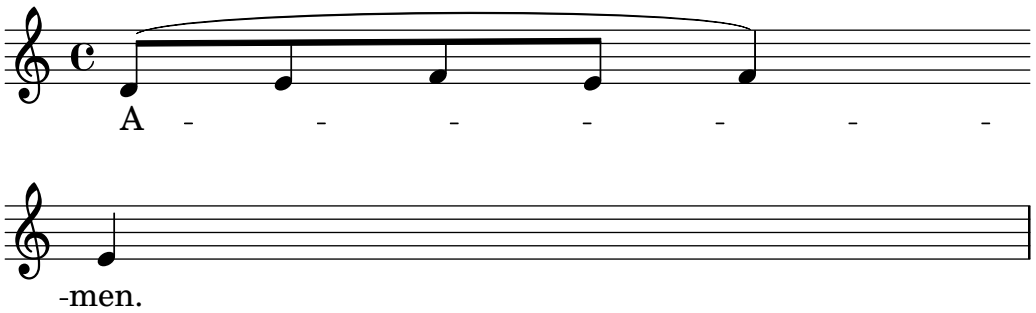
Funcionalidades nuevas de la versión 2.22 desde la 2.20

- Ahora se contemplan los marcadores de PDF y permiten que las entradas de `\tocItem` aparezcan dentro del panel de 'tabla de contenidos' de los visores de PDF que lo permiten.
- `\table-of-contents` ahora acepta una estructura jerárquica; las entradas de `\tocItem` pueden opcionalmente aceptar un símbolo (como `\label`) o una lista de símbolos separados por puntos, indicando su posición dentro de la estructura de la partitura. Un efecto secundario de esta funcionalidad es que `\tocItem` ya no puede tomar una simple cadena como argumento; se debe utilizar una instrucción `\markup`.
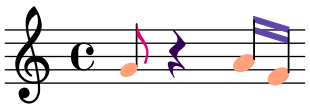
```
\tocItem \markup "Entrada principal"
\tocItem symbol "Entrada principal con nombre"
\tocItem symbol.list "Entrada secundaria (de segundo nivel)"
```

- Ahora se puede repetir un guión de letra al comienzo de un sistema que comienza con una sílaba nueva:



- Ahora se pueden usar colores de estilo CSS directamente como cadenas de texto; ya sea como nombres de color predefinidos (como ocurre con la función `css-color` que ya está disponible), o con códigos de color hexadecimales con el prefijo `#`. Todos los sellos o stencils que contemplan una propiedad `color` ahora aceptan tanto una lista como una cadena; en este último caso, la cadena se utiliza directamente en la salida de SVG. Esto permite usar transparencia alpha (introducido como "#RRGGBBAA" o "#RGBA") dentro del SVG.
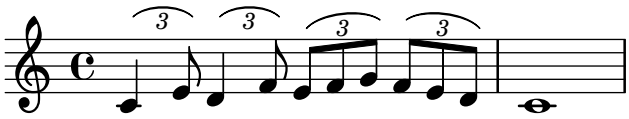
```
\override NoteHead.color = "lightsalmon"
\override Flag.color = "#E30074"
\override Beam.color = "#5e45ad"
\override Rest.color = "#3058"
g'8 \huge r4 a'16 f'
```



- Es posible obtener salida de SVG a través de la opción de línea de órdenes `--svg` (o su forma canónica `--format=svg`, que también se puede abreviar como `-fsvg`). A causa de su especificidad respetco al backend, esa opción no es compatible aún con otros formatos de salida; para obtener un archivo en los dos formatos SVG y PDF, PNG o EPS, es necesaria otra ejecución del programa LilyPond.
- COmo se anunció en la versión 2.17.3 hace casi ocho años, la opción `'relative-includes` está activada de forma predeterminada; los archivos de inclusión que contienen a su vez alguna instrucción `\include`, deben tener en cuenta su propia ruta en lugar del directorio del archivo principal. Sin embargo, este comportamiento se puede desactivar mediante el establecimiento de `'relative-includes` al valor falso `#f`, ya sea como una opción de la línea de órdenes o utilizando `ly:set-option` dentro de los archivos de entrada.
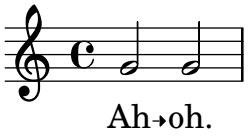
- Los grupos de valoración especial se pueden imprimir ahora con ligaduras en lugar de corchetes:

```
\relative {
  \tuplet 3/2 4 {
    \override TupletBracket.tuplet-slur = ##t
    c'4 e8 d4 f8
    \override TupletBracket.bracket-visibility = ##t
    e f g f e d
  } c1
}
```
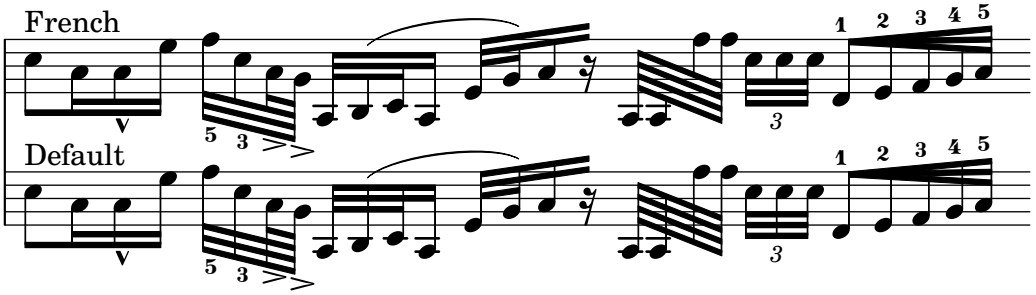


