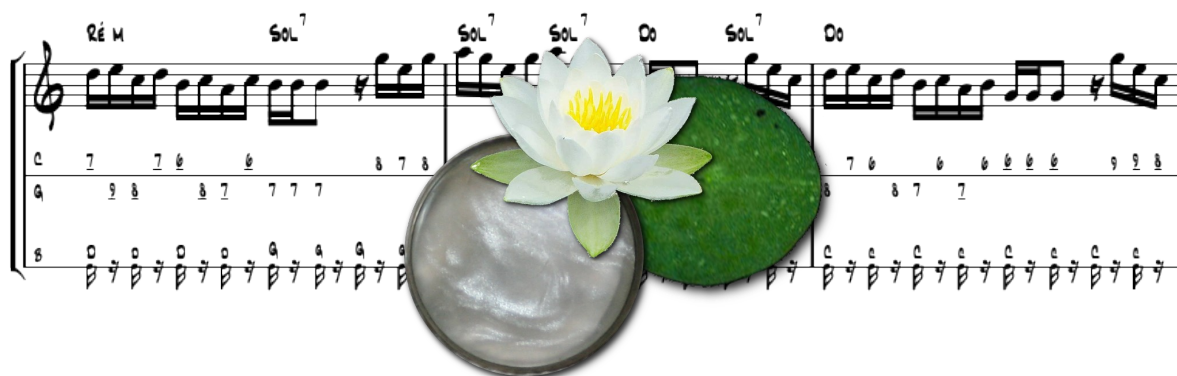


FISARMONICA

Gravez vos tablatures d'accordéon diatonique à partir d'un fichier LILYPOND



FONCTIONNALITÉS

PRÉ-REQUIS

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

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 boutons/rangée de clavier).
- Support de la quasi-totalité des configurations de claviers à une, deux ou trois rangées.
- Support des systèmes de tablature *Cadb* et *Corgeron*, ainsi que du système de l'auteur.
- Possibilité de saisir une musique dans n'importe quelle tonalité et obtenir une tablature dans n'importe quelle autre tonalité.
- Accessibilité : amélioration de la lisibilité avec la possibilité d'obtenir des partitions et tablatures en couleur (personnalisables) en fonction du sens de jeu, ou d'afficher les notes tirées encadrées au lieu du soulignement habituel.
- Toutes les clés de tablatures sont calculées automatiquement mais de plus elles sont individuellement personnalisables.
- Indication des doigtés sur une ligne séparée de la partition.
- Mode *brouillon* permettant de voir et tester toutes les possibilités afin d'affiner progressivement l'écriture.
- Écriture facilitée de la ligne de basse grâce à des motifs rythmiques pré-définis.
- Sortie *Midi* réaliste grâce aux mêmes motifs rythmiques.
- Utilitaire de création de nouvelles configurations de clavier.
- Pack de personnalisation de l'éditeur **Kile** permettant la saisie et compilation en mode graphique.
- 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
Installation et utilisation.....	3
Considérations algorithmiques.....	3
Configuration de Fisarmonica.....	4
Configurations de clavier.....	4
Déclaration de configuration.....	5
Nom d'instrument.....	5
Dictionnaires et solfège.....	5
Tonalité.....	6
Système de tablatures.....	7
Système Corgeron.....	7
Système Cadb.....	7
Système Cartron.....	7
Clé de basses et de doigtés.....	7
Accessibilité.....	7
Style de jeu.....	8
En-tête de fichier.....	8
Écriture de la musique.....	8
Règles générales.....	8
Écriture de la mélodie.....	8
Indication de sens.....	9
Indication de rangée.....	9
Indication de voix.....	9
Note inexistante.....	9
Indications de doigté.....	9
Exemple.....	9
Rendu graphique de la mélodie.....	10
Rendu graphique de la tablature.....	10
Rendu graphique des doigtés.....	10
Écriture directe de tablature.....	10
Utilisation du langage fisarmonica.....	11
Écriture des basses - accords.....	12
Indication de tierce.....	13
Motifs rythmiques.....	13
Écriture manuelle de la tablature de basses.....	15
Affichage des noms d'accords au-dessus de la portée.....	15
Sortie midi.....	16
Historique, remerciements et dédicace.....	17
Annexes.....	18
A)Licence.....	18
B)Réflexions sur les plans de clavier.....	18
Gammes aberrantes.....	18
Altérations supplémentaires.....	18
Sol/Do.....	19
Autre tonalités.....	19
Conclusion.....	19
C)Nouvelle configuration de clavier.....	19
D)Nouveau motif rythmique.....	21
E)Nouveau motif midi.....	21
F)Modification des diagrammes de voix.....	22
G)EmptyTab.ly.....	22
H)Découvrir Lilypond avec Fisarmonica.....	23
Utilisation de Lilypond.....	23
En-tête de morceau.....	23
Expressions musicales et variables.....	23
Données globales.....	23
Écriture des notes.....	24
Écriture de la musique.....	24
Les répétitions.....	25
Les anacrouses.....	25
Les basses.....	25
Les accords.....	25
Autres variables et Score.....	25
La compilation.....	25

INSTALLATION ET UTILISATION

FISARMONICA est composé des fichiers suivants :

INSTALLATION

Ces trois 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.

