

Concepts utiles pour retoucher une partition LilyPond

Une constellation de commandes...

- ▶ `\set`
- ▶ `\unset`
- ▶ `\override`
- ▶ `\revert`
- ▶ `\tweak`
- ▶ `\once`
- ▶ `\single`
- ▶ `\temporary`
- ▶ `\propertyTweak`
- ▶ `\overrideProperty`
- ▶ `\undo`
- ▶ `\consists`
- ▶ `\remove`
- ▶ ...

Comment est compilé un fichier ?

Traitement de « x.ly »

Analyse...

Interprétation en cours de la musique...

Pré-traitement des éléments graphiques...

Détermination du nombre optimal de pages...

Répartition de la musique sur une page...

Dessin des systèmes...

Conversion à « x.pdf »...

Compilation menée à son terme, avec succès.

Un concept omniprésent : les propriétés

Une propriété est un réglage d'un objet. Les propriétés sont partout : dans les contextes, les grobs, les `\paper`, ...

Étape 1 : analyse

Analyse...

LilyPond interprète la syntaxe à l'aide d'un *analyseur syntaxique*.

Passe de

{ c' }

à

```
(make-music
  'SequentialMusic
  'elements
  (list (make-music
        'NoteEvent
        'pitch
        (ly:make-pitch 0 0)
        'duration
        (ly:make-duration 2))))))
```

Étape 1 : analyse

Dans le fichier, on peut utiliser des fonctions.

```
\transpose c d { c' }
```

Durant l'analyse, ces fonctions sont exécutées.

```
(make-music
  'TransposedMusic
  'element
  (make-music
    'SequentialMusic
    'elements
    (list (make-music
          'NoteEvent
          'pitch
          (ly:make-pitch 0 1)
          'duration
          (ly:make-duration 2))))))
```

Étape 2 : interprétation (ou traduction)

Interprétation en cours de la musique...

L'entrée se compose d'expressions musicales.

Étape 2 : interprétation (ou traduction)

Interprétation en cours de la musique...

L'entrée se compose d'expressions musicales.

La sortie graphique se compose d'objets graphiques : têtes de notes, hampes, ligatures, ...

Étape 2 : interprétation (ou traduction)

Interprétation en cours de la musique...

L'entrée se compose d'expressions musicales.

La sortie graphique se compose d'objets graphiques : têtes de notes, hampes, ligatures, ...

L'interprétation fait le lien. Transforme la musique en objets graphiques.

Les acteurs de l'interprétation

- ▶ Contextes,
- ▶ Traducteurs.

À quoi sert un traducteur ?

La plupart des traducteurs servent à créer les objets graphiques.

Par exemple :

- ▶ `Note_heads_engraver` \longrightarrow `NoteHead`
- ▶ `Stem_engraver` \longrightarrow `Stem`
- ▶ `Staff_symbol_engraver` \longrightarrow `StaffSymbol`

À quoi sert un traducteur ?

La plupart des traducteurs servent à créer les objets graphiques.

Par exemple :

- ▶ `Note_heads_engraver` → `NoteHead`
- ▶ `Stem_engraver` → `Stem`
- ▶ `Staff_symbol_engraver` → `StaffSymbol`

D'autres ont des fonctions auxiliaires.

- ▶ `Spanner_break_forbid_engraver` → empêche qu'un saut de ligne n'apparaisse au milieu d'un glissando ou une ligature,
- ▶ `Pitch_squash_engraver` → met les notes toutes à la même position verticale,
- ▶ `Repeat_acknowledge_engraver` → décide du type de barre de mesure en fonction des répétitions et autres.

À quoi sert un traducteur ?

La plupart des traducteurs servent à créer les objets graphiques.

Par exemple :

- ▶ `Note_heads_engraver` → `NoteHead`
- ▶ `Stem_engraver` → `Stem`
- ▶ `Staff_symbol_engraver` → `StaffSymbol`

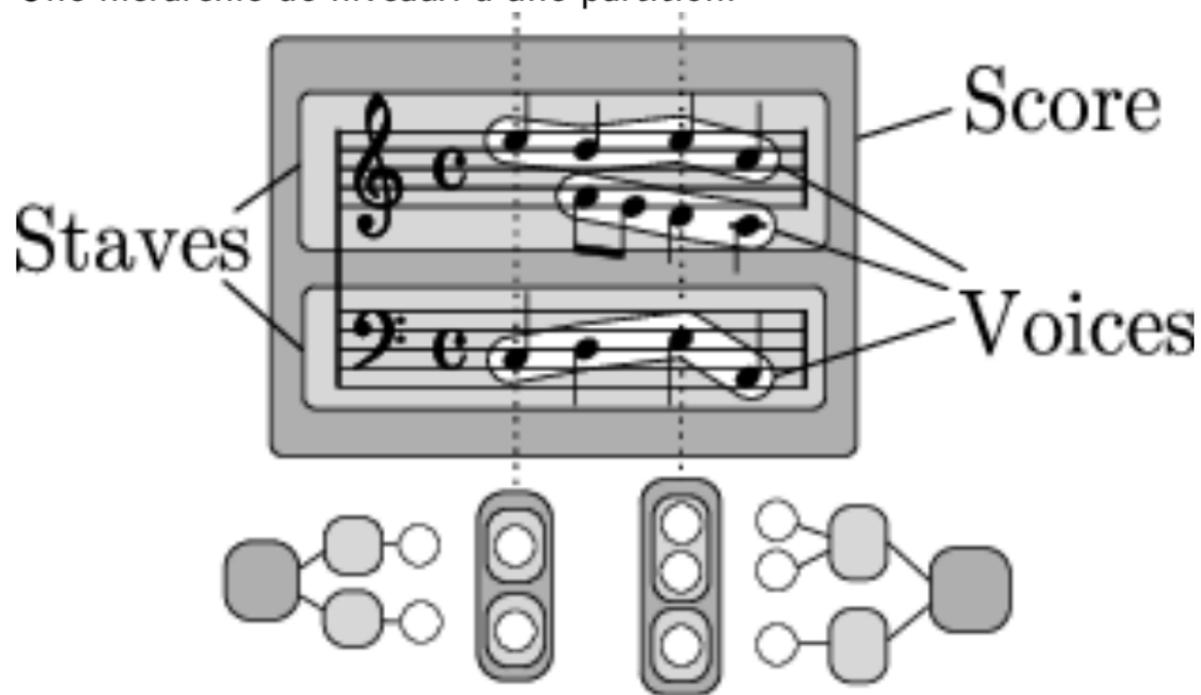
D'autres ont des fonctions auxiliaires.

