }

altoILyrics = \sopranoLyrics

altoIIlyrics = \lyricmode { Ah -- ah }

tenorMusic = \relative c' { \clef "treble_8" \key g \major g1 b }

tenorLyrics = \sopranoLyrics

pianoRHMusic = \relative c { \key g \major g''1 b }

pianoLHMusic = \relative c { \clef bass \key g \major g1 b }

violinIMusic = \relative c' { \key g \major g'1 b }

violinIIMusic = \relative c' { \key g \major g'1 b }

violaMusic = \relative c { \clef alto \key g \major g'1 b }

celloMusic = \relative c { \clef bass \key g \major g1 b }

bassMusic = \relative c { \clef "bass_8" \key g \major g,1 b }

\score {
  \new StaffGroup = "StaffGroup_woodwinds" <<
    \new Staff = "Staff_flute" \with { instrumentName = "Flute" }
      fluteMusic
    \new Staff = "Staff_clarinet" \with {
        instrumentName = \markup { \concat { "Clarinet in B" \flat } }
    }
}
% Declare that written Middle C in the music
% to follow sounds a concert B flat, for
% output using sounded pitches such as MIDI.
\transposition bes

% Print music for a B-flat clarinet
\transpose bes c' \clarinetMusic
>>

\new StaffGroup = "StaffGroup_brass" <<
\new Staff = "Staff_hornI" \with { instrumentName = "Horn in F" } 
% \transposition f
\transpose f c' \hornMusic

\new Staff = "Staff_trumpet" \with { instrumentName = "Trumpet in C" } 
\trumpetMusic
>>
\new RhythmicStaff = "RhythmicStaff_percussion"
\with { instrumentName = "Percussion" }
<<
\percussionMusic
>>
\new PianoStaff \with { instrumentName = "Piano" }
<<
\new Staff { \pianoRHMusic }
\new Staff { \pianoLHMusic }
>>
\new ChoirStaff = "ChoirStaff_choir" <<
\new Staff = "Staff_soprano" \with { instrumentName = "Soprano" }
\new Voice = "soprano"
\sopranoMusic

\new Lyrics \lyricsto "soprano" { \sopranoLyrics }
\new GrandStaff = "GrandStaff_altos"
\with { \accepts Lyrics } <<
\new Staff = "Staff_altoI" \with { instrumentName = "Alto I" }
\new Voice = "altoI"
\altoIMusic

\new Lyrics \lyricsto "altoI" { \altoILyrics }
\new Staff = "Staff_altoII" \with { instrumentName = "Alto II" }
\new Voice = "altoII"
\altoIIMusic

\new Lyrics \lyricsto "altoII" { \altoIIlyrics }
>>

\new Staff = "Staff_tenor" \with { instrumentName = "Tenor" }
\new Voice = "tenor"
\tenorMusic
\new Lyrics \lyricsto "tenor" { \tenorLyrics }

\new StaffGroup = "StaffGroup_strings" <<
  \new GrandStaff = "GrandStaff_violins" <<
    \new Staff = "Staff_violinI" \with { instrumentName = "Violin I" }
      \violinIMusic

    \new Staff = "Staff_violinII" \with { instrumentName = "Violin II" }
      \violinIIMusic
  >>

  \new Staff = "Staff_viola" \with { instrumentName = "Viola" }
    \violaMusic

  \new Staff = "Staff_cello" \with { instrumentName = "Cello" }
    \celloMusic

  \new Staff = "Staff_bass" \with { instrumentName = "Double Bass" }
    \bassMusic
  >>

  \layout { }
}
Modello per pianoforte ( semplice) 

Ecco un comune doppio pentagramma per pianoforte con un po’ di note. 

upper =  \texttt{relative c' }  
\texttt{\{  
\texttt{\texttt{clef treble}} 
\texttt{\texttt{\texttt{\key c\ major}}}  
\texttt{\texttt{\texttt{\time 4/4}}  

\texttt{a4 b c d}  
\texttt{\}}  

lower =  \texttt{relative c}  
\texttt{\{  
\texttt{\texttt{\texttt{\clef bass}} } 
\texttt{\texttt{\texttt{\key c\ major}}}  
\texttt{\texttt{\texttt{\time 4/4}}  