- *fisarmonica-init.ly* est le fichier d'initialisation qui contient toutes les configurations de clavier, ainsi que la configuration des systèmes de tablature. Hormis pour ajouter une nouvelle configuration de clavier, il n'est pas supposé être modifié.
- *fisarmonica-setup.ly* contient toutes les indications de configuration, toutes les valeurs par défaut, et les motifs rythmiques pour les basses. Il est prévu pour enregistrer toutes vos préférences, ainsi que les tableaux de traduction des noms d'instrument ou de notes. Il peut aussi être utilisé pour enregistrer d'autres préférences comme par exemple le fait que les indications de parties sont présentées sous forme de lettre encadrée, ce qui est d'ailleurs le cas dans le fichier fourni.
- *fisarmonica.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 !*

De plus les fichiers suivants, non indispensables mais probablement utiles, sont également fournis :

- *keyboard-layout.ods*, classeur utilisable avec LibreOffice ou autre logiciel compatible, est un petit utilitaire permettant de générer simplement de nouvelles configurations de clavier si nécessaire ;
- *EmptyTab.ly* est un modèle de fichier que l'on peut recopier pour écrire ses partitions, il est préformaté pour l'utilisation de **FISARMONICA**, et existe également en version optimisée pour la saisie directe de tablatures ;
- *fisarmonica.ttf* est une police de caractères contenant les symboles des boutons des trois rangées de clavier permettant la saisie de tablatures directement avec les numéros de boutons au lieu de notes ;
- afin de permettre une saisie facile avec le langage intégré *fisarmonica*, un pack d'icônes est également fourni, ainsi qu'un petit script d'installation dans l'éditeur **Kile**.

CONSIDÉRATIONS ALGORITHMIQUES

CONVENTIONS

Pour toutes les explications qui vont suivre nous nous baserons sur un modèle dit « Viennois » en Sol/Do, le plus répandu en France.

Par ailleurs, les noms de note seront écrits en français (langage *italiano* dans LILYPOND)

Mettre au point l'algorithme qui générera automatiquement la tablature idéale à partir d'une partition relève de l'utopie.

En effet, bien que la grille d'accord d'un morceau soit une contrainte assez forte dans le choix du sens de jeu, lequel permet de déterminer quel bouton sur quelle rangée devra être utilisé, il reste un bon nombre de paramètres flous :

- Il existe d'une part cinq notes (trois sol et deux la) qui peuvent se jouer indifféremment sur les deux rangées mais dans un seul sens respectivement. Écrire un algorithme qui saurait quelle rangée utiliser suppose que l'on puisse déterminer quels enchaînements de doigtés sont ergonomiques ou non, rapides ou non, et encore cela dépendrait-il de quel doigt on démarre la séquence, donc des notes précédentes et suivantes, de l'agilité du musicien, etc. On sera donc obligé de faire un choix manuel, après avoir testé les différentes possibilités.
- Il existe d'autre part deux basses (sol et fa) qui peuvent se jouer dans les deux sens (mais pas au même endroit, ce qui a moins d'importance, les indications de boutons et rangées des basses étant généralement absentes des tablatures). On pourrait ici imaginer d'analyser les notes qui composent la mesure, ce qui permettrait de déterminer si on doit pousser ou tirer, et mettre ceci en relation avec la quantité d'air disponible dans le soufflet, calculée en fonction du sens de jeu des mesures précédentes la quantité d'air disponible dans le soufflet, etc. Il est beaucoup plus logique (et facile...) de laisser le musicien

choisir s'il veut tirer ou pousser.

- Enfin, même en choisissant de pousser (ou tirer) une basse qui permet les deux sens, il est tout à fait possible d'imaginer que sur les contre-temps (où la basse n'est pas audible) il soit plus ergonomique (ou plus musical, plus dynamique, etc.) d'inverser le sens du soufflet brièvement.

Pour l'instant aucun de ces algorithmes complexes n'est disponible dans **FISARMONICA**, donc il vous faudra faire ces choix manuellement.

Dans le cas contraire, **FISARMONICA** affichera tout simplement tous les boutons possibles pour la ou les notes concernées, ce qui alourdira la tablature, particulièrement pour le système *Cadb*, puisqu'il n'est pas conçu pour afficher simultanément tous les claviers. Pour le système *Corgeron*, l'affichage d'un bouton de la troisième rangée étant prévu pour être rehaussé verticalement, ce décalage restera présent y compris si ce bouton est juxtaposé avec un bouton de la deuxième rangée.

Malgré ces quelques défauts esthétiques, la tablature restera lisible, et ceci peut vous permettre de tester les différentes possibilités, puis de corriger progressivement votre partition jusqu'à obtenir un résultat satisfaisant.

CONFIGURATION DE FISARMONICA

L'utilisation de **FISARMONICA** impose la saisie de quelques déclarations de configuration supplémentaires à celles habituelles de **LILYPOND**.

Ceci peut se faire de deux façons, l'une globale, écrite dans un fichier de configuration séparé nommé *fisarmonica-setup.ly*, ou de façon locale dans chaque fichier de partition. Le comportement de **FISARMONICA** est de lire d'abord le fichier de configuration, puis les indications locales, lesquelles auront priorité sur la configuration globale.

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

CONFIGURATIONS DE CLAVIER

La famille des accordéons diatoniques comporte de nombreux modèles, certains courants, d'autres beaucoup plus confidentiels, voire totalement personnalisés. **FISARMONICA** essaye de vous proposer en standard les plus répandus. Les fabricants connus pour proposer ces configurations sont indiqués par une abréviation de deux lettres : **Aa**.

INDÉCIS ?
Le détail des configurations de clavier existantes se trouve en Annexe X.

ABRÉVIATIONS POUR LES NOMS DE FABRICANTS :
Ca = Castagnari - **BL** = Bernard Loffet - **Ho** = Hohner

MODÈLE	DESCRIPTION	CODE
MÉLODÉONS		
Modèle standard	Une seule rangée de 10 boutons sans aucune altération	me1
Modèle <i>Bouchard</i>	Une rangée plus 4 boutons uni-sonores	bou
Modèle <i>Sagné</i>	Une rangée plus 5 boutons sur la deuxième rangée	sag
DIATONIQUES À DEUX RANGÉES - MODÈLE VIENNOIS - CONFIGURATION DITE « CONTINENTALE » Deuxième rangée décalée d'une quarte vers l'aigu		
Modèle standard	Deux rangées sans altérations supplémentaires	ct1
Modèle standard amélioré	Altérations supplémentaires sur les boutons 1 de chaque rangée	cte
Modèle basque <i>Trikitixa</i>	Deux rangées avec 12 et 11 boutons, basses uni-sonores	tkx
DIATONIQUES À DEUX RANGÉES - MODÈLE IRLANDAIS Deuxième rangée décalée d'un demi-ton		
Modèle standard	Deuxième rangée décalée d'un demi-ton vers l'aigu	eir
Modèle inversé	Deuxième rangée décalée d'un demi-ton vers le grave	eiv

MODÈLE	DESCRIPTION	CODE
DIATONIQUE À DEUX RANGÉES PLUS DEUX BOUTONS POUR LES ALTÉRATIONS SUPPLÉMENTAIRES		
On considérera que dans ce cas les altérations disparaissent des deux premières rangées, vu le peu d'intérêt qu'elles y présentent.		
Modèle standard	Le bouton 1 de la rangée 1 est transféré sur le bouton 1 de la rangée 3 et le bouton 1 de la rangée 2 sur le bouton 2 de la rangée 3 (Ca)	cx5
Modèle inversé	La position des boutons transférés est inversée (BL)	cx1
DIATONIQUE À DEUX RANGÉES ET DEMI		
Deux rangées « continentales » plus altérations et notes en sens inversé sur les boutons de la troisième rangée.		
Modèle « Leray »	Deux rangées plus 5 boutons (Ca)	j12
Continental + 5 boutons	Variante du précédent (BL)	cx5
Continental + 6 boutons	Deux altérations supplémentaires (BL)	cx6
		cxt
DIATONIQUE À TROIS RANGÉES		
Modèle continental	Trois rangées décalées d'une quinte, plus quelques altérations.	clt
Modèle « Corgeron »	Deux rangées décalées d'une quinte, avec des altérations supplémentaires et des notes avec un sens de jeu inversé sur la troisième. Les modèles portent le nom de leur concepteur.	jmc
Modèle « Heim »		fhs
Modèle « Leray »		j13
<i>British chromatic</i>	Trois rangées décalées d'un demi-ton ascendant, quelques altérations.	bch

ERREURS ET IMPRÉCISIONS

Les configurations de clavier et les noms de fabricants les commercialisant sont fatalement incomplètes voire erronées. Toute information permettant d'affiner ces données sera la bienvenue, particulièrement en ce qui concerne les configurations de clavier manquantes.

CONFIGURATION MANQUANTE

Si la configuration de votre clavier n'est pas implémentée, vous pouvez très simplement l'ajouter en suivant les explications données en *Annexe C*.

DÉCLARATION DE CONFIGURATION

`#(define kbdOut code)` où `code` correspond au modèle de votre accordéon.

TRADUCTION

Tous les termes apparaissant dans une tablature et pouvant nécessiter une traduction sont définis dans une liste associative suffixée définie dans le fichier de configuration. Il suffit de déclarer la liste par défaut avec le nom suffixé comme expliqué ci-contre pour que les termes soient automatiquement traduits.

NOM D'INSTRUMENT

Les noms d'instrument sont par défaut en français, mais on peut modifier l'instruction `#(define kbd-name kbd-name_FR)` pour sélectionner une autre liste de noms, à créer au préalable bien entendu. 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.

DICTIONNAIRES ET SOLFÈGE

Les plans de clavier fournis par les fabricants sont bien souvent fantaisistes au niveau de la notation des altérations : rangée de clavier en `do #` avec un `fa #` et un `fa` mais pas de `mi #` , et ne parlons pas des altérations supplémentaires notées sans aucun respect de la théorie musicale. Les dictionnaires de **FISARMONICA** ne peuvent tenir compte de toutes ces erreurs et ne contiennent que les notes telles qu'elles doivent être écrites. Pour plus de détails voir *Annexe B*.

TONALITÉ

Il n'existe pas de règle d'auto-transposition pour le diatonique, mais étant donné qu'écrire un dictionnaire « note ⇔ bouton » pour chaque configuration de clavier et pour chaque tonalité représenterait un travail énorme et par surcroît totalement inutile, une seule tonalité a été retenue pour la rangée extérieure de boutons, à savoir le Do, les rangées intérieures étant par conséquent en Fa pour les configurations continentales, et en Do # ou Si pour le modèle irlandais.

Il est cependant possible d'écrire une partition dans n'importe quelle autre tonalité, LILYPOND transposera automatiquement votre partition dans la tonalité du dictionnaire. Pour que ceci soit possible il est impératif d'indiquer la tonalité dans laquelle vous saisissez la partition. En réalité il faudrait dire : tonalité de l'accordéon pour lequel on écrit la partition. Seule la valeur de la rangée extérieure est nécessaire, la tonalité de la rangée intérieure sera déduite de l'indication de la configuration du clavier.

Pour un accordéon diatonique en Sol/Do, on indiquera `scoreIn = g` OU `scoreIn = sol` si on a choisi d'utiliser le langage *italiano*.

Cette indication ne se trouve pas dans le fichier *fisarmonica-setup.ly*, car elle est par essence propre à chaque partition. Elle est par conséquent absolument indispensable, et il est tout autant absolument indispensable de ne plus la modifier une fois la partition écrite.

De même, afin que F~~ISAR~~MONICA puisse calculer une tablature adaptée à votre instrument, il faudra lui indiquer sa tonalité ainsi : `scoreOut = sol`, ce qui est la valeur par défaut.

Il est important de noter que si on utilise un autre langage que le langage standard (anglais) il est indispensable que les indications du fichier de configuration soient cohérentes avec la déclaration de langage. Le fichier fourni est en langage standard, et le choix du langage doit donc être fait après son inclusion. Si vous préférez écrire votre fichier de configuration dans un autre langage son inclusion devra logiquement être placée avant celle de ce fichier.

En d'autres termes, `scoreIn` correspond à la tonalité de la musique que vous écrivez, et `scoreOut` à celle que vous voulez lire sur votre partition et votre tablature.

Ainsi, vous pouvez par exemple recopier telle quelle une partition écrite pour un diatonique en Si^b/Mi^b (`scoreIn = sib`), celle-ci sera transposée en Do/Fa pour la génération de la tablature, et la mélodie et les basses le seront en Ré/Sol (`scoreOut = re`) par exemple si c'est la tonalité de votre instrument. Mais si votre voisin joue sur un La/Ré, écrivez `scoreOut = la` et vous pourrez instantanément lui offrir partition et tablature dans sa tonalité.

Prenez toutefois garde que LILYPOND transpose en utilisant l'écart le plus faible possible, donc pour passer de do à sol, au lieu de monter d'une quinte, la partie transposée descendra d'une quarte, ce qui veut dire éventuellement une sortie des limites de l'instrument. Dans ce cas, il faudra indiquer l'une ou l'autre des tonalités avec une apostrophe (') pour indiquer l'octave supérieure, ou une virgule (,) pour indiquer l'octave inférieure. Quelques essais résoudre tous les problèmes.

Si vous écrivez votre partition directement dans la tonalité du dictionnaire (Do) et que c'est également celle de votre instrument, ces deux indications de tonalité peuvent vous sembler superflues voire totalement inutiles, mais pensez que si vous partagez votre partition avec d'autres instrumentistes ils pourront être heureux de bénéficier d'une tablature dans la tonalité de leur instrument uniquement en changeant la valeur de `scoreOut` ce qui générera automatiquement toutes les transpositions nécessaires.

D'autre part si vous trouvez une partition ou tablature écrite pour un instrument d'une autre tonalité, là encore toutes les transpositions seront automatiques, il vous suffira de recopier la partition telle quelle.

INSTRUMENT

AUTO

TRANSPOSITEUR

L'accordéon diatonique est un instrument auto-transpositeur, c'est à dire que quelque soit sa tonalité le doigté sera identique pour jouer un même morceau.

Pour de tels instruments, on écrit les partitions non dans la tonalité de chacun d'entre eux, mais dans une tonalité virtuelle, en clé d'ut, et chaque musicien lit en fait non les notes réelles mais les notes correspondant au doigté.

Bien que l'accordéon diatonique fasse partie de cette famille d'instruments, aucune règle n'a été définie en ce qui le concerne, nous avons donc toute liberté pour utiliser la nôtre.

ÉCRITURE

Les définitions 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 `corg` ou `cte`.

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 définition.

SYSTÈME DE TABLATURES

Les deux systèmes de tablature communément répandus sont *Cadb* (`cadb`) et *Corgeron* (`corg`). La valeur par défaut est : `#(define syst corg)`.

SYSTÈME CORGERON

MÉLODÉONS

Les utilisateurs de mélodéons à une rangée préféreront certainement une tablature *Cadb*, il n'y a donc pas de tablature *Corgeron* à une seule rangée.

Le nombre de rangées à afficher sera déduit automatiquement de la configuration de clavier qui aura été déclarée.

La clé de tablature est habituellement la tonalité de chaque rangée en notation anglaise. Celle-ci est automatiquement calculée en fonction de la valeur de `scoreOut` et de la configuration de clavier. Il est cependant possible de choisir une autre langue en modifiant l'instruction `#(define clefNames noteNames_EN)` et en indiquant par exemple `noteNames_IT`. Il est également possible d'opter pour le nom complet (`so1` et non `s`) en adaptant l'instruction `#(define clefLongName #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 *EmptyTab.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.

On peut enfin préférer une indication numérique des rangées, ce qui s'obtient en adaptant l'instruction : `#(define clefNum #f)`.

SYSTÈME CADB

La clé de tablature est habituellement constituée des lettres **P T**, mais on peut également les personnaliser dans les instructions suivantes :

- `#(define diatoTabCacbUpperClef "P")` pour la ligne supérieure ;
- `#(define diatoTabCacbUpperClef "T")` pour la ligne inférieure.

SYSTÈME CARTRON

Ce système reprend les caractéristiques les plus intéressantes de chacun des deux autres systèmes :

- les notes poussées sont écrites au-dessus de la ligne, les notes tirées en-dessous ;
- chaque rangée de clavier a sa propre ligne.

Ceci donne une tablature un peu plus volumineuse que les autres systèmes, mais plus claire et plus lisible (plus de soulignements ni de `'`). On la notera : `#(define syst cart)`.

UNIVERSALITÉ

Les clés de basses et de doigté sont les mêmes quel que soit le système choisi.

CLÉ DE BASSES ET DE DOIGTÉS

Ces clés peuvent être personnalisées en modifiant les valeurs contenues respectivement dans les instructions `#(define diatoTabBassClef "B")` et `#(define diatoTabFingeringClef "")`.

ACCESSIBILITÉ

Afin d'améliorer la lisibilité de la tablature, trois options inhabituelles sont proposées mais désactivées par défaut :

- **Partition et/ou tablature colorées** : avec ces options, les notes et/ou numéros de bouton sont écrits en bleu pour le jeu poussé et en rouge pour le jeu tiré. Les instructions correspondantes sont respectivement `#(define colorscore #f)` et `#(define colortab #f)`. Les couleurs, nommées respectivement `pushcolor` et `pullcolor`, sont définies le fichier de configuration au format **RGB** (soit trois valeurs comprises entre 0 et 1) et sont personnalisables dans ce fichier uniquement.
- **Indications de notes tirées encadrées** : au lieu du soulignement habituel du système *Corgeron* on peut préférer un encadrement complet du numéro de bouton. L'option par défaut est : `#(define boxmode #f)`.

CUMUL D'OPTIONS

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

ENCADREMENT

Cette option n'a d'effet que si le système *Corgeron* a été sélectionné.

STYLE DE JEU

Par défaut **FISARMONICA** considérera que vous jouez en croisé, sur toutes les rangées de votre instrument (sauf évidemment si vous avez déclaré une configuration de clavier du type mélodéon). Cette valeur par défaut est notée `#(define row 0)`, ce qui signifie qu'aucune rangée de clavier (*row* en anglais) n'est imposée.

Si vous préférez que la tablature soit écrite uniquement en poussé/tiré, il vous suffit d'indiquer sur quelle rangée le morceau doit être joué de la façon suivante :

- on impose l'utilisation de la première rangée : `#(define row 1)` ;
- on impose l'utilisation de la deuxième rangée : `#(define row 2)`.

EN-TÊTE DE FICHIER

En conséquence de tout ce qui vient d'être dit, un fichier `.ly` devrait comporter les lignes suivantes en en-tête :

```
\include "fisarmonica-init.ly"
\include "fisarmonica-setup.ly"
\include "fisarmonica.ly"
% #(define kbdOut ctx)
% #(define syst corg)
scoreIn = sol
% scoreOut = sol
```

L'inclusion du fichier de configuration `fisarmonica-setup.ly` se fait préalablement à toutes les déclarations locales, et `fisarmonica.ly` est chargé à la fin.

Évidemment on peut préférer saisir ses partitions en *italiano* et dans ce cas la ligne `\language "italiano"` devra être insérée avant ou après `\include "fisarmonica-setup.ly"` selon que celui-ci est écrit ou non dans ce langage.

ÉCRITURE DE LA MUSIQUE

RÈGLES GÉNÉRALES

Autant il est indispensable de respecter strictement la syntaxe des indications de configuration pour que **FISARMONICA** puisse générer une tablature répondant à vos besoins, autant les règles de saisie de la musique sont moins strictes (à une exception près pour les basses).

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* ;
- les basses/accords du morceau se trouvent dans une variable nommée *Basses*.

D'autres variables seront utilisées pour la génération de la tablature mais ne font pas partie de l'écriture de la musique à proprement parler.

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.

CHANGEMENT DE STYLE

L'indication de style peut être modifiée à tout moment en cours de morceau, on peut donc passer d'un jeu en poussé/tiré sur la rangée intérieure à un jeu croisé suivi d'un jeu en poussé/tiré sur la rangée extérieure par exemple.

NOTATION

On remarque que les quelques indications de configuration locale qui se trouvent dans cet en-tête sont commentées (précédées d'un `%`), puisqu'on considère que celles qui sont contenues dans le fichier de configuration sont correctes.

Mais il est *fair-play* de laisser ces indications si on pense partager le fichier. Il est en outre important de vérifier que la valeur de `scoreIn` est correcte, car c'est la seule qui est nécessaire pour obtenir une tablature correcte.

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...*

MNÉMONIQUE

Il y a un o dans pousser et un i dans tirer, mais en réalité il faut penser à *in* (l'air rentre) et *out* (l'air sort).

NON SENS

L'absence d'indication de sens peut être forcée en écrivant `\x`, ce qui permet d'annuler la répétition de la valeur précédente, et autorise par conséquent l'affichage des boutons de chaque sens, relativement au système de tablature choisi.

SURCHARGE

L'indication de rangée outrepassera toujours une indication de style. On peut donc définir le style de jeu à `$(define row 1)` et indiquer sur une note `\2`, cette note et seulement celle-ci sera alors affichée sur la rangée 2 (si elle y existe).

ÉCRITURE

Ces notations s'utilisent comme les indications d'interprétation.

Les diagrammes seront affichés par défaut au-dessus de la portée, sauf ceux à une ou deux tirettes qui sont a priori destinés aux basses, et qu'il faudra ici faire précéder d'un `^`.

MNÉMONIQUE

Signification des lettres :
- t pour tirette ;
- o pour ouvert ;
- x pour fermé.

ÉCRITURE

L'ordre des indications de sens, de doigté et de rangée de clavier n'a aucune incidence sur la génération de la tablature.

INDICATION DE SENS

Le sens de jeu est essentiellement dépendant des basses, mais c'est malgré tout dans la mélodie qu'il sera noté, ce qui évite de devoir analyser de façon synchrone celle-ci et les basses.

Cette indication est absolument nécessaire pour obtenir une tablature correcte.

La notation utilisée est `\o` pour un jeu en poussant et `\i` pour un jeu en tirant.

Il n'est pas nécessaire de répéter cette indication pour chaque note, mais uniquement aux changements de sens, quelle que soit la basse jouée.

Les changements ponctuels causés par une impossibilité d'obtenir une note dans le sens indiqué seront gérés par `FISARMONICA`, il est donc inutile de les indiquer.

INDICATION DE RANGÉE

Il n'est pas nécessaire *a priori* d'indiquer sur quelle rangée une note est jouée, sauf pour les quelques notes qui existent sur deux rangées ou plus. On indiquera les rangées de clavier avec `\1` pour la première, celle de l'extérieur, `\2` et `\3` pour les suivantes. Si cette indication est erronée (note impossible sur la rangée indiquée), elle sera tout simplement ignorée.

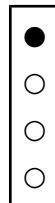
On peut en outre surcharger de cette façon une note qui par exemple devrait être jouée en tiré parce que la basse l'impose mais que l'on veut cependant pousser pour quelque raison que ce soit. Dans ce cas l'indication de rangée outrepassera celle du sens, même si on écrit l'indication de sens après celle de la rangée.

INDICATION DE VOIX

Certains modèles, notamment les mélodéons, possèdent de deux à quatre voix (ou registres) activées par des tirettes. Afin de pouvoir indiquer si celles-ci doivent être tirées ou poussées (les registres ouverts ou fermés) on peut utiliser les notations `\txox`, ou `\toxxo` qui indiquent respectivement que les registres 1 et 3 sont fermés et le 2 ouvert, ou que les 1 et 4 sont ouverts et les 2 et 3 fermés. Toutes les notations de deux à quatre voix sont disponibles.

Les registres ouverts seront représentés par un \circ et les registres fermés par un \bullet . Étant donné qu'il n'y a apparemment pas de norme pour ce genre d'indication chacun pourra décider des règles d'utilisation de ce diagramme (tirette extérieure en haut ou en bas, par exemple).

Quelle que soit la convention adoptée, dans la notation `\txooo` représentée ci-contre, le registre fermé est affiché en haut du diagramme.



NOTE INEXISTANTE

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

INDICATIONS DE DOIGTÉ

Elles se notent de façon classique, avec un `-` suivi du numéro du doigt. On ne peut par contre pas utiliser la notation dite *pimax*, celle-ci étant réservée pour le sens de jeu.

EXEMPLE

Une mesure qui s'écrirait normalement ainsi :

`si4 re do mi`

deviendra, avec toutes les indications supplémentaires :

`si4\1-1\i re-2\2 do-3 mi-4`

On a ajouté pour la première note : rangée 1, doigt 1, sens tiré, et pour la deuxième : doigt 2 , rangée 2 ...

RENDU GRAPHIQUE DE LA MÉLODIE

Afin d'avoir une partition qui ne contienne pas toutes les indications de rangées et de doigté, lesquelles sont inutiles à ce niveau, il suffit d'utiliser le contexte prédéfini nommé `diatoStaffSetup`, de la façon suivante :

```
\new Staff \with { \diatoStaffSetup } << \Melody >>
```

En fait on écrira plus précisément :

```
\new Staff \with { \diatoStaffSetup } << \transpose \scoreIn \scoreOut \Melody >>
```

ce qui permet d'obtenir automatiquement la partition dans la tonalité désirée, comme on l'a vu plus haut.

RENDU GRAPHIQUE DE LA TABLATURE

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

```
\new Staff \with { \diatoTabSetup } << { \tabDiatomelody \Melody } >>
```

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

Les effets de ces modifications sont les suivants :

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

RENDU GRAPHIQUE DES DOIGTÉS

Ceux-ci ne sont pas affichés, comme c'est l'usage, au-dessus ou au-dessous de la portée mais sur une ligne spéciale située entre la portée et la tablature, que vous insérez ainsi :

```
\new Staff \with { \diatoFingeringSetup } << \Melody >>
```

ÉCRITURE DIRECTE DE TABLATURE

Si vous ne connaissez pas suffisamment la musique pour écrire vos partitions mais que vous souhaitez simplement recopier une tablature, ou écrire votre morceau directement sous forme de tablature, **FISARMONICA** vous le permet.

Cette police possède six séries de vingt caractères supplémentaires (seuls les douze premiers, voire moins, seront utiles) permettant de saisir directement les boutons de votre accordéon. Leur codification suit les règles suivantes :

- les nombres en **positif** (noir sur fond blanc) représentent les boutons **poussés** ;
- les nombres en **négatif** (blanc sur fond noir) représentent les boutons en **tiré** ;
- les nombres soulignés représentent la première rangée ;
- les nombres sans soulignement représentent la deuxième rangée :

CONTEXTE

Le contexte `diatoStaffSetup`, ainsi que tous les autres contextes, est défini à la fin du fichier `fisarmonica.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.

EMPTYTAB.LY

Par défaut cette ligne est désactivée (commentée) dans le fichier modèle, toutes les tablatures ne comportant pas les indications de doigté.

PRÉ-REQUIS

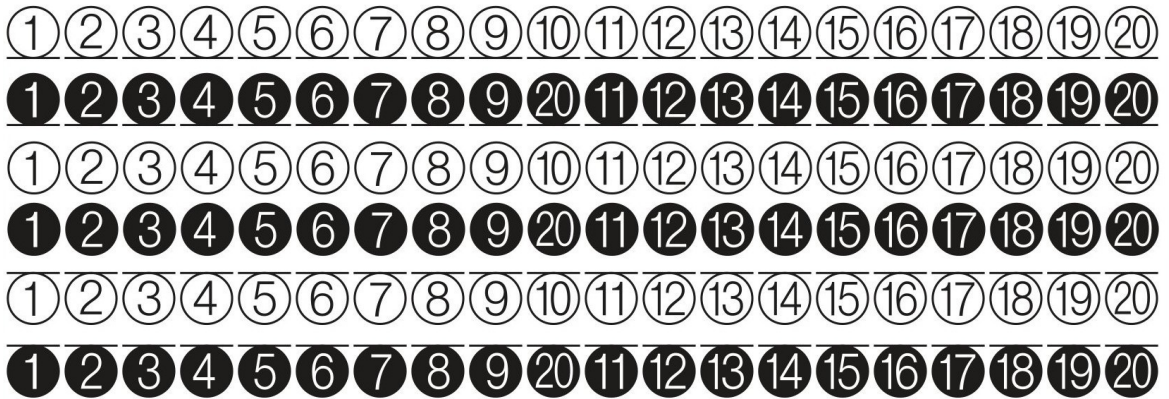
Pour pouvoir écrire directement en mode tablature, vous devrez installer sur votre système la police de caractères `Fisarmonica.ttf`, qui est directement dérivée de la police `FreeSans.ttf`, et choisir d'afficher votre fichier `.ly` avec cette police dans l'éditeur que vous utilisez.

MNÉMONIQUE

Les lettres en gras indiquent comment on peut associer les syllabes po et ti pour mémoriser la signification des boutons.

Les traits inférieurs et supérieurs symbolisent les bords extérieurs et intérieurs du clavier, l'absence de trait indique donc la rangée du milieu.

➤ les nombres surlignés représentent la troisième rangée.



La saisie au clavier est assez problématique et dépend des systèmes d'exploitation. Si votre système en propose, essayez d'installer un clavier virtuel, ces symboles sont classés dans la page des caractères alphanumériques entourés.

Autre solution : le fichier *EmptyTab.ly* contient six lignes commentées en début de fichier comportant tous ces symboles. Il suffit de faire des copier/coller pour les insérer dans la tablature. Ce n'est pas une solution parfaite mais elle fonctionne.

ÉDITEURS

Certains éditeurs permettent de personnaliser les barres d'outil ou raccourcis clavier, ce qui peut être une piste pour avoir plus de facilité à utiliser ces symboles.

ALTERNATIVE

On pourrait parfaitement se contenter de la paire d'instructions :

```
\language "fisarmonica"
```

musique...

```
\language "italiano"
```

mais elles sont moins universelles.

La notation de la musique en est très simplifiée, puis qu'on ne saisit plus le nom de la note et son octave, ni le sens ou la rangée, mais uniquement le bouton, la durée, et éventuellement le doigté.

Cependant, il sera nécessaire d'indiquer à partir de quelle configuration de clavier il faut générer la partition et la tablature, de la même façon que l'indication de clavier `kbdOut` et pour les mêmes raisons que les indications de tonalité.

Cette indication se fera ainsi : `#(define kbdIn code)` en début de fichier.

Par ailleurs il faudra également indiquer à LILYPOND que la musique à compiler n'est plus écrite en anglais ou *italiano*, mais en *fisarmonica*, ce qui donnera au final une variable *Tablature* qui ressemblera à ce qui suit :

```
Tablature = {  
  \languageSaveAndChange "fisarmonica"  
  ⑦4 ⑧8 ④  
  \languageRestore  
}
```

Les instructions de langage sont un peu différentes de l'instruction habituelle `\language`, la première ne se contentant pas de changer de langage mais de plus elle enregistre le langage courant, et la seconde le restaure. Ces deux instructions sont donc parfaitement opérationnelles quelque soit le langage retenu pour le reste du fichier.

Lors de la compilation la conversion se fera alors de ces symboles vers les notes, mais uniquement, la tablature ne pouvant être calculée qu'à partir de notes. Il faudra donc dans un premier temps commenter la ligne

```
\new Staff \with { \diatoTabSetup } << { \tabDiatomelody \Melody } >>
```

mais ajouter sous la variable *Tablature* la commande

```
\displayLilyMusic \Tablature
```

ce qui permet de récupérer les dites notes dans le terminal qui a servi à lancer la compilation (ou dans la fenêtre journal, message ou log de votre éditeur s'il en possède une).

Il suffira ensuite d'un simple copier/coller pour intégrer ces notes dans la variable *Melody*, ce qui permettra avec une deuxième compilation d'avoir la tablature. Au préalable il faudra évidemment penser à (dé)-commenter les commandes idoines. À ce stade on peut très bien supprimer la variable *Tablature* ainsi que l'indication de clavier `kbdIn`, sauf si on suppose qu'une nouvelle compilation est à envisager.

ÉCRITURE DES BASSES - ACCORDS

Les basses ont en général la particularité suivante : elles sont affichées d'une part au-dessus de la portée de cette façon :

Do Ré Fa Sol Ré Mi

et d'autre part on attend que la tablature affiche une ligne de basses comme ceci :

C c C c C c C c D d D d D d D d F f F f G g G g D d D d E e E e

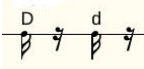
Pour que ce résultat puisse être obtenu à partir d'une seule saisie il va falloir adapter l'écriture des basses.

Le plus simple est de partir d'un exemple, la grille d'accords correspondant à ce qui est affiché ci-dessus (avec des noms de notes et pas des noms d'accords) :

```
Basses = {
    do2 re2
    fa4 sol re mi
}
```

Pour afficher ce résultat directement à partir du contenu de la seule variable *Basses*, la fonction de calcul de la tablature de basses `tabDiatobass` utilise des motifs rythmiques. Un motif contient deux types d'informations :

- la rythmique proprement dite, à savoir la décomposition en croches ou doubles-croches, avec ou non des silences ;
- les notes qui seront indiquées comme étant des basses (majuscule) ou des accords (minuscule).

Dans notre exemple le rythme  s'écrirait normalement : `re16^"D" r re^"d" r`.

L'utilisation des motifs permet de ne plus avoir à écrire que `re`, qui sera automatiquement transformé en `re16^"D" r re^"d" r`.

Le principe de fonctionnement de `