- ▶ `Spanner_break_forbid_engraver` → empêche qu'un saut de ligne n'apparaisse au milieu d'un glissando ou une ligature,
- ▶ `Pitch_squash_engraver` → met les notes toutes à la même position verticale,
- ▶ `Repeat_acknowledge_engraver` → décide du type de barre de mesure en fonction des répétitions et autres.

Traducteur = graveur (*engraver*), exécutant (*performer*) ou les deux.

Contextes

Une hiérarchie de niveaux d'une partition.



Contextes courants

- ▶ Score
- ▶ Staff
- ▶ Voice
- ▶ Lyrics
- ▶ ChordNames
- ▶ TabStaff, TabVoice

À quoi sert un contexte ?

Les contextes contiennent deux choses :

- ▶ Des propriétés,
- ▶ Des traducteurs (*translators*).

Les traducteurs communiquent avec l'utilisateur et entre eux à l'aide des propriétés des contextes.

Étape 3 : début du formatage

Pré-traitement des éléments graphiques...

À ce stade, la traduction est terminée, et certains objets commencent à être dessinés. Par contre, toute la partition est sur un seul système.

Étape 4 : calcul des sauts de page

Détermination du nombre optimal de pages...

Calcul des sauts de page à l'aide d'un algorithme (obscur).
S'appuie sur l'étape précédente. Découpe le grand système en systèmes plus petits de la taille de la page.

Étape 5 : espacement vertical

Répartition de la musique sur une page...

Les systèmes et portées sont répartis verticalement sur chaque page.

Étape 6 : fin du formatage

Dessin des systèmes...

Les derniers objets graphiques sont dessinés.

Étape 7 : conversion en PDF

Conversion à « x.pdf »...

LilyPond sort du code PostScript. GhostScript se charge de le convertir en PDF.

Où peut-on agir dans ce processus ?

Réglages courants :

- ▶ Écriture de fonctions musicales personnelles,

Où peut-on agir dans ce processus ?

Réglages courants :

- ▶ Écriture de fonctions musicales personnelles,
- ▶ Modification des propriétés de contexte,

Où peut-on agir dans ce processus ?

Réglages courants :

- ▶ Écriture de fonctions musicales personnelles,
- ▶ Modification des propriétés de contexte,
- ▶ Ajout ou suppression de graveurs,

Où peut-on agir dans ce processus ?

Réglages courants :

- ▶ Écriture de fonctions musicales personnelles,
- ▶ Modification des propriétés de contexte,
- ▶ Ajout ou suppression de graveurs,
- ▶ Modification des propriétés d'objet graphique.

Où peut-on agir dans ce processus ?

Réglages courants :

- ▶ Écriture de fonctions musicales personnelles,
- ▶ Modification des propriétés de contexte,
- ▶ Ajout ou suppression de graveurs,
- ▶ Modification des propriétés d'objet graphique.

D'autres possibilités sont ouvertes par la programmation Scheme, comme le codage de nouveaux graveurs.

Pour modifier une propriété de contexte

- ▶ `\set`
- ▶ `\unset`

\set

Modifie une propriété de contexte.

Syntaxe :

```
\set propriété = valeur
```

ou

```
\set Contexte.propriété = valeur
```

\unset

Annule un \set.

```
\unset propriété
```

ou

```
\unset Contexte.propriété
```

\override

Modifie une propriété d'objet graphique.

```
\override Grob.propriété = valeur
```

ou

```
\override Contexte.Grob.propriété = valeur
```

\revert

Comme `\unset` pour `\set`.

```
\revert Grob.propriété
```

ou

```
\revert Contexte.Grob.propriété
```

\once

Restreint une modification à un instant musical.

```
\once \set ...
```

```
\once \override ...
```

`\tweak`

Problème : comment « pointer du doigt » un objet graphique ?

`\override` → nom et contexte. `\tweak` → événement source.

\tweak

Problème : comment « pointer du doigt » un objet graphique ?

`\override` → nom et contexte. `\tweak` → événement source.

`\tweak propriété valeur élément`

ou

`\tweak Grob.propriété valeur élément`

Pas de signe =

\single

`\single` convertit un `\override` en `\tweak`.

```
\single \omit NoteHead
```

\undo

\undo sert à faire automatiquement des \unset ou \revert.

```
\undo \stemUp
```

`\consists et \remove`

Ajoute un graveur.

```
\layout {  
  \context {  
    \Score  
    \remove Timing_translator  
    \remove Default_bar_line_engraver  
  }  
  \context {  
    \Staff  
    \consists Timing_translator  
    \consists Default_bar_line_engraver  
  }  
}
```

Trouver les propriétés

Allons sur <https://lilypond.org/doc/v2.23/Documentation/internals/index.html>.