\texttt{a2 c}  
\texttt{\}}
Invece di destinare un rigo a parte alla linea melodica e al suo testo, è possibile collocare il testo al centro di un doppio pentagramma per pianoforte.

```latex
\score {
  \new PianoStaff \with { instrumentName = "Piano" }
  \new Staff = "upper" \upper 
  \new Staff = "lower" \lower 
  \layout { }
  \midi { }
}

\new PianoStaff << 
  \new Staff = upper { \new Voice = "singer" \upper } 
  \new Lyrics \lyricsto "singer" \text 
  \new Staff = lower { \lower } 
  \layout { }
  \midi { }
}
```

**Modello per pianoforte con testo al centro**

La seguente immagine rappresenta il modello per pianoforte con testo al centro.
Modello per pianoforte con melodia e testo

Ecco un tipico formato per canzoni: un rigo con linea melodica e testo, e sotto l'accompagnamento per pianoforte.

\begin{verbatim}
melody = \relative c' { 
  \clef treble 
  \key c \major 
  \time 4/4 
  a b c d 
}

text = \lyricmode { 
  Aaa Bee Cee Dee 
}

upper = \relative c' { 
  \clef treble 
  \key c \major 
  \time 4/4 
  a4 b c d 
}

lower = \relative c { 
  \clef bass 
  \key c \major 
  \time 4/4 
  a2 c 
}

\score { 
  << 
    \new Voice = "mel" { \autoBeamOff \melody } 
    \new Lyrics \lyricsto mel \text 
    \new PianoStaff << 
      \new Staff = "upper" \upper 
      \new Staff = "lower" \lower 
    >> 
  >> 
  \layout { 
    \context { \Staff \RemoveEmptyStaves } 
  } 
\end{verbatim}
Modello per coro SATB - quattro righi

Modello per coro SATB (quattro righi)

global = {
  \key c \major
  \time 4/4
  \dynamicUp
}
sopranonotes = \relative c'' { c2 \p < d c d \f }
sopranowords = \lyricmode { do do do do }
altonotes = \relative c'' { c2\p d c d }
alтовords = \lyricmode { re re re re }
tenornotes = { \clef "G_8" c2\mp d c d }
tenorwords = \lyricmode { mi mi mi mi }
bassnotes = { \clef bass c2\mf d c d }
basswords = \lyricmode { mi mi mi mi }

\score {
  \new ChoirStaff <<
  \new Staff <<
    \new Voice = "soprano" <<
      \global
      \sopranonotes
    >>
    \new Lyrics \lyricsto "soprano" \sopranowords
  >>
  \new Staff <<
    \new Voice = "alto" <<
Score for diatonic accordion

A template to write a score for a diatonic accordion.

- There is a horizontal staff indicating if the accordion must be pushed (thick line) or pulled (thin line)
- There is a small rhythmic staff with lyrics that describes the bass buttons to press. The bar lines are made from gridlines
- The tabulator staff for diatonic accordions shows the geographic position of the buttons and not (as for every other instrument) the pitch of the notes; the keys on the melody-side of the accordion are placed in three columns and about 12 rows
In the tabulator staff notation the outermost column is described with notes between lines, the innermost column is described with notes between lines and a cross as accidental, and the middle column is described with notes on a line, whereby the row in the middle is represented on the middle line in the staff.

Some words to transpose piano notes to the diatonic accordion:

1. Every diatonic accordion is built for some keys only (for example, for the keys of C major and F major), so it is important to transpose a piano melody to match one of these keys. Transpose the source code, not only the output because this code is required later on to translate it once more to the tabulator staff. This can be done with the command displayLilyMusic.

2. You have to alternate the push- and pull-direction of the accordion regularly. If the player has a too long part to pull the accordion gets broken. On the other hand, some harmonies are only available in one direction. Considering this, decide which parts of the melody are the push-parts and which the pull-parts.

