

# HARPOTAB

Gravez vos tablatures d'harmonica avec LILYPOND



HARPOTAB est un ensemble de fonctions permettant de transformer très facilement une partition de musique pour LILYPOND en superbe tablature pour harmonica.

## FONCTIONNALITÉS

### PRÉ-REQUIS

HARPOTAB ne fonctionne qu'avec une version de LILYPOND supérieure ou égale à 2.16.0.

### DISPONIBILITÉ DES OPTIONS

Les options décrites ne sont pas encore toutes implémentées. Dans ce cas le numéro de version où elles seront disponibles est indiqué entre parenthèses.

### AVERTISSEMENT

Cette documentation est destinée au plus grand nombre, familier ou non de la syntaxe des fichiers LILYPOND, elle sera par conséquent probablement inutilement détaillée pour les habitués, qu'ils veuillent bien être indulgents et éventuellement la lire en diagonale.

- Saisie sous forme de partition (notes de musique) ou de tablature (numéros de trous/altérations).
- Signes d'interprétation spécifiques : notes fantômes, *tongue slapping*...
- Génération automatique de la partition ou tablature à partir de la saisie.
- Compatible avec tous les accordages existants. Seul le modèle *Richter standard* a été implémenté mais un outil de création d'autres modèles est fourni.
- Transposition automatique pour toutes les tonalités de partition ou d'harmonica.
- Support des systèmes de tablature à une ou deux lignes (**A**spiré/**S**oufflé).
- Possibilité d'inverser les lignes : **A/S** ou **S/A**.
- Personnalisation des clés de tablatures : **D/B** au lieu de **A/S**, par exemple.
- Possibilité de masquer individuellement la ligne de la tablature, ses barres de mesures ou ses silences.
- Possibilité d'afficher les altérations à la suite du numéro de trou (mode linéaire) mais également au-dessus (ou en-dessous pour les notes soufflées).
- Possibilité de personnaliser tous les symboles : note soufflée, aspirée, altérations (*bend* et *over-tone*), piston (pour les chromatiques) et de nombreuses autres indications ainsi que leur placement avant ou après le numéro du trou.
- En cas de notes pouvant être jouées de plusieurs façons différentes, les notes alternatives seront indiquées en exposant (si l'option est activée).
- Les notes alternatives sont sélectionnables manuellement, ce qui permet de voir et tester toutes les possibilités afin d'affiner progressivement l'écriture.
- Si l'option d'affichage des notes alternatives n'est pas activée il est cependant possible d'activer une option signalant leur l'existence.
- Accessibilité : amélioration de la lisibilité avec la possibilité d'obtenir des tablatures en couleur (personnalisables et au format RGB) en fonction du sens de jeu, ou d'afficher les notes aspirées encadrées au lieu de la notation habituelle.
- Compatible avec le greffon LILYJAZZ qui permet d'obtenir des partitions et tablatures utilisant une police de type manuscrite façon Jazz.

## TABLE DES MATIÈRES

Fonctionnalités.....	1
Démarrage rapide.....	3
Installation.....	3
Le fichier modèle empty-harpoTab.ly.....	4
Accordages pré-enregistrés.....	4
Types d'accordage.....	4
Noms d'instruments.....	4
Considérations algorithmiques.....	5
Configuration d'harpoTab.....	5
Les indispensables.....	5
Tonalités.....	5
Accordage.....	6
Les facultatives.....	6
Système de tablature.....	6
Notes alternatives.....	7
Altérations.....	7
Harmonicas chromatiques.....	7
Accessibilité.....	8
Écriture de la musique.....	8
Règles générales.....	8
Écriture de la mélodie.....	8
Notes alternatives.....	8
Note inexistante.....	9
Signes d'interprétation.....	9
Notes fantômes.....	9
Accords.....	9
Blocages de langue.....	9
Signes spécifiques à l'harmonica.....	9
Glissandos.....	9
Vocalisation de consonnes et autres sons.....	9
Rendu graphique de la mélodie.....	10
Rendu graphique de la tablature.....	10
Affichage des accords.....	10
Écritures alternatives de la musique.....	10
Le langage tabmode.....	11
Annexes.....	11
Remerciements.....	11
Licence.....	11
Nouvelle configuration d'accordage.....	12
Compatibilité avec LilyJazz.....	12
Installation.....	12
Utilisation.....	13
Comment ça marche ?.....	13
Le principe.....	13
Le mode tablature.....	14
Découvrir Lilypond avec harpoTab.....	15
Utilisation de Lilypond.....	15
En-tête de morceau.....	15
Expressions musicales et variables.....	15
Données globales.....	15
Écriture des notes.....	16
Écriture de la musique.....	16
Les répétitions.....	17
Les anacrouses.....	17
Les accords.....	17
Autres variables et Score.....	17
La compilation.....	17

# DÉMARRAGE RAPIDE

Vous voulez utiliser HARPO<sup>TAB</sup> sans plus attendre ? Si vous connaissez déjà LILYPOND, pas de problème, sinon lisez *Découvrir Lilypond avec harpoTab*, page 15.

- **Installation** : décompressez l'archive dans le répertoire de votre choix (explications détaillées ci-dessous). Vous pourrez ultérieurement installer les fichiers d'HARPO<sup>TAB</sup> dans un répertoire dédié pour une meilleure organisation.
- **Modèle de fichier** : utilisez le fichier *empty\_harpoTab.ly* situé dans l'archive, qui est pré-configuré pour une utilisation immédiate (explications détaillées page 4). Ce fichier contient en en-tête toutes les indications nécessaires pour une configuration rapide.
- **Accordages et tonalités** : les différents accordages sont référencés page 4, et les explications sur les tonalités page 5 mais si vous utilisez un harmonica standard de type Richter en Do, vous n'avez rien à changer.
- **Configuration** : les autres options par défaut devraient vous suffire pour démarrer, sinon elles sont détaillées page 5 et largement documentées en début du fichier *empty\_harpoTab.ly*.
- **Saisie de la mélodie** : utilisez votre éditeur habituel, et saisissez la mélodie dans la variable *Melody* (explications détaillées page 8).
- **Corrections mélodiques** : si les indications de trou proposées ne vous conviennent pas, indiquez un numéro de note alternative comme expliqué dans *empty\_harpoTab.ly* et relancez la compilation, jusqu'à satisfaction. Ensuite désactivez le mode *brouillon* (explications détaillées page 8).

Pour des explications plus détaillées continuez votre lecture.

## INSTALLATION

### INSTALLATION

Les trois premiers fichiers peuvent être installés n'importe où sur votre disque dur, mais ils devront être appelés dans cet ordre au début de chaque fichier .ly, avec les chemins d'accès adéquats.

HARPO<sup>TAB</sup> est composé des fichiers suivants :

- *harpoTab-init.ly* est le fichier d'initialisation qui contient la configuration d'accordage par défaut, ainsi que quelques autres informations. Hormis pour ajouter une nouvelle configuration d'accordage, il n'est pas supposé être modifié.
- *harpoTab-setup.ly* contient toutes les indications de configuration et toutes les valeurs par défaut. Il est prévu pour enregistrer toutes vos préférences, qui n'auront ainsi pas à être écrites dans vos fichiers de partition.
- *harpoTab.ly* qui contient toutes les fonctions *Scheme*, les contextes et les raccourcis. Ce fichier ne doit surtout pas être modifié, *sauf si vous savez très exactement ce que vous faites* !
- *empty-harpoTab.ly* est un modèle de fichier que l'on peut recopier pour écrire ses partitions, il est préformaté pour l'utilisation de HARPO<sup>TAB</sup>.

Décompressez l'archive les contenant dans le répertoire de choix de votre disque dur, éventuellement dans un répertoire dédié, ou dans celui où vous rangez vos partitions.

Vous pourrez également télécharger les fichiers suivants, non indispensables mais pouvant vous être utiles :

- *harmonyTunings.ods*, classeur utilisable avec LibreOffice ou autre logiciel compatible, est un petit utilitaire permettant de générer simplement de nouvelles configurations d'accordage si nécessaire.
- le fichier de configuration et les polices de caractère de LilyJAZZ sont également disponibles sous forme d'archive séparée.

L'utilitaire de génération d'accordage peut s'enregistrer n'importe où, mais les fichiers de LilyJazz devront être traités comme les autres fichiers d'HARPO<sup>TAB</sup>.

# LE FICHER MODÈLE EMPTY-HARPO<sup>T</sup>AB.LY

Ce fichier modèle est abondamment commenté, ce qui permettra à chacun de l'adapter à sa manière.

En tête de fichier on trouvera les inclusions obligatoires des fichiers d'initialisation, de configuration et de fonctions de HARPO<sup>T</sup>AB, ainsi que la totalité des options modifiables localement.

On trouve ensuite plusieurs variables dédiées à la saisie de la musique :

- le bloc `\header`, qui permet d'enregistrer le titre, l'auteur, etc. ;
- *Global* : qui est utilisée pour noter les indications de clés, tempo, tonalité, etc. que vous pourrez préférer noter avec la musique ;
- *Melody* qui est destinée à recevoir la mélodie jouée à la main droite, ce qui n'interdit nullement d'écrire tout ou partie de cette mélodie dans d'autres variables et de les insérer à cet endroit ;
- *Tablature* qui est destinée à recevoir la mélodie sous forme de numéros de trous et non de notes, ce qui sera expliqué en détail dans page ;
- *Chords* est pré-configuré pour la saisie des accords (façon accords de guitare) ;
- et enfin, *Lyrics* qui permet de saisir des paroles.

La partie restitution de la musique, à savoir le bloc `\score`, est également pré-configurée pour utiliser les variables précédentes. Normalement les seules modifications à y apporter consisteront à (dé-)commenter certaines options, tout ceci étant bien entendu expliqué dans le fichier.

Enfin, le bloc `\paper` comporte lui aussi toutes les données nécessaires pour avoir une sortie papier conforme à vos envies.

Toute cette partie de restitution, qui ne change quasiment jamais, peut être externalisée dans un fichier nommé *harpoTab-footer.ly* par exemple, qui sera appelé avec une instruction `\include` placée en fin de fichier. Cette externalisation n'est pas faite par défaut, le choix vous appartient.

## ACCORDAGES PRÉ-ENREGISTRÉS

### TYPES D'ACCORDAGE

Il existe une très grande diversité d'accordage d'harmonicas, et tous les implémenter dans HARPO<sup>T</sup>AB représenterait une énorme charge de travail. Le tableau ci-dessous représente les premiers accordages intégrés, dont il est à espérer qu'il prenne de l'ampleur grâce aux contributions de la communauté des harmonicistes.

Le code indiqué en deuxième colonne est celui à indiquer dans la configuration.

Nom de l'accordage	Code
Richter standard	<code>richt-std</code>
Lydien Seb Charlier	<code>lyd-char</code>

### NOMS D'INSTRUMENTS

Les noms d'instrument sont par défaut internationalisés. Par exemple, « *Variation numéro 1 du Melody Maker* » sera écrit « *Melody Maker variation 1* ». Le nom de l'instrument sera automatiquement inséré dans tous les blocs `\header` (balise `instrument`) de vos fichiers, sauf si vous surchargez la-dite balise du fichier.

#### PERSONNALISER

Il sera probablement judicieux de retirer une grande partie des options de configuration et ne garder que celles convenant à vos besoins.

#### AUTRES NOMS

Il est parfaitement possible de choisir d'autres noms de variables à condition de les modifier partout où ils apparaissent. Donc, l'avertissement d'usage vaut ici aussi : *ne changez rien sauf si vous savez très exactement ce que vous faites...*

#### CONTRIBUER

Si votre harmonica ne figure pas dans le tableau ci-contre, pas de panique !

Il vous suffit de créer son tableau d'accordage à l'aide du fichier *harmonyTunings.ods* fourni. Voir les indications à ce sujet page 12.

# CONSIDÉRATIONS ALGORITHMIQUES

## MODIFIER UN ACCORDAGE

Si une note par défaut ne correspond pas à votre style de jeu, et que vous voulez changer celle-ci de façon permanente, vous pouvez le faire dans le fichier `harpoTab-init.ly`, en inversant la note par défaut et l'alternative. Pour ce faire reportez-vous aux indications de la page 12.

Il serait absolument illusoire de vouloir écrire un algorithme qui déterminerait comment doit être jouée une note disponibles à plusieurs endroits. Ceci dépend non seulement des capacités du musicien à obtenir plus ou moins facilement des notes altérées, mais aussi du morceau lui-même, où les enchaînements de notes et de sens de jeu peuvent modifier les préférences.

Néanmoins, afin de pouvoir produire facilement et rapidement une tablature, il est nécessaire de déterminer une note par défaut dans le cas de notes dupliquées.

Les critères de détermination ont été les suivants :

- une note non altérée aura la priorité sur une note altérée ;
- si les deux notes (ou plus) sont altérées le *bend* aura la priorité sur l'*over-tone* ;
- si aucune des notes n'est altérée (ou si elles le sont de la même façon) la note la plus proche du centre de l'harmonica aura la priorité.

Cependant il n'est pas exclus qu'une version ultérieure implémente un ou plusieurs algorithmes, en fonction des besoins et des demandes, dont voici quelques exemples possibles :

- on ne peut pas aspirer ou souffler plus d'un certain nombre de notes à la suite ;
- écrire un signe de respiration imposera de changer le sens de celle-ci ;
- un mode *entraînement* choisira par défaut la note la plus difficile à jouer...

## CONFIGURATION D'HARPO TAB

### SIMILARITÉ

Les déclarations de configuration s'écrivent de la même façon dans les deux cas.

L'utilisation d'`HARPO TAB` impose la saisie de quelques déclarations de configuration supplémentaires à celles habituelles de `LILYPOND`.

Les déclarations peuvent se faire de deux façons, l'une globale, écrite dans un fichier de configuration séparé nommé `harpoTab-setup.ly`, ou de façon locale dans chaque fichier de partition. Ces deux modes de déclaration sont volontairement redondants.

### SYNTAXE

Les fonctions de configuration (sauf les expressions musicales) s'écrivent : `#(define param value)` où `param` est le paramètre à configurer et `value` la valeur à lui attribuer, dont l'écriture dépend du type de l'information.

#### VARIABLE

On attribue à `param` le contenu d'une autre variable, par exemple `et1` ou `jp1`.

#### TEXTE

On écrira `value` sous la forme "A".

#### VRAI OU FAUX

Les valeurs `#t` et `#f` signifient respectivement vrai (*true*) ou faux (*false*). Il suffit de les inverser pour modifier le comportement de la fonction.

En effet, le comportement d'`HARPO TAB` est de lire d'abord le fichier de configuration, puis les indications locales, lesquelles surchargeront la configuration globale.

Chacun déterminera quelles sont les indications à conserver dans les fichiers de partition et celles à n'indiquer que dans le fichier de configuration. On pourra donc à la limite n'avoir quasiment aucune déclaration dans les fichiers de partitions, ou au contraire ne pas avoir de fichier global et tout écrire dans les partitions.

## LES INDISPENSABLES

Seules trois informations sont indispensables pour générer une tablature.

### TONALITÉS

La tonalité d'entrée (`songTone`) est celle de l'harmonica qui peut jouer la mélodie telle qu'elle est écrite dans la partition. Ce n'est pas forcément celle du morceau lui-même, un morceau en ré mineur se jouant sur un harmonica en sol. Si une tablature existe déjà vous connaissez sans doute cette indication, sinon vous pouvez vous baser sur l'armure. Si la tablature obtenue vous semble trop difficile à jouer, ou s'il manque des notes, essayez une tonalité voisine (une altération de plus ou de moins).

La tonalité de sortie est (`harmonicTone`) celle de l'harmonica qui jouera la mélodie, parfois le même mais pas forcément, c'est pourquoi les deux informations sont indispensables.

L'existence de ces deux informations permet en outre d'automatiser les transpositions.

```
songTone = sol
```

```
harmonicTone = sol
```

## ACCORDAGE

L'accordage de votre harmonica (`tunOut`) est indispensable pour que la tablature générée y corresponde.

```
 #(define tunOut code)
```

L'accordage de l'harmonica de saisie (`tunIn`) ne l'est par contre que si vous saisissez votre musique en mode tablature, afin que les numéros de trous soient convertis correctement.

```
 #(define tunIn code)
```

## LES FACULTATIVES

`HARPOTAB` vous propose de nombreuses options pour obtenir la tablature de vos rêves.

### SYSTÈME DE TABLATURE

Les deux systèmes de tablature communément répandus sont soit à une ligne avec des indications de sens de jeu, soit sur deux lignes, les notes aspirées en haut et les soufflées en bas. La valeur par défaut est le système à deux lignes, défini avec :

```
 #(define db-mode #t) (d et b comme draw et blow, aspirer et souffler en anglais).
```

#### SYSTÈME À DEUX LIGNES

La clé de tablature est constituée de deux lettres personnalisables :

```
 #(define tabClef '("A" "S")) la première lettre pour les notes aspirées, et la seconde pour les notes soufflées.
```

En outre l'instruction 

```
 #(define reverse-db-mode #f)
```

 vous permettra d'inverser les deux lignes et de passer en **S A**.

#### SYMBOLES ET POSITIONS

Un certain nombre d'indications de jeu, tels que note soufflée ou aspirée, sont affichées sous forme de symboles, placés avant, après, ou à la verticale du numéro de trou. Dans ce dernier cas, ce sera au-dessus pour une indication de trou placée au-dessus de la ligne, sinon au-dessous. Pour tous ces symboles, la notation sera 

```
 #(define symbol '("x" -1))
```

 où `x` est le caractère à afficher, et le chiffre sa position : -1 avant, 0 à la verticale, et 1 après. Si on choisit de ne pas afficher de symbole, il suffira de laisser vide l'indication de caractère : "".

#### SYSTÈME À UNE LIGNE

Le sens de jeu n'étant plus indiqué par l'emplacement du numéro de trou sur une ligne ou l'autre, il est nécessaire de l'indiquer au moyen d'un symbole adapté.

Les indications les plus courantes sont les signes + et -. Mais il est bien entendu possible de leur préférer d'autres symboles, tels que < ou > par exemple. Ces symboles ne peuvent être placés qu'avant ou après le numéro (-1 ou 1).

- ```
 #(define draw-symbol '("-" -1))
```

 pour le symbole de note aspirée ;
- ```
 #(define blow-symbol '("<" 1))
```

 pour le symbole de note soufflée.

#### LIGNES ET BARRES DE MESURES

Les tablatures à une seule ligne ne nécessitent pas forcément l'affichage de la dite ligne et celle-ci peut donc être masquée.

```
 #(define hide-staff #f)
```

On peut également masquer les barres de mesure, ce qui est recommandé si on masque la ligne.

```
 #(define hide-barline #f)
```

#### INDICATIONS D'OPTIONS

Toutes les options décrites ici sont non seulement pré-déclarées dans le fichier de configuration mais également présentes dans leur intégralité en début du fichier `harpoTab.ly`.

Cependant, hormis pour les indications dont il est explicitement indiqué qu'elles doivent se trouver dans les fichiers de musique, vous préférerez probablement les définir une bonne fois pour toute dans votre fichier de configuration, et ne pas les voir apparaître dans les fichiers de vos morceaux de musique.

#### INVERSION

Attention à ne pas inverser les lettres A et S dans la définition de la clé, en imaginant que cela inversera les lignes : les notes aspirées s'afficheraient toujours au-dessus de la ligne, quand bien même elle porterait l'indication S.

#### SYNTAXE

Tous les symboles doivent être entourés de guillemets informatiques " et séparés par une espace.

## HAMPES

Par défaut seuls les numéros de trous sont affichés, mais on peut choisir d'afficher également les hampes des notes dans la tablature.

```
#(define show-stems #f)
```

## LIGNES DE SUIVI ET SILENCES

Les notes dont la durée est supérieure à un temps (tel que défini par la métrique) peuvent être prolongées par une ligne horizontale indiquant de façon graphique leur durée. Ceci peut être utile pour qui ne sait pas lire le rythme.

```
#(define show-span #t)
```

Dans ce cas on pourra également désactiver l'affichage des silences.

```
#(define hide-rests #f)
```

## NOTES ALTERNATIVES

Si une note peut être jouée de plus d'une façon, les alternatives seront affichées sous forme d'exposant, mais on peut les masquer pour alléger la tablature.

```
#(define warn-altern #t)
```

Dans ce cas les notes en question seront simplement affichées en italique pour signaler l'existence de notes alternatives, ce qui peut aussi être désactivé.

```
#(define show-altern #t)
```

Cependant, le style italique n'existant pas sur toutes les polices, il est également possible d'opter pour un caractère de substitution (position avant ou après uniquement) :

```
#(define warn-symbol '("." -1))
```

## ALTÉRATIONS

### AFFICHAGE

Les altérations peuvent être écrites soit après le numéro de trou (mode linéaire), soit à sa verticale.

### SYMBOLES D'ALTÉRATIONS

Les signes utilisés pour afficher les altérations sont totalement personnalisables. On pourra soit choisir un des divers modèles existant, soit en créer un ex-nihilo.

	<b>Bends</b>	<b>Over-tones</b>	<b>Affichage</b>
<i>alt1</i>	' " '''	o o <sup>2</sup> o <sup>3</sup>	après le numéro
<i>alt2</i>	- = ≡	o oo ooo	à la verticale du numéro

Pour sélectionner un de ces modèles, il faut l'indiquer dans les instructions adéquates :

```
#(define alt-model alt1).
```

### PERSONNALISATION

Il suffira d'indiquer non plus le numéro de modèle mais une liste (ou plutôt un vecteur pour utiliser la dénomination du langage utilisé, à savoir le *scheme*) :

```
#(define alt-model '(#" ":" "::" "::::" "●" "●●" "●●●") 0)) pour un positionnement vertical.
```

## HARMONICAS CHROMATIQUES

Lorsqu'il faut indiquer une utilisation de la tirette (ou piston) le choix par défaut est d'afficher le numéro de bouton dans un cercle (*bar0*), mais on peut lui préférer un symbole alphabétique que l'on pourra personnaliser comme les altérations :

```
#(define bar-symbol bar3) OU #(define bar-symbol '("←" 0)).
```

<b>bar1</b> :	★ (avant)	<b>bar2</b> :	◀ (après)	<b>bar3</b> :	← (verticale)
---------------	-----------	---------------	-----------	---------------	---------------

### ACCORDS

Pour les accords, qui nécessitent plus de place qu'une simple note, l'affichage des notes alternatives est d'office désactivé, mais pas l'affichage en italique.

### BON CARACTÈRE

Il est préférable de choisir un caractère relativement discret et étroit, tels que le point, ou le point d'exclamation si on souhaite un caractère plus visible.

### PERSONNALISER

Attention à bien respecter la syntaxe : un ' et un # avant les parenthèses et des " pour encadrer les symboles.

Tous les caractères ne conviennent pas pour indiquer une altération. L'esthétique et la lisibilité sont laissées à votre appréciation.

En outre il est impératif de saisir six caractères ou suites de caractères, et de tenir compte de l'espace nécessité par ces caractères, qui doit rester cohérent avec les choix par défaut.

Enfin pensez au fait que les symboles existant ont pour particularité d'avoir un positionnement vertical exactement au milieu de la ligne, et que si ceux que vous choisissez en remplacement n'ont pas ce comportement, ils ne s'afficheront pas correctement (voir page).

## ACCESSIBILITÉ

Quelques options inhabituelles mais bien utiles pour faciliter la lecture.

### MODE COULEUR

Les indications de trous de la tablature peuvent être affichées dans une autre couleur que les notes et portées. Toutes les couleurs s'indiquent sous forme **RGB**.

```
$(define def-color '( 0 0 0 )) couleur générale de la tablature
```

Si le mode couleur est activé, les notes aspirées et soufflées seront affichées dans une couleur différente, personnalisables.

```
$(define color-score #f) partition en couleur
```

```
$(define color-tab #f) tablature en couleur
```

```
$(define blowcolor '( 0 0 255)) couleur des notes soufflées
```

```
$(define drawcolor '( 255 0 0)) couleur des notes aspirées
```

```
$(define errorcolor '( 255 0 0)) couleur des notes inexistantes
```

### MODE ENCADREMENT

En système à une ligne, on pourra préférer un cadre autour du numéro de trou au lieu du symbole de sens aspiré. Les notes alternatives seront affichées en dehors de ce cadre : `$(define boxmode #f)`

#### CUMUL D'OPTIONS

Si on cumule les options d'encadrement et de tablature colorée, les encadrements seront également colorés.

# ÉCRITURE DE LA MUSIQUE

## RÈGLES GÉNÉRALES

Autant il est indispensable de respecter strictement la syntaxe des indications de configuration pour que `HARPO1TAB` puisse générer une tablature répondant à vos besoins, autant les règles de saisie de la musique sont moins strictes tant qu'elles respectent celles de `LILYPOND`.

Néanmoins toutes les explications qui vont suivre considèrent que les données musicales sont situées dans des variables portant les noms suivants ;

- la mélodie du morceau se trouve dans une variable nommée *Melody* ;
- si vous saisissez le morceau sous forme de tablature, ce sera dans *Tablature* ;
- les accords du morceau se trouvent dans une variable nommée *Chords*.

## ÉCRITURE DE LA MÉLODIE

Saisissez celle-ci de la façon habituelle, dans la variable *Melody*. Vous devrez cependant penser à ajouter là où elles sont nécessaires les indications dont l'explication suit.

### NOTES ALTERNATIVES

Les notes alternatives sont indiquées en exposant de la note par défaut, avec les caractéristiques suivantes, quel que soit le type de tablature choisi :

- les altérations seront affichées après le numéro de trou (' ou °) ;
- le sens de jeu (aspirer ou souffler) sera indiqué uniquement avec un signe moins devant une note aspirée ;
- l'indication de tirette (piston) se fera uniquement avec un cercle autour du numéro.

Pour imposer une note alternative si la note par défaut ne correspond pas au trou que l'on veut utiliser, il suffit de l'indiquer dans la partition avec un `\1` pour la première note alternative et `\2` pour la deuxième.

#### AUTRES NOMS

Il est parfaitement possible de choisir d'autres noms de variables à condition de les modifier partout où ils apparaissent. Donc, l'avertissement d'usage vaut ici aussi : *ne changez rien sauf si vous savez très exactement ce que vous faites...*

#### FLEXIBILITÉ

En réalité, on peut saisir n'importe quel chiffre différent de 0 pour afficher la note alternative.

Cependant, s'il y a deux notes alternatives et que l'on veut la première, il faudra impérativement saisir un 1.



Exemple : `g,4\1` pour jouer le sol grave en aspirant dans le deuxième trou au lieu de souffler dans le troisième. Si l'affichage des notes alternatives est activé celui-ci se mettra à jour, note par défaut deviendra alternative et sera affichée comme telle.

## NOTE INEXISTANTE

Au cas où une note n'existerait pas pour la tonalité et la configuration d'accordage sélectionnées, le numéro de trou sera remplacé par un ? rouge.

## SIGNES D'INTERPRÉTATION

Ils s'utilisent comme tous les autres signes d'interprétation, en faisant suivre la note de leur indication : `note\code`.

### POSITION

L'indication de note fantôme doit se faire avant la note, puisqu'il s'agit de remplacer la tête de note habituelle par un x.

## NOTES FANTÔMES

On utilisera l'instruction `\gn` (comme *ghost note*) avant la ou les notes à transformer ainsi : `\gn note` ou `\gn {note1 note2 ...}` pour traiter plusieurs notes successives.

Ceci aura pour effet de transformer les têtes de note en x et d'afficher également ce x sur la ligne de tablature (que sa visibilité soit activée ou non).

## ACCORDS

On parlera d'accord dès que plusieurs trous sont utilisés simultanément, avec ou sans blocage de langue sur une ou plusieurs notes.

La notation se fera comme tout accord, entre `<>`. Exemple : `<c e g b>`. Les numéros de trou seront affichés entre parenthèses : `(2345)`.

## BLOCAGES DE LANGUE

Pour indiquer un blocage de langue, que ce soit dans un accord ou non, et quelle que soit la forme du blocage (en U ou non), il faudra explicitement indiquer le(s) trou(s) bloqué(s) avec l'indication `\block` : `<c e\block g\block b>`, ce qui aura pour effet d'afficher l'accord ainsi sur la tablature : `(2345)`.

### RACCOURCIS

Si une indication telle que `\block` vous semble trop longue à saisir et que vous voulez `\tb` à la place, ajoutez une ligne contenant `tb=\block` dans votre fichier au-dessus de la variable Global. Ne faites pas cette modification ailleurs, harpo-Tab n'aimerait pas trop...

## SIGNES SPÉCIFIQUES À L'HARMONICA

Description	Code	Symbole	Position
Attaque par en dessous ( <i>dip bend</i> )	<code>\atdn</code>	↵	Avant
Faire pleurer en descendant (attaque du haut)	<code>\atup</code>	↶	Après
Tongue slapping	<code>\slap</code>	∇	Centré
Coup sec de la main sur le coté de l'harmonica	<code>\tap</code>	⊥	Avant
Arrêt net de la note	<code>\stop</code>		Après
Déplacement latéral avec la tête ou l'harmonica	<code>\sld</code>	↔	Centré
Déplacement latéral par le mouvement de la langue	<code>\slt</code>	↔	Centré
Flutter Tongue	<code>\flt</code>	↔	Centré

Pour tous ces symboles on pourra personnaliser le caractère à utiliser et sa position.

## GLISSANDOS

Ils seront notés de façon habituelle dans Lilypond, avec `\glissando`. Il est possible de les faire démarrer d'une note invisible (`\hideNotes`) qui devra cependant être indiquée. Ils seront transcrits avec des flèches `↑` et `↓` dans la tablature, avec ou sans le numéro de trou de la note de départ selon que la note est cachée ou non.

`g8\glissando c` note cachée : `\hideNotes c,8\glissando \unHideNotes c'`

## VOCALISATION DE CONSONNES ET AUTRES SONS

Toutes ces indications sont trop nombreuses et trop précises pour se satisfaire d'un simple symbole, on utilisera par conséquent un markup explicite qu'il faudra position-

### HABILLAGE

On peut transformer le glissando (dans la partition uniquement) en zigzag en décommentant l'instruction `\override Glissando #'style = #'zigzag` dans la variable Global.

ner avec un ^ ou un \_ en fonction du système de tablature utilisé et de la note à laquelle elle s'applique : do4\_ "Tah hooka tooka hooka" ou do4^ "Wah" pour les effets de Wah-Wah, "Rrrl" pour les roulements de la langue, "Grrr" pour les roulements de gorges (ou grognements)...

Vous avez toute liberté pour vous exprimer.

## RENDU GRAPHIQUE DE LA MÉLODIE

Afin d'avoir une partition qui ne contienne pas les éventuelles indications de notes alternatives, altérations et autres, lesquelles sont inutiles à ce niveau, il suffit d'utiliser le contexte prédéfini nommé `melodyStaff`, de la façon suivante :

```
\new Staff \with { \ melodyStaff } << \transpose \songTone \harmonicTone \Melody >>
```

La commande `\transpose \songTone \harmonicTone` permet d'obtenir automatiquement la partition dans la tonalité désirée, comme expliqué plus haut.

## RENDU GRAPHIQUE DE LA TABLATURE

Pour transformer la partition en tablature, il suffit d'utiliser la fonction `\harpoTab`, associée à son contexte prédéfini :

```
\new Staff \with { \tabStaff } << { \ harpoTab \Melody } >>
```

Il ne faut pas à cet endroit indiquer d'instruction de transposition, celle-ci se faisant obligatoirement de `songTone` vers le do, elle est directement prise en charge par la fonction `mkTab`. Par contre, comme indiqué plus haut, il pourra être nécessaire, selon les tonalités de départ et d'arrivée, de jouer sur les octaves de `songTone` et `harmonicTone`, voire la note de référence de la mélodie (celle qui suit l'instruction `\relative`).

Les effets de ces modifications sont les suivants :

- modifier uniquement `harmonicTone` ne changera pas la tablature, mais la partition, et ce dans le même sens que `harmonicTone`;
- modifier `songTone` et `harmonicTone` de la même façon ne changera aucunement la partition mais décalera la tablature dans le même sens ;
- modifier uniquement `songTone` changera à la fois la partition et la tablature, mais en sens inverse : la tablature suivra le sens de `songTone`, et la partition fera l'inverse, ce qu'on peut compenser en jouant de la même façon sur `\relative`.

## AFFICHAGE DES ACCORDS

Les accords seront classiquement écrits dans une variable `Chords`, qui sera affichée de façon habituelle avec un `Staff ChordNames` placé au-dessus de la portée :

```
\new ChordNames { \transpose \songTone \harmonicTone \Chords }
```

Cette notation permettra la transposition automatique des accords également.

On utilisera également selon ses habitudes soit :

```
\set chordChanges = ##t
```

pour indiquer que le nom des accords doit ou non être répété à chaque mesure même s'il ne change pas, ou :

```
\frenchChords
```

qui affiche le nom des accords en français.

### CONTEXTE

Le contexte `melodyStaff`, ainsi que tous les autres contextes, est défini à la fin du fichier `harpoTab.ly`. Il est possible de les y modifier, si les réglages par défaut ne vous conviennent pas et que vous savez quoi écrire et où...

### VALEURS PAR DÉFAUT

Elles donneront dans la majorité des cas un résultat satisfaisant, sinon il faudra probablement adapter au cas par cas.

## ÉCRITURES ALTERNATIVES DE LA MUSIQUE

Si vous ne connaissez pas suffisamment la musique pour écrire vos partitions, vous pouvez malgré tout obtenir de belles tablatures avec `HARPOTAB`. en recopiant simplement une tablature existante avec le langage `tabmode`.

### LANGAGE

Le choix des lettres est imposé par le fait qu'elles ne doivent pas correspondre à des lettres existant déjà dans les différents langages de notes de Lilypond.

### CONVENTION

On pourrait utiliser un ^ à la place du tiret de soulignement, mais ce dernier est beaucoup plus facile à saisir, raison pour laquelle il a été retenu.

### MNÉMO

Pour les francophones, il est facile de penser p = piston, pour les anglophones, ce sera p = push.

## LE LANGAGE TABMODE

Des lettres permettant de saisir des boutons et non des notes ont été rajoutées au dictionnaire de notes de LILYPOND :

- `o` (comme *output*, ou **sO**uffler, ou **blO**w) : note soufflée ;
- `i` (comme *input*, ou **In**spirer) : note aspirée.

On utilisera ces pseudo-notes de la même façon que les autres, en les faisant suivre de leur durée.

Ensuite on notera le numéro du trou en le faisant précéder d'un - (qu'il ne faut pas considérer comme le signe mathématique moins mais comme un tiret de liaison).

Une noire aspirée sur le troisième trou se notera donc `i4-3`.

Si la note est altérée ou si le piston doit être actionné, on indiquera ces informations en les faisant précéder d'un \ suivi d'une ou plusieurs lettres :

- pour les raisons indiquées plus haut on ne pourra pas utiliser le `b` pour les *bends*, Lilypond comprendrait que c'est la note si et refuserait qu'elle soit indiquée à cet endroit, on utilisera donc un `l` (comme *low*) puisque les *bends* abaissent la note ;
- les *over-notes* seront pour les mêmes raisons notés `u` (comme *up*) et non `o` ;
- le piston sera noté `sb`, comme *Sliding Bar Button*.

On mettra autant de `l` et de `u` qu'il y a de niveau d'altération, et on accolera le `l` et le `s` de `sb` dans le cas d'utilisation du piston avec altération (les deux notations `ls` et `s1` sont valides). Il ne doit absolument pas y avoir d'espace dans cette notation.

Exemple : un sol dièse sera noté `o4-1\u` ou `i4-3\111`.

Une fois la tablature compilée, les notes alternatives seront également calculées, ce qui peut vous permettre, en fonction de votre accordage, de trouver une autre façon de jouer le morceau, plus simple ou qui correspond mieux à votre style.

L'indication de note alternative se fait exactement comme pour la saisie normale, avec un \ suivi du numéro de la-dite note alternative (1 pour la première, etc.).

## ANNEXES

### REMERCIEMENTS

Bien évidemment sans l'aide des participants à la liste de discussion `lilypond-user-fr@gnu.org` je n'aurais jamais pu arriver au bout de ce projet, et tous à un moment ou à un autre ont apporté leur pierre à l'édifice, parfois sans même le savoir.

J'aimerais particulièrement citer (sans que cela puisse être considéré comme exhaustif, sans aucun caractère de préséance, et en respectant les pseudonymes pour ceux qui ont choisi de ne pas apparaître à découvert sur la liste) : Gilles Thibault, Christophe Pochon, Pierre Perol-Schneider, Jean-Charles Malahieude ; Seventies ; Gérard (pour ses idées que j'ai développées dans la section Accessibilité), Jean-Romain Roussel qui m'a incité à me remettre à mes essais de *scheme*, que j'avais abandonné par faute de résultats, et finalement harpo\_bzh pour toutes ses explications sur l'harmonica, car, le croirez-vous, je ne sais pas jouer de cet instrument, mais j'ai eu envie de relever le défi de voir si je saurais faire quelque chose.

Et bien évidemment je tiens à m'excuser d'oublier probablement encore beaucoup de contributeurs. Qu'ils n'hésitent pas à se manifester !

### LICENCE

Sauf mention contraire explicite, le contenu de l'ensemble des fichiers composant **HARPO<sup>T</sup>AB** est placé sous la licence **Creative Commons BY NC ND**, y compris la présente documentation.

# NOUVELLE CONFIGURATION D'ACCORDAGE

Afin de faciliter la conception d'une nouvelle configuration d'accordage, **HARPO TAB** est accompagné d'un fichier nommé *harmoTunings.ods* (donc lisible avec **LibreOffice** ou **OpenOffice.org**) qui automatise l'écriture du code, mais il est également parfaitement possible d'écrire le code manuellement.

Une configuration d'accordage est définie dans ce qu'on appelle une *liste associative* (en langage clair : un dictionnaire ou tableau de correspondance, selon le terme qui vous parle le plus) qui ressemble à ceci :

```
(define code
  '( ( "harmo-name" ."nom complet" )
    ( entiersigne . ( reelsigne reelsigne ... ) ) ( reelsigne . entiersigne )
  ) )
```

Seuls les termes en maigre et en italique sont à remplacer par ceux définissant une nouvelle configuration.

Le code représente un nom abrégé, comme *richt* pour Richter, ou *mel-mak* pour Melody Maker. Il est recommandé de l'abrégé suffisamment pour pouvoir ensuite noter toutes les variantes : *richt-std*, *mel-mak-var1*, etc.

Le nom complet sera noté de façon compréhensible quelle que soit la langue de l'utilisateur, et de façon la plus courte possible également. Par exemple *Variation n° 1 du Melody Maker* est à éviter, il faut lui préférer *Melody Maker Variation 1*.

La ligne importante est celle qui comprend les correspondances clé/valeur, et qui est à dupliquer autant de fois que nécessaire :

- *entiersigne* est un nombre entier signé (positif ou négatif) représentant une note sous forme de nombre de demi-tons en-dessus ou en-dessous du do central. Ceci a l'énorme avantage qu'un sol dièse ou un la bémol auront exactement le même code, et bien que ce ne soit pas la même note d'un pur point de vue musical, on utilisera bien le même trou pour jouer les deux.
- *reelsigne* est un nombre réel signé, représentant le trou à utiliser pour produire la note. Son signe représente le sens de jeu, positif pour soufflé et négatif pour aspiré. La partie entière représente le numéro de trou, et la partie décimale le code de l'altération comme indiqué dans le tableau ci-dessous. Si plusieurs trous sont possibles on les indiquera à la suite les uns des autres, la note par défaut en premier.
- la première parenthèse représente la correspondance entre une note et un ou plusieurs trous, raison pour laquelle les codes des trous sont également entre parenthèses.
- le ou les blocs de parenthèses suivantes représentent la notation inverse, la correspondance entre un numéro de trou et une note. Il peut donc y avoir plusieurs blocs.

## CODIFICATION DES ALTÉRATIONS

0	1	2	3	4	5	6	7	8
aucune	1 bend	2 bends	3 bends	1 overtone	2 overtones	3 overtones	tirette seule	tirette et 1 bend

## COMPATIBILITÉ AVEC LILYJAZZ

### INSTALLATION

Le greffon **LILYJAZZ** a été créé à la base pour de la musique Jazz, ce qui fait que certains symboles sont manquants. De plus la police de caractères qui l'accompagne ne propose pas de vraies minuscules mais des petites majuscules, ce qui n'est pas la même chose. En conséquence, cette police a été partiellement ré-écrite (il manque encore les vraies minuscules).

### PARTAGEZ

Pensez à communiquer votre configuration à l'auteur de **HARPO TAB** afin qu'elle puisse ensuite être partagée avec la communauté.

### RECOPIER

Pour commencer une nouvelle configuration manuellement, le plus simple est de recopier la plus proche de la vôtre sous un autre nom puis de modifier celle-ci d'après les indications données ci-contre.

### EMPLACEMENT DES FICHIERS

Comme pour le greffon **HARPO TAB**, les fichiers de **LILYJAZZ** peuvent être installés n'importe où. Il faudra ensuite installer la police **FreeJazz**, et inclure le fichier **Lilyjazz.ily** avec le bon chemin d'accès.

### PRÉSENCE

L'inclusion de LilyJAZZ doit impérativement se trouver avant les inclusions des fichiers de **HARPO-TAB**, sinon les configurations de celui-ci seraient écrasées.

De même le fichier *LilyJAZZ.ily* a été modifié, un certain nombre d'options ont été rajoutées, modifiées ou commentées, et il faudra donc utiliser celui-ci en lieu et place de la version originale.

## UTILISATION

La première chose à faire est d'inclure le fichier *LilyJAZZ.ily* dans votre fichier de musique, avec l'instruction habituelle :

```
\include "LilyJAZZ.ily"
```

Ensuite il faut déclarer le mode Jazz en ajoutant dans le bloc `\score` l'instruction `\jazzOn` avant l'indication de contexte, comme dans l'exemple suivant :

```
\new Staff \with { \jazzOn \melodyStaff }
```

Enfin il faut modifier l'arbre des polices dans le bloc `\paper`. On peut soit remplacer les polices existantes, soit ajouter LilyJAZZ et commenter la ligne existante, ce qui permet de changer de police facilement. Dans ce cas il faudra commenter la ligne inutilisée avec un point-virgule et non un signe de pourcentage, puisque nous sommes à l'intérieur de code *scheme*.

```
TailleGlobale = #20
```

```
#{define fonts (make-pango-font-tree  
  ; "Liberation Sans" "Liberation Sans" "Liberation Mono"  
  " LilyJAZZ Text" " LilyJAZZ Text" " LilyJAZZ Text"  
(/ TailleGlobale 20))}
```

Il ne vous reste plus qu'à graver votre partition.

## COMMENT ÇA MARCHE ?

Pour ceux qui aiment comprendre comment fonctionnent les choses, mais aussi pour ceux qui auraient envie de lire le code et éventuellement l'améliorer (il y a certainement beaucoup à faire !).

## LE PRINCIPE

C'est simple, la musique est tout simplement affichée deux fois, mais sur deux portées différentes :

- la première est une portée normale, hormis le fait que l'affichage des numéros de cordes et de doigtés ont été désactivés, ces indications étant détournées de leur usage normal pour enregistrer les notes alternatives et autres indications ;
- la seconde est une portée de percussion (plus adaptée à l'affichage qui nous intéresse) à une seule ligne, où on n'affiche pas les notes mais uniquement un *markup* généré à la volée par une fonction *scheme*. Et ce *markup* contient le numéro de trou et autres indications utiles au jeu.

La fonction *scheme harpoTab* effectue les tâches suivantes.

- Transformation de la hauteur de note en nombre de demi-tons à partir du do central. Ceci a pour avantage de permettre de traiter un  $\text{si}_b$  et un  $\text{la}_\sharp$  de la même façon, car si musicalement ce ne sont pas les mêmes notes, le son en est le même et il faudra donc utiliser le même trou pour les jouer.
- Récupération et stockage des valeurs des articulations attachées à la note.
- Transformation de la tonalité dans laquelle est écrite la musique et décalage de chaque note de cette valeur, ce qui revient à faire une transposition en do. Ceci permet de n'avoir qu'un seul accordage, quelle que soit la tonalité.
- Si on saisit la musique en mode tablature, on analyse la note différemment (voir ci-dessous), afin de construire la valeur de demi-ton. Ensuite on la traite comme si c'était une note normale.
- Récupération des valeurs de trous dans le dictionnaire. Elles peuvent aller jusqu'à trois. Selon qu'une indication de note alternative a été détectée ou non on

classe les valeurs entre note principale et alternative(s).

- Extraction des informations à partir des valeurs de trous. Chaque valeur est écrite sous forme de nombre réel signé. Le signe représente le sens de jeu (-1 aspiré et +1 soufflé). La partie entière représente le numéro de trou, et la partie décimale le code de l'altération éventuelle.
- En notation inversée (mode tablature) le dictionnaire utilisé sera le même, le langage *scheme* ayant la bonne idée de ne pas confondre un nombre entier et en nombre réel. Il n'y aura pas de confusion entre -3 (nombre entier, donc demi-ton indiquant la note) et -3,0 (nombre réel indiquant un trou et autres indications de jeu).
- Génération du *markup* avec ces informations et toutes les indications de configuration. Probablement la partie du code la plus délicate à mettre au point.
- Pour finir on attache le markup à la note.

## LE MODE TABLATURE

Les pseudo-notes i et o ont les particularités suivantes :

- elles sont sur la huitième octave, il faudra donc ajuster cette valeur ;
- elles sont doublement altérées, la valeur de leur altération sera de -1 pour les notes aspirées et +1 pour les notes soufflées, ce qui est bien pratique, c'est la codification retenue.

Ces deux particularités les rend impossibles dans une partition, il n'y aura pas de confusion possible.

Elles sont par ailleurs affectées de plusieurs indications supplémentaires :

- un numéro de doigté, correspondant au trou ;
- éventuellement des indications d'altérations sous forme d'articulations ;
- éventuellement une indication de note alternative, qui sera traitée comme dans une partition normale (cette indication permet de modifier *a posteriori* une tablature pour l'adapter à son style de jeu).

À partir du sens, du numéro de trou et des altération, on peut facilement fabriquer un réel signé, puis chercher dans le dictionnaire d'accords à quel demi-ton il correspond.

Il reste à remplacer la hauteur de la note par celle qui correspond au demi-ton, ce qui écrira une partition en do. Elle sera ensuite automatiquement transposée en fonction des valeurs de *songTone* et *harmoTone*.

# DÉCOUVRIR LILYPOND AVEC HARPO TAB

## ÉCRITURE DE TABLATURES

Les explications qui suivent sont également valables si vous écrivez directement votre tablature en **tabmode**.

## WYSIWYG

LILYPOND n'est pas du tout *Wysiwyg*, mais ne nous leurons pas, ce terme est une pure arnaque, puisque le rendu final dépendra toujours de plusieurs paramètres externes dont notamment l'imprimante.

## AUTRES CHAMPS

Il existe d'autres champs d'en-tête possibles mais qui n'ont pas d'intérêt dans le cadre d'une musique pour harmonica.

## NOMS DE VARIABLES

Il n'est pas possible d'utiliser de chiffres dans les noms de variables. Ce qui n'est pas très pratique, mais c'est comme ça...

Si vous connaissez (ne serait-ce qu'un peu) le solfège mais absolument pas LILYPOND cette section peut vous aider à démarrer, il vous suffit juste d'utiliser le fichier *empty-harpoTab.ly* prévu à cet effet. Les explications qui suivent sont volontairement simplistes et seront avantageusement complétées par la lecture de la documentation officielle, très complète, que l'on trouvera ici : <http://www.lilypond.org/index.fr.html>.

## UTILISATION DE LILYPOND

Contrairement à ce qu'on est accoutumé de voir, LILYPOND est un logiciel sans interface graphique. Il s'agit d'un programme qui lit un simple fichier texte contenant la description de votre musique, et le transforme en PDF et éventuellement en Midi. Cette action s'effectue en ligne de commande, ce qui nécessite de maîtriser un tant soit peu le système d'exploitation. L'alternative est d'utiliser un des éditeurs de texte dédiés à LILYPOND, qui vous proposera non seulement des aides à la saisie de la musique, mais également de compiler vos fichiers d'un simple clic. Nous considérerons que cet aspect technique est résolu pour la suite des explications.

Pour en savoir plus sur les éditeurs compatibles : <http://www.lilypond.org/easier-editing.fr.html>.

## EN-TÊTE DE MORCEAU

C'est la partie qui se trouve dans le bloc `\header { }`. Elle comprend plusieurs champs relativement simples à comprendre (titre, auteur, etc.), et dont la signification est expliquée dans *empty-harpoTab.ly*. Il suffit de saisir le texte adéquat entre les guillemets en prenant bien garde à ne pas les supprimer.

## EXPRESSIONS MUSICALES ET VARIABLES

Il est très important de retenir que la musique doit être constituée d'une unique expression musicale délimitée par des accolades `{ }`. Ce qui n'interdit pas d'écrire plusieurs sous-expressions musicales, elles-mêmes délimitées par des accolades, puis de les insérer les unes dans les autres. Afin de faciliter ces insertions toutes ces sous-expressions seront affectées à des variables que l'on nommera de façon explicite, comme *ThemeUn* ou *ThemeDeux*. On les utilisera ensuite ainsi :

```
MeLody = {  
  \ ThemeUn  
  \ ThemeDeux  
}
```

## DONNÉES GLOBALES

Certaines données, qui s'appliquent à tout le morceau de musique, donc à toutes les variables qui la composent, sont regroupées dans une autre variable que l'on peut insérer là où c'est nécessaire. C'est le cas des indications de tonalité, de tempo, etc. qui se trouvent dans la variable *Global*.

```
Global = {  
  \clef treble   La clé de sol se nomme treble, la clé de fa bass.  
  \key do \major La tonalité de do majeur, \minor pour une tonalité en mineur.  
  \time 4/4      La métrique.  
  \tempo 4=120   Le tempo, avec indication de la note.  
}
```

On insérera ensuite ces informations de la façon suivante dans la variable *Melody* :

```
MeLody = \relative sol' {  
  \Global  
}
```

On saisira la musique à la suite de `\Global`, soit directement, soit en insérant d'autres variables, ou les deux.

## ÉCRITURE DES NOTES

Dans LILYPOND une note est caractérisée par (dans l'ordre) son nom, son altération, son octave et sa durée.

Les noms peuvent s'écrire dans de nombreuses langues, mais toujours en minuscule et sans accents : `re` mais jamais `ré`. La notation par défaut est la notation batave (`a b c d e f g`) qui a l'avantage de ne demander qu'une seule frappe excepté pour les altérations (`is` pour les dièses et `es` pour les bémols). Pour écrire en français (`do re mi fa sol la si`) il faut ajouter l'instruction `\language "italiano"` en début de fichier. Elle s'y trouve par défaut dans `harpoTab.ly`, mais on peut la supprimer ou la transformer en commentaire.

L'altération en français se note simplement `d` pour dièse et `b` pour bémol. En anglais c'est respectivement `s` et `f` (*sharp* et *flat*). En langue batave, la langue par défaut de LILYPOND, ce sera `is` et `es`.

L'octave peut se noter de deux façons, absolue ou relative. Le fichier d'exemple utilise la notation relative, à savoir que les notes sont *centrées* sur celle qui suit la mention `\relative` (généralement la note du milieu de la rangée extérieure).

En mode relatif, LILYPOND considérera qu'en l'absence d'indication explicite il doit utiliser l'écart le plus faible entre deux notes qui se suivent. Donc si vous écrivez `so1 do` le `do` sera celui situé au-dessus du `sol` (écart d'une quarte alors que le `do` du dessous est à une quinte) et dans `so1 mi` le `mi` sera en dessous. On peut forcer le passage à une autre octave en écrivant `sol do`, (avec une virgule accolée au nom de la note) pour forcer l'écriture de la note en dessous, et `sol mi'` (avec une apostrophe) pour afficher une note située plus haut.

La durée se note avec un nombre égal à 1 pour la ronde, 2 pour la +blanche, 4 pour la noire, 8 pour la croche... Pour une noire pointée on écrit tout simplement 4. (un 4 suivi d'un point). On s'y fait très vite également.

Toutes ces indications sont écrites dans l'ordre donné ci-dessus et sans espace : `fad'4 so1` se lit : `fa` dièse qu'on force à monter au-dessus de la note précédente, durée une noire, suivi d'un `sol` qui dure lui aussi une noire (persistance de la durée).

Les silences se notent `r` (comme *rest* en anglais) et leur durée s'indique de la même façon que les notes. On peut également écrire des silences invisibles avec un `s`.

## ÉCRITURE DE LA MUSIQUE

On l'a vu, celle-ci se trouve dans ce qu'on appelle une variable, et qui se présente ainsi :

```
Melody = relative sol' { ... }
```

Le `so1'` qui suit le mot clé `relative` indique la note centrale de la musique (ici en fait la note centrale de l'harmonica, ce qui est plus simple). Dans cette variable on écrira la musique comme indiqué précédemment, en respectant les conventions suivantes :

- une mesure par ligne ;
- un `|` à la fin de chaque mesure (il n'est pas obligatoire mais permet de repérer les erreurs de durées de notes, de mesures incomplètes...)

On ajoutera aux indications purement musicales les indications nécessaires à la création de la tablature, comme indiqué plus haut :

```
Melody = relative sol'' {  
  la2 do4 | la blanche, aspiré, rangée 1, doigt 2 ; do noire, doigt 4  
  si2 so14-1 | si blanche, poussé, doigt 2 ; sol noire, rangée 1, doigt 1  
  la2. | la blanche pointée, tiré, rangée 1, doigt 2  
}
```

### COMMENTAIRES

Pour que du texte ne soit pas considéré comme du code on le formate en tant que commentaire en faisant précéder celui-ci d'un caractère dévolu à cet effet. Chaque langage a le sien, et dans LILYPOND c'est le `%` qui a cet effet.

### ACCOUTUMANCE

La notation relative peut paraître un peu déroutante au début mais on l'apprécie très vite.

### PRATIQUE !

Il n'est pas nécessaire de répéter l'indication de durée tant qu'elle reste identique.

### CONTRÔLES

Le `|` de fin de mesure n'est pas obligatoire mais permet à LILYPOND de signaler une éventuelle erreur dans les barres de mesure, à savoir une ou plusieurs notes dont la durée n'est pas correcte.

### COPIER/COLLER

Il est souvent tentant de faire des copies de mesures d'un endroit à l'autre de la mélodie.

Dans ce cas attention aux absences d'indication de durée qui provoqueront inmanquablement des erreurs.

Lesquelles seront facilement évitées si la première note de la mesure a une indication de durée explicite, et identifiées si les `|` précédents ont été placés en fin de mesure.



### UTILISATION

Le code donné en exemple ci-contre sera inséré (autant de fois que nécessaire) entre les accolades ouvrantes et fermantes de la variable *Melody* (ou une de ses sous-variables si nécessaire).

### ACCOLADES

Elles s'utilisent un peu à la façon des parenthèses dans une expression mathématique.

Il est important d'être rigoureux dans leur écriture sous peine d'erreurs. C'est donc une bonne habitude d'indenter (ajouter des espaces en début de ligne) à chaque fois que l'on passe à une nouvelle imbrication, et de les retirer lorsque l'on ferme un niveau d'imbrication, comme dans l'exemple ci-contre.

## LES RÉPÉTITIONS

Certains morceaux sont conçus comme un ensemble d'une ou plusieurs phrases musicales répétées, parfois avec des fins différentes. On écrira ceci sous forme de plusieurs expressions musicales imbriquées :

```
\repeat volta 4 {
  la4\i do8 la |
  si4\o si8 do |
}
\alternative {
  { re8\o mi re si |
    do4\i la | }
  { re8\o mi re do |
    la4\i la | }
  { re8\o mi re si |
    do4\i la | }
  { re8\o mi re do |
    la2\i | }
}
```

La partie commune et le nombre de reprises de celle-ci. Cette partie est une expression musicale à part entière et est donc placée entre accolades

Fin de la partie commune

Fins alternatives

La première fin, sur deux mesures donc deux lignes, est aussi une expression musicale donc entre accolades.

La deuxième fin....

Etc.

Fin des fins alternatives

## LES ANACROUSES

Elles sont aussi appelées levées, et sont notées ainsi :

```
\partial 4. { do16 re mi }
```

La seule différence avec les autres expressions musicales (hormis le mot-clé `partial`) est qu'il faut indiquer la durée de l'expression musicale contenant l'anacrouse avant d'écrire celle-ci.

## LES ACCORDS

Contrairement aux notes, les accords sont écrits dans un mode spécifique, nommé `\chordmode`, et qui permet d'ajouter au nom de l'accord des précisions telles que mineur, septième ou augmenté. Reportez-vous au manuel de LILYPOND pour trouver les syntaxes correctes.

## AUTRES VARIABLES ET SCORE

Ces variables et la façon de les utiliser a été largement décrite plus haut, il n'est pas utile d'y revenir. Ne touchez pas au reste du code et tout marchera parfaitement.

## LA COMPILATION

Selon l'éditeur que vous utiliserez, la procédure sera différente, donc non expliquée ici. Si vous lancez l'édition en ligne de commande, tapez simplement :

```
lilypond monfichier.ly
```

et les fichiers PDF et Midi seront générés dans le répertoire courant.

### ALLER PLUS LOIN

Si vous souhaitez approfondir vos connaissances, le site de LILYPOND regorge d'explications et d'exemples.

L'étude du fichier *empty-harpoTab.ly* peut constituer un bon départ pour étudier le fonctionnement précis de ce fantastique programme.