- La palabra clave `\compressFullBarRests` ha cambiado su nombre por `\compressEmptyMeasures`, para evitar cualquier confusión con `\compressMMRests`. De manera similar, `\expandFullBarRests` es ahora `\expandEmptyMeasures`.
- Un cambio progresivo en una vocal (o en una consonante mantenida) puede indicarse añadiendo una transición de vocal entre sílabas de la letra con la instrucción `\vowelTransition`.

```
{ g'2 g' } \addlyrics { Ah
\vowelTransition oh. }
```
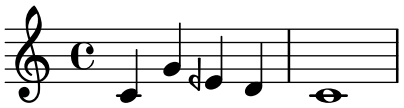


- LilyPond para Windows (MinGW) maneja correctamente nombres de archivo en Unicode sobre Windows 10 1903 o superior.
- El barrado francés (`\override Stem.french-beaming = ##t`) se comporta ahora *exactamente* como el barrado estándar (o predeterminado) a todos los efectos (posicionamiento de cualquier articulación respecto de la barra, digitaciones, etc.). La única diferencia que aún permanece es que las plicas interiores no atraviesan las barras.
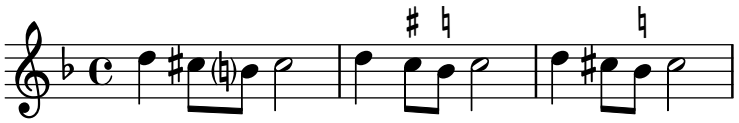


- Los nombres de nota en cuartos de tono ahora están disponibles en todos los idiomas de entrada.

```
\language "català"
\relative { do'4 sol' miqb re do1 }
```
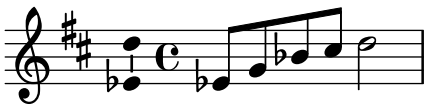


- Todos los idiomas de entrada (instrucción `\language`) se pueden introducir utilizando su nombre exacto codificado en UTF-8 (esto es, incluyendo caracteres especiales). Ahora se pueden usar los nombres que faltaban, `català` y `português`, además de los originales `catalan` y `portugues`.
- La configuración de `suggestAccidentals` a `'cautionary` convierte en alteraciones sugeridas `AccidentalSuggestion`s (por encima) solamente las alteraciones de precaución o de cortesía. Esto se puede usar para diferenciar entre alteraciones editoriales y de facsímil.

```
\relative {
  \key d \minor
  d''4 cis8 b? cis2
  \set suggestAccidentals = ##t
  d4 cis8 b? cis2
  \set suggestAccidentals = #'cautionary
  d4 cis8 b? cis2
}
```



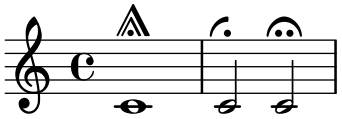
- Se ha añadido la instrucción `\ambitusAfter`. Se usa para desplazar la indicación de ámbito o tesitura a una posición distinta.

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



- Se han añadido las variantes de calderón muy corto y de Henze a los calderones largo y corto.

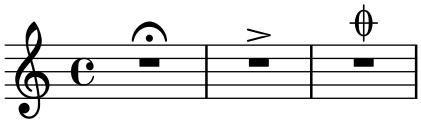
```
{
  c'1\veryshortfermata
  c'2\henzeshortfermata c'\henzelongfermata
}
```



- Ahora se pueden añadir calderones y otras articulaciones directamente a los silencios multicompás. Como consecuencia, se convierte en obsoleta la instrucción `\fermataMarkup`.

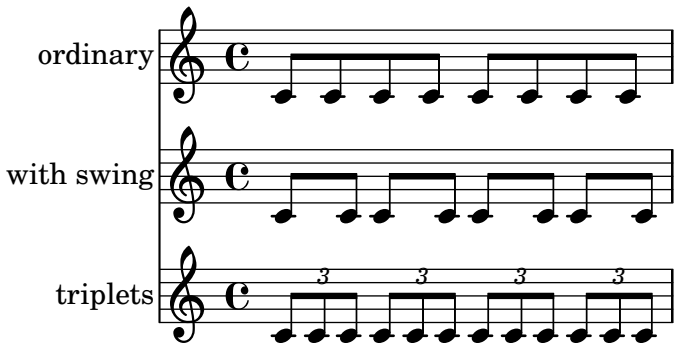
```
{
  R1\fermata
  R->
  R\coda
```

```
}
```



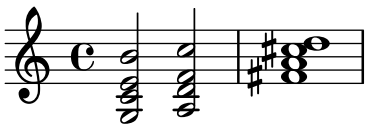
- Ahora se pueden aplicar patrones rítmicos irregulares y de Swing a expresiones musicales compuestas por duraciones regulares, algo que puede utilizarse para generar interpretaciones rítmicas desiguales en el MIDI.

```
\include "swing.ly"
<<
  \new Staff \with {
    instrumentName = "ordinary"
  } \repeat unfold 8 c'8
  \new Staff \with {
    instrumentName = "with swing"
  } \repeat unfold 4 \tripletFeel 8 { c'8 c' }
  \new Staff \with {
    instrumentName = "triplets"
  } \tuplet 3/2 4 \repeat unfold 12 c'8
>>
```



- Ahora se pueden invertir automáticamente los acordes con notas suprimidas.

```
\chordmode {
  \dropNote 2 {
    c2:maj7 d:m7
  }
  \invertChords 1 d1:maj7
}
```

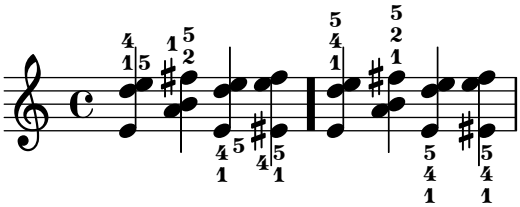


- Se han arreglado algunos diagramas de posiciones de acordes para los trastes del ukelele, y se han añadido algunos nuevos.
- Las funciones antiguas de nomenclatura de acordes `banter-chord-names` y `jazz-chord-names` se han eliminado del código principal, así como las propiedades `chordNamesExceptionsFull` y `chordNamesExceptionsPartial`. Como parte de dicha remodelación, los acordes 'de potencia' están incluidos ahora dentro de las excepciones predeterminadas; se imprimen correctamente (con un superíndice como el resto de los

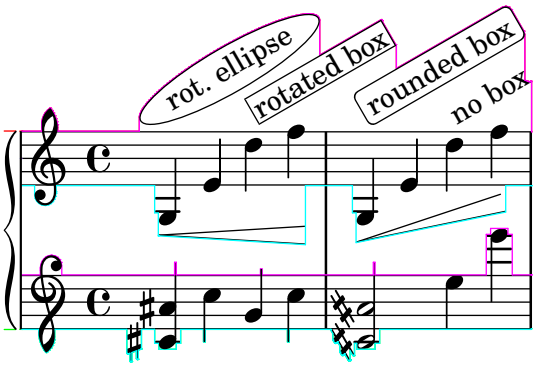
acordes) y sin que sea necesario ningún otro trucaje adicional, suprimiendo así la necesidad de la instrucción predefinida `\\powerChords`. Las personas interesadas en las funciones antiguas que se mantenían por compatibilidad tienen una copia bastante funcional en el fragmento de código `chord-names-alternative.ly`.

- La instrucción `\\partcombine`, así como todas las instrucciones, subrutinas y nombres de propiedad que llevaban el prefijo `partCombine` se escriben ahora con C mayúscula, tal y como `\\partCombine`, `\\partCombineApart`, etc. La instrucción `\\autochange` se escribe también con la C mayúscula como `\\autoChange`.
- EL grabador `New_fingering_engraver` tiene en cuenta ahora la propiedad `X-align-on-main-noteheads` del `self-alignment-interface`. Si se establece al valor verdadero (`##t`), todas las digitaciones orientadas hacia arriba o hacia abajo (`up` o `down`) se disponen en una columna normal y se alinean sobre las cabezas de nota en el lado correcto de la cabeza de la figura:

```
{
  \set fingeringOrientations = #'(up)
  <e'-1 d''-4 e''-5>4 <a'-1 b'-2 fis''-5>
  \set fingeringOrientations = #'(down)
  <e'-1 d''-4 e''-5> <eis'-1 e''-4 fis''-5>
  \bar "."
  \override Fingering.X-align-on-main-noteheads = ##t
  \set fingeringOrientations = #'(up)
  <e'-1 d''-4 e''-5> <a'-1 b'-2 fis''-5>
  \set fingeringOrientations = #'(down)
  <e'-1 d''-4 e''-5> <eis'-1 e''-4 fis''-5>
}
```



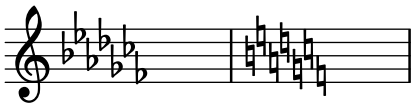
- Ahora las líneas de horizonte o Skylines tienen en cuenta la propiedad de rotación (`rotation`) de los objetos de presentación. Por ejemplo, al rotar un regulador de crescendo mediante la aplicación de `\override Hairpin.rotation = #'(15 0 0)`, se producirá una influencia real sobre las líneas de horizonte, lo que será de ayuda para mejorar el espaciado.



- Se ha añadido un pequeño relleno entre los glifos de becuadro que apenas tocan por la esquina en las cancelaciones de armadura.

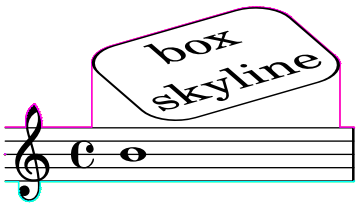
```
{
```

```
  \omit Staff.TimeSignature
  \key ces \major s1 \key c \major s
}
```
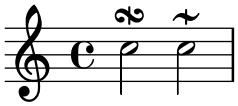


- Las líneas de horizonte o Skylines de los rectángulos reflejan ahora el trazo real del rectángulo, incluyendo las esquinas redondeadas y cualquier rotación que presenten.

```
#(ly:set-option 'debug-skylines #t)
{
  b'1^\markup \scale #'(2 . 1)
              \rotate #30
              \override #'(corner-radius . 3)
              \rounded-box \center-column { box skyline }
}
```

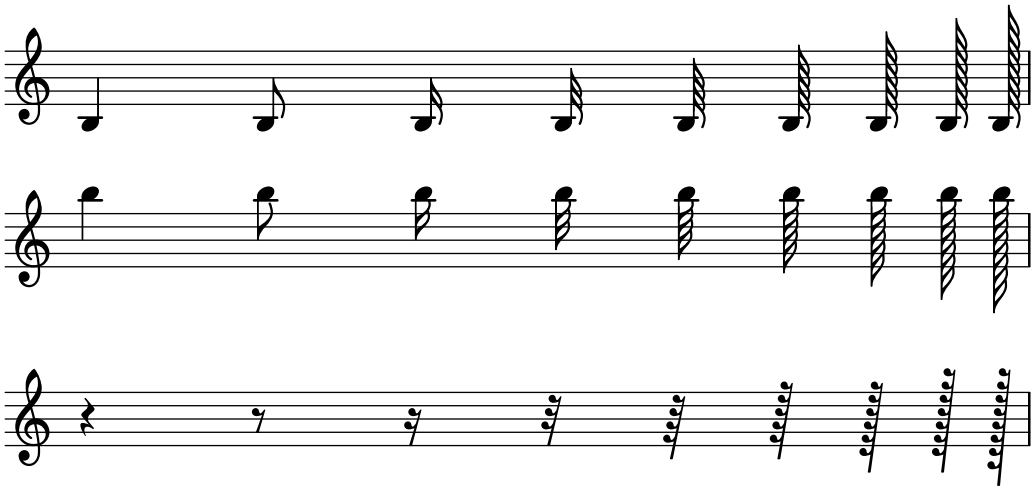


- Se han añadido dos símbolos de adorno nuevos.

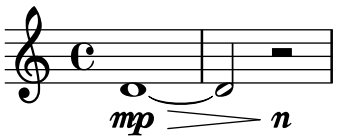
```
{
  c''2\slashturn c''\haydnturn
}
```



- Se han añadido a las fuentes tipográficas los glifos de los corchetes y silencios de las figuras de semigarrapatea (1/256 de redonda, con 6 corchetes o barras), así como de las de 1/512 y 1/1024, con 7 y 8 corchetes o barras respectivamente, que no tienen nombre en español.



- Se ha añadido una instrucción de matiz dinámico `\n` con el significado de *niente*.



- Ahora se pueden imprimir los diagramas invertidos de posiciones de acordes para instrumentos de trastes, para personas zurdas, configurando la propiedad `handedness` al valor `LEFT` (como una subpropiedad de `fret-diagram-details`).

```
\markup
  \center-column {
    "C"
    "(left-handed)"
     \override #`(fret-diagram-details . ((handedness . ,LEFT)))
     \fret-diagram "6-x;5-3-3;4-2-2;3-o;2-1;1-o;"
  }
```

C

(left-handed)



- EL procedimiento `output-classic-framework` y la opción `-dclip-systems` están ahora disponibles con el backend `SVG`.
- Se ha añadido una opción, `-dcrop`, que formatea la salida de `SVG` y de `PDF` sin márgenes ni saltos de página.