

28. Sobreescritura de propiedades: Marte, de Holst (II)

28.1. Modelo

Para completar el ejemplo orquestal, hoy aprenderemos a mover objetos para ajustar su posición; en el caso que nos ocupa, esto ahorra espacio y permite un tamaño de los pentagramas algo mayor, sin que se produzcan colisiones entre los objetos de los distintos pentagramas, y todo ello de forma que la música quepa en una sola página. En el título usaremos el efecto *smallcaps* de mayúsculas pequeñas para “Mars”.

En el fragmento aparecen los pentagramas de los fagotes y el contrafagot, con el matiz *mezzopiano* y la indicación “III” del tercer fagot desplazadas a la izquierda y hacia arriba. Se ha enmascarado en blanco el pentagrama detrás de estas indicaciones para evitar la superposición. Las pautas de percusión se han acercado entre sí para ahorrar espacio. La indicación *piano* de los violines está también desplazada para hacer sitio al texto *col legno* del siguiente pentagrama. Para finalizar, hemos reducido el grosor de las líneas de pauta para suavizar el aspecto demasiado negro de una partitura orquestal a tamaño reducido.

I. MARS, the Bringer of War

28.2. Sobreescritura de propiedades

Es importante aprovechar al máximo las posibilidades de tipografiado automático de partituras que LilyPond ofrece, sin ninguna intervención manual. Sin embargo, en el apartado 22.3 (pág. 32) utilizamos tímidamente la sobreescritura de propiedades para modificar el tamaño de un pentagrama. Las propiedades de los objetos gráficos tienen un valor determinado que se usa para especificar la forma en que el objeto se imprime. Hay varias instrucciones que hacen posible la modificación de estos valores, y la más frecuente es `\override`. Los valores exactos son algo que se puede determinar mediante ensayo y error, aunque existen ayudas muy valiosas como la herramienta Regla de LilyPondTool (que no explicaremos aquí). La instrucción `\override` se utiliza de la siguiente manera:

```
\override contexto.objeto #'propiedad = #valor
```

Que significa: asignar el *valor* a la *propiedad* del *objeto* dentro del *contexto*. El contexto predeterminado es *Voice* y muchas veces se puede dejar sin especificar. Veamos a continuación un ejemplo del uso de la sobreescritura de propiedades para mover objetos.

28.3. Mover objetos

Los matices dinámicos son objetos llamados internamente `DynamicText`, que se imprimen en el lugar determinado por una serie de variables. Apliquemos la formulación general de la instrucción `\override` que acabamos de mostrar, y consignemos lo siguiente para cada uno de los apartados:

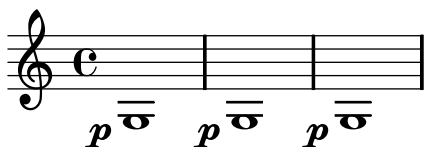
Contexto	Objeto	Propiedad	Valor
Voice	DynamicText	'extra-offset	'(-0.8 . 1)
Voice	DynamicText	'X-offset	-2.5

Estas medidas están expresadas en espacios de pentagrama, por lo que (afortunadamente) no dependen del tamaño de éste. Los dos números entre paréntesis se refieren a las dimensiones X e Y. El efecto de la sobreescritura permanece hasta que se vuelva a sobrecribir o hasta que se encuentre una instrucción `\revert` con el nombre del objeto y la propiedad. En el ejemplo se ve que las tres indicaciones están afectadas por una sola sobreescritura:

```

\override DynamicText #'extra-offset = #'(-0.8 . 1)
\override DynamicText #'X-offset = #-2.5
g1\p g\p g\p

```



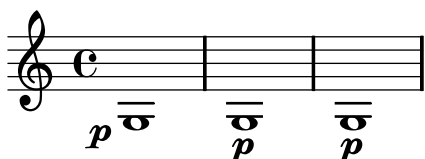
28.4. Aplicación por una sola vez

Las sobreescrituras permanecen hasta nueva orden, pero por comodidad, en caso de que sólo se necesite una vez, podemos preceder la instrucción de sobreescritura por la palabra clave `\once`. Aquí podemos ver que sólo la primera indicación dinámica está afectada por `\once \override`:

```

\once \override DynamicText #'extra-offset = #'(-0.8 . 1)
\once \override DynamicText #'X-offset = #-2.5
g1\p g\p g\p

```



28.5. Enmascarar en blanco

Cuando se quieren tapar las líneas que caen detrás de una indicación dinámica o textual, se le da un valor verdadero a la propiedad `whiteout`.

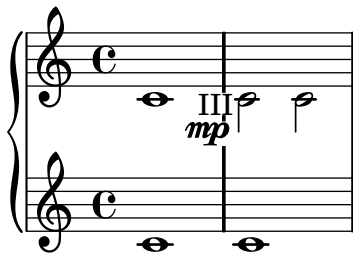
Contexto	Objeto	Propiedad	Valor
Voice	TextScript, DynamicText	'whiteout	verdadero (#t) o falso (#f)

Por ejemplo:

```

\override TextScript #'whiteout = ##t

```



28.6. Grosor de las líneas del pentagrama

Pruebe la siguiente sobreescritura para conseguir líneas más delgadas en pautas sueltas o en toda la partitura:

Contexto	Objeto	Propiedad	Valor
Staff, Score	StaffSymbol	'thickness	#(magstep -3)

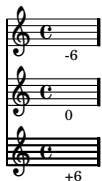
Por ejemplo:

```

\new Staff \with { \override StaffSymbol #'thickness = #(magstep -3) }

```

En el ejemplo que aparece a continuación podemos ver dos aplicaciones de sentido opuesto, y el aspecto predefinido en segundo lugar.



28.7. Separación de pautas

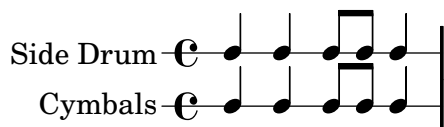
El espaciado vertical es un asunto delicado. El ajuste de la separación entre cada pauta y la siguiente se hace también mediante sobreescritura de propiedades, pero desde la versión de desarrollo 2.13 de LilyPond los objetos y propiedades han cambiado de nombre y la sobreescritura tiene otro formato. En cualquier caso aquí tenemos las que valdrán para la futura versión estable 2.14:

Contexto	Objeto	Propiedad	Valor
Staff	VerticalAxisGroup	'next-staff-spacing	#((space . 6) (padding . 0))

Esta sobreescritura se puede almacenar en una variable para utilizarla repetidas veces (sólo para 2.13):

```
juntaPauta = \with {
  \override VerticalAxisGroup #'next-staff-spacing =
    #'((space . 3) (padding . 0))
}

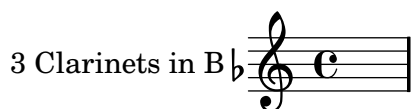
<<
\new RhythmicStaff
  \with {
    instrumentName= "Side Drum"
    \juntaPauta
  }
  { c4 c c8 c c4 }
\new RhythmicStaff
  \with {
    instrumentName= "Cymbals"
    \juntaPauta
  }
  { c4 c c8 c c4 }
>>
```



28.8. Notas

- Para los nombres de instrumentos que contienen un bemol, use `\flat` dentro del elemento de marcado.

```
\new Staff
  \with {
    instrumentName= \markup { "3 Clarinets in B" \flat }
  }
  s1
```



- La instrucción de marcado `\smallCaps` produce un estilo “versalitas” en que las minúsculas son mayúsculas pequeñas:

```
\markup { \smallCaps "Marte" }
```

MARTE