3. For each pull- or push-part translate the piano notes to the according tablature representation.

```latex
verse = \lyricmode \{ Wie gross bist du! Wie gross bist du! \}

harmonies = \new ChordNames \chordmode \{
  \germanChords
  \set chordChanges = ##t
  bes8 bes8 bes8
  es2 f
  bes1
}\n
NoStem = \{ \hide Stem \}
NoNoteHead = \hide NoteHead
ZeroBeam = \override Beam.positions = #'(0 . 0)

staffTabLine = \new Staff \with \{
  \remove "Time_signature_engraver"
  \remove "Clef_engraver"
\} \{
  \override Staff.StaffSymbol.line-positions = #'(0)
  % Shows one horizontal line. The vertical line
  % (simulating a bar-line) is simulated with a gridline
  \set Staff.midiInstrument = #"choir aahs"
  \key c \major
  \relative c''
  {
    % disable the following line to see the noteheads while writing the song
    \NoNoteHead
    \override NoteHead.no-ledgers = ##t
    % The beam between 8th-notes is used to draw the push-line
    % How to fast write the push-lines:
    % 1. write repeatedly 'c c c c c c c |' for the whole length of the song
    % 2. uncomment the line \NoNoteHead
    % 3. compile
    % 4. Mark the positions on which push/pull changes.
    % In the score-picture click on the position
```
the push- or pull-part starts
(on the noteHead, the cursor will change to a hand-icon).
The cursor in the source code will jump just at this position.
a) If a push-part starts there, replace the 'c' by an 'e[
 b) If a pull-part starts there, replace the 'c' by an 's'
 5. Switch into 'overwrite-mode' by pressing the 'ins' key.
 6. For the pull-parts overwrite the 'c' with 's'
 7. For every push-part replace the last 'c' with 'e]'
 8. Switch into 'insert-mode' again
 9. At last it should look like e.g.
 (s s e[ c c c c c c | c c c c c c e] s s)
10. re-enable the line \NoNoteHead\autoBeamOff\ZeroBeam
  s8 s s e[ c c c c c e] | s s s s s

% Accordion melody in tabulator score
% 1. Place a copy of the piano melody below
% 2. Separate piano melody into pull- and push-parts
% according to the staffTabLine you've already made
% 3. For each line: Double the line. Remark the 1st one
% (Keeps unchanged as reference) and then change the second
% line using the transformation paper
% or the macros 'conv2diaton push.bsh' and 'conv2diaton pull.bsh'
% Tips:
% - In jEdit Search & Replace mark the Option 'Keep Dialog'

AccordionTabTwoCBesDur = {
% pull 1
<\%<f' bes'>8 <f' a'>8 <d' bes'>8 |
<g'' a'">8 <g'' b'">8 <e'' a'">8 |
% push 2
<\%<g' c'">4 <f' d'"> <g' ees'"> <f' a'"> |
<g'' a'">4 <d'' eisis'"> <g'' bisis'"> <d'' f'"> |
% pull 3
% <f' bes'>2 r8 }
<g'' a'">2 r8 }

AccordionTab= { \dynamicUp
% 1. Place a copy of the piano melody above
% 2. Separate piano melody into pull- and push-parts
% according to the staffTabLine you've already made
% 3. For each line: Double the line. Remark the 1st one
% (Keeps unchanged as reference) and then
% change the second line using the transformation paper
% Tips:
% - In jEdit Search & Replace mark the Option 'Keep Dialog'
% -
\AccordianTabTwoCBesDur}
\layout {
  \context {
    \Score
    \context {
      \Score
      % The vertical line (simulating a bar-line) in
      % the staffBassRhythm is a gridline
      \consists "Grid_line_span_engraver"
    }
    \context {
      \Staff
      \consists "Grid_point_engraver"
      \gridInterval = #(ly:make-moment 4/4) % 4/4 - tact. How many beats per bar
      % The following line has to be adjusted O-F-T-E-N.
      \override GridPoint.Y-extent = #'(-2. -21)
    }
    \context {
      \ChoirStaff
      \remove "System_start_delimiter_engraver"
    }
  }
}

staffVoice = new Staff = astaffvoice {
  \time 4/4
  \set Staff.instrumentName = "Voice"
  \set Staff.midiInstrument = "voice oohs"
  \key bes \major
  \partial 8*3
  \clef treble
  { \context Voice = "melodyVoi"
    { <f' bes'>8 <f' a'>8 <d' bes'>8 | 
      <g' c''>4 <f' d''> <g' es''> <f' a'> | 
      <f' bes'>2 r8
    }
  }
}

staffAccordionMel = new Staff \with { \remove "Clef_engraver" } {
  \accidentalStyle forget %Set the accidentals (Vorzeichen) for each note,
  %do not remember them for the rest of the measure.
  \time 4/4
  \set Staff.instrumentName="Accordion"
  \set Staff.midiInstrument="voice oohs"
  \key c \major
  \clef treble
  { \AccordionTab \bar "|." }
}
\new Staff = staffbass \with { \remove "Clef_engraver" } {
% This is not a RhythmicStaff because it must be possible to append lyrics.

  \override Score/GridLine.extra-offset = #'( 13.0 , 0.0 ) \% x.y
  \override Staff/StaffSymbol.line-positions = #'( 0 )
  \% Shows one horizontal line. The vertical line
  \% (simulating a bar-line) is simulated by a grid
  \% Search for 'grid' in this page to find all related functions
  \time 4/4
  {
    \context Voice = "VoiceBassRhythm"
    \stemDown \AltOn #0.6
    \relative c''
    {
      \BassRhythm
    }
    \AltOff
    \bar "|.
  }
}

\score {
  \new ChoirStaff <<
    \harmonies
    \staffVoice
    \context Lyrics = "lmelodyVoi"
    \with { \alignBelowContext = astaffvoice }
    \lyricsto melodyVoi \verse
    \staffAccordionMel
    \staffTabLine
    \staffBassRhythm
    \context Lyrics = "lBassRhythmAboveI"
    \with { \alignAboveContext = staffbass }
    \lyricsto VoiceBassRhythm \LyricBassRhythmI
  >>
}
Modello di rigo singolo con note, testo e accordi

Ecco il modello di un comune spartito semplificato (lead sheet): include linea melodica, testo vocale e sigle degli accordi.

```plaintext
melody = \relative c' { 
    \clef treble
    \key c \major
    \time 4/4
    a4 b c d
}

text = \lyricmode { 
    Aaa Bee Cee Dee
}

harmonies = \chordmode { 
    a2 c
}

\score { 
    \new ChordNames { 
    \set chordChanges = ##t 
    \harmonies 
    } 
    \new Voice = "one" { \autoBeamOff \melody } 
    \new Lyrics \lyricsto "one" \text 
    >> 
    \layout { } 
    \midi { } 
    }
```
Single staff template with notes, lyrics, chords and frets

Here is a simple lead sheet template with melody, lyrics, chords and fret diagrams.

verseI = \lyricmode {
  \set stanza = "1."
  This is the first verse
}

verseII = \lyricmode {
  \set stanza = "2."
  This is the second verse.
}

theChords = \chordmode {
  \% insert chords for chordnames and fretboards here
  c2 g4 c
}

staffMelody = \relative c' {
  \key c \major
  \clef treble
  \% Type notes for melody here
  c4 d8 e f4 g
  \bar "|."
}

\score {
  <<
    \context ChordNames { \theChords }
    \context FretBoards { \theChords }
    \new Staff {
      \context Voice = "voiceMelody" { \staffMelody }
    }
    \new Lyrics = "lyricsI" {
      \lyricsto "voiceMelody" \verseI
    }
    \new Lyrics = "lyricsII" {
      \lyricsto "voiceMelody" \verseII
    }
  >>
  \layout { }
  \midi { }
}
Modello di rigo singolo con note e accordi

Vuoi preparare uno spartito semplificato (lead sheet) con melodia e accordi? La tua ricerca è finita!

melody = \relative c' {\clef treble \key c \major \time 4/4 
f4 e8[ c] d4 g 
a2 - a }

harmonies = \chordmode {c4:m f:min7 g:maj c:aug d2:dim b4:5 e:sus}

\score {<< \new ChordNames { \set chordChanges = ##t \harmonies } \new Staff \melody >> \layout{} \midi{} }

Modello di rigo singolo con note e testo

Questo piccolo modello presenta una semplice linea melodica con un testo. Copialo e incollalo, aggiungi le note e le parole. Questo esempio disabilita la disposizione automatica delle travature, come è consuetudine per le parti vocali. Per usare la disposizione automatica delle travature, cambia o commenta la relativa linea di codice.

melody = \relative c' {
  \clef treble
Modello di rigo singolo con solo note

Questo modello molto semplice mette a disposizione un rigo con delle note ed è quindi adatto per uno strumento non accompagnato o per un frammento melodico. Copialo e incollalo in un file, aggiungi le note e hai finito!

```plaintext
\key c \major
\time 4/4
\relative c' {
    \clef treble
    \key c \major
    \time 4/4
    \new Staff
    \score {
        \layout { }
        \midi { }
    }
    a4 b c d
}
```

Aaa Bee Cee Dee
Modello per quartetto d’archi (semplice)
Questo modello presenta un semplice quartetto d’archi. Impiega anche una sezione `global` per definire il tempo e l'armatura di chiave.

```plaintext
global  = {  
    \time 4/4  
    \key c \major
}

violinOne = \new Voice \relative c' {  
    c2 d  
    e1  
    \bar "|." }

violinTwo = \new Voice \relative c' {  
    g2 f  
    e1  
    \bar "|." }

viola = \new Voice \relative c' {  
    \clef alto  
    e2 d  
    c1  
    \bar "|." }

cello = \new Voice \relative c' {  
    \clef bass  
    c2 b  
    a1  
    \bar "|." }

\score {  
    \new StaffGroup <<  
    \new Staff \with { instrumentName = "Violin 1" } << \global violinOne >>  
    \new Staff \with { instrumentName = "Violin 2" } << \global violinTwo >>  
    \new Staff \with { instrumentName = "Viola" } << \global viola >>  
    \new Staff \with { instrumentName = "Cello" } << \global cello >>  
    >>  
    \layout { }  
    \midi { }  
}```
Modello per quartetto d’archi con parti separate

Il frammento di codice del “Modello per quartetto d’archi” crea un bel quartetto, ma cosa fare se si ha bisogno di creare le singole parti? Questo nuovo modello mostra come usare la funzionalità \tag per dividere facilmente un pezzo in parti staccate.


Non dimenticare di togliere i commenti quando usi i file separati!

---

\global = {
    \time 4/4
    \key c \major
}

Violinone = \new Voice {
    \relative c'' {
        c2 d e1
        \bar "| ."
    }
}

Violintwo = \new Voice {
    \relative c' {
        g2 f e1
        \bar "| ."
    }
}

Viola = \new Voice {
    \relative c' {
        \clef alto
    }
}

---
e2 d c1
\bar "|."
{% Uncomment this block when using separate files

% vn1.ly
% (This is the Violin 1 part file)
\include "piece.ly"
\score {
  \keepWithTag #'vn1 \music
  \layout { }
}

% vn2.ly
% (This is the Violin 2 part file)
\include "piece.ly"
\score {
  \keepWithTag #'vn2 \music
  \layout { }
}

% vla.ly
% (This is the Viola part file)
\include "piece.ly"
\score {
  \keepWithTag #'vla \music
  \layout { }
}

% vlc.ly
% (This is the Cello part file)
\include "piece.ly"
\score {
  \keepWithTag #'vlc \music
  \layout { }
}
%
%}
Modello per complesso vocale

Ecco una tipica partitura corale a quattro parti, SATB. Se il complesso è più ampio, è spesso comodo scrivere gli elementi comuni in un’unica sezione, che verrà poi inclusa in tutte le parti. Ad esempio, l’indicazione di tempo e l’armatura di chiave sono quasi sempre le stesse per tutte le parti. Come nel modello dell’“Inno”, le quattro voci sono ripartite in due soli righi.

\paper {
  top-system-spacing.basic-distance = #10
  score-system-spacing.basic-distance = #20
  system-system-spacing.basic-distance = #20
  last-bottom-spacing.basic-distance = #10
}

\key c \major 
\time 4/4

sopMusic = \relative {
  c''4 c c8[\( b)] c4
}

sopWords = \lyricmode {
  hi hi hi hi
}

altoMusic = \relative {
  e'4 f d e
}

altoWords = \lyricmode {
  ha ha ha ha
}

tenorMusic = \relative {
  g4 a f g
}

tenorWords = \lyricmode {
  hu hu hu hu
}

bassMusic = \relative {
  c4 c g c
}
bassWords = \lyricmode {
    ho ho ho ho
}

\score {
    \new ChoirStaff <<
    \new Lyrics = "sopranos" \with {
        % this is needed for lyrics above a staff
        \override VerticalAxisGroup.staff-affinity = #DOWN
    }
    \new Staff = "women" <<
    \new Voice = "sopranos" {
        \voiceOne << \global \sopMusic >>
    }
    \new Voice = "altos" {
        \voiceTwo << \global \altoMusic >>
    }
    >>
    \new Lyrics = "altos"
    \new Lyrics = "tenors" \with {
        % this is needed for lyrics above a staff
        \override VerticalAxisGroup.staff-affinity = #DOWN
    }
    \new Staff = "men" <<
    \clef bass
    \new Voice = "tenors" {
        \voiceOne << \global \tenorMusic >>
    }
    \new Voice = "basses" {
        \voiceTwo << \global \bassMusic >>
    }
    >>
    \new Lyrics = "basses"
    \context Lyrics = "sopranos" \lyricsto "sopranos" \sopWords
    \context Lyrics = "altos" \lyricsto "altos" \altoWords
    \context Lyrics = "tenors" \lyricsto "tenors" \tenorWords
    \context Lyrics = "basses" \lyricsto "basses" \bassWords
    >>
}
Modello per gruppo vocale con riduzione per pianoforte automatica

Questo modello aggiunge una riduzione automatica per pianoforte alla tipica partitura vocale SATB illustrata in “Modello per complesso vocale”. Si dimostra così uno dei punti di forza di LilyPond – è possibile usare una definizione musicale più di una volta. Qualsiasi modifica venga fatta alle note delle voci (ad esempio, tenorMusic) verrà applicata anche alla riduzione per pianoforte.

```latex
\global = {
  \key c \major
  \time 4/4
}

sopMusic = \relative {
  c''4 c c8[(b)] c4
}
sopWords = \lyricmode {
  hi hi hi hi
}

altoMusic = \relative {
  e'4 f d e
}
altoWords =\lyricmode {
  ha ha ha ha
}

tenorMusic = \relative {
  g4 a f g
}
tenorWords = \lyricmode {
  hu hu hu hu
}```
bassMusic = \relative { c4 c g c }
bassWords = \lyricmode { ho ho ho ho }

\score { <<
  \new ChoirStaff <<
  \new Lyrics = "sopranos" \with {
    \override VerticalAxisGroup.staff-affinity = #DOWN
  }
  \new Staff = "women" <<
    \new Voice = "sopranos" { \voiceOne << \global \sopMusic } 
    \new Voice = "altos" { \voiceTwo << \global \altoMusic } 
  >>
  \new Lyrics = "altos"
  \new Lyrics = "tenors" \with {
    \override VerticalAxisGroup.staff-affinity = #DOWN
  }
  \new Staff = "men" <<
    \clef bass
    \new Voice = "tenors" { \voiceOne << \global \tenorMusic } 
    \new Voice = "basses" { \voiceTwo << \global \bassMusic } 
  >>
  \context Lyrics = "sopranos" \lyricsto "sopranos" \sopWords
  \context Lyrics = "altos" \lyricsto "altos" \altoWords
  \context Lyrics = "tenors" \lyricsto "tenors" \tenorWords
  \context Lyrics = "basses" \lyricsto "basses" \bassWords 
>>

  \new PianoStaff <<
  \new Staff <<
    \set Staff.printPartCombineTexts = ##f
    \partCombine
    << \global \sopMusic >>
    << \global \altoMusic >> 
  >>

  \new Staff <<
    \clef bass
    \set Staff.printPartCombineTexts = ##f
    \partCombine
    << \global \tenorMusic >>
    << \global \bassMusic >> 
  >>
>>
Modello per gruppo vocale con testo allineato sotto e sopra i righi

 Questo modello è fondamentalmente analogo al semplice modello “Complesso vocale”, con l’unica differenza che qui tutti i versi del testo sono posizionati usando alignAboveContext e alignBelowContext.

```
global = {
    \key c \major
    \time 4/4
}
```

```
sopMusic = \relative c'  { 
    c4 c c8[( b)] c4
}
sopWords = \lyricmode { 
    hi hi hi hi
}
```

```
altoMusic = \relative c'  { 
    e4 f d e
}
altoWords = \lyricmode { 
    ha ha ha ha
}
```

```
tenorMusic = \relative c'  { 
    g4 a f g
}
tenorWords = \lyricmode { 
    hu hu hu hu
}
\score {
  \new ChoirStaff <<
  \new Staff = "women" <<
    \new Voice = "sopranos" { \voiceOne << \global \sopMusic >> }
    \new Voice = "altos" { \voiceTwo << \global \altoMusic >> }
  >>
  \new Lyrics \with { \alignAboveContext = "women" }
    \lyricsto "sopranos" \sopWords
  \new Lyrics \with { \alignBelowContext = "women" }
    \lyricsto "altos" \altoWords
  \%
  \%
  \%
  \%
  \%
  \%
  % we could remove the line about this with the line below, since
  % we want the alto lyrics to be below the alto Voice anyway.
  \%
  \%
  \%
  \%
  \%
  \%
  \%
  \%
  \%
  % \new Lyrics \lyricsto "altos" \altoWords

  \new Staff = "men" <<
    \clef bass
    \new Voice = "tenors" { \voiceOne << \global \tenorMusic >> }
    \new Voice = "basses" { \voiceTwo << \global \bassMusic >> }
  >>
  \new Lyrics \with { \alignAboveContext = "men" }
    \lyricsto "tenors" \tenorWords
  \new Lyrics \with { \alignBelowContext = "men" }
    \lyricsto "basses" \bassWords
  \%
  \%
  \%
  \%
  \%
  \%
  \%
  \%
  \%
  % again, we could replace the line above this with the line below.
  % \new Lyrics \lyricsto "basses" \bassWords
  >>
}
Modello per gruppo vocale con strofa e ritornello

Questo modello crea una partitura che inizia con una sezione solistica e prosegue in un ritornello a due voci. Illustra anche l’uso delle pause spaziatrici all’interno della variabile `global` per definire i cambi di tempo (e altri elementi comuni a tutte le parti) nel corso di tutta la partitura.

```latex
global = { \\
    \key g \major \time 3/4 \\
    s2.*2 \break \\

    \time 2/4 \\
    s2*2 \bar "|." \\
}
```

SoloNotes = `relative g' { \\
    \clef "treble" \\
    % verse \\
    g^4 g g | \\
    b^4 b b | \\
    % refrain \\
    R2*2 |
}

SoloLyrics = `lyricmode { \\
    One two three | \\
    four five six | 
}

SopranoNotes = `relative c'' { \\
    \clef "treble" \\
    % verse \\
    R2.*2 | \\
    % refrain \\
    c^4 c | \\
    g^4 g |
}

SopranoLyrics = `lyricmode { \\
    la la | \\
    la la |
}

BassNotes = `relative c { \\

\clef "bass"

% verse
R2.*2 |

% refrain
c4 e |
d4 d |

BassLyrics = \lyricmode {
  dum dum |
  dum dum |
}

\score {
  <<
    \new Voice = "SoloVoice" << \global \SoloNotes >>
    \new Lyrics \lyricsto "SoloVoice" \SoloLyrics

    \new ChoirStaff <<
      \new Voice = "SopranoVoice" << \global \SopranoNotes >>
      \new Lyrics \lyricsto "SopranoVoice" \SopranoLyrics

      \new Voice = "BassVoice" << \global \BassNotes >>
      \new Lyrics \lyricsto "BassVoice" \BassLyrics
    >>
  >>

  \layout {
    ragged-right = ##t
    \context { \Staff
      % these lines prevent empty staves from being printed
      \RemoveEmptyStaves
      \override VerticalAxisGroup.remove-first = ##t
    }
  }
}

One two three four five six

la la la la la

dum dum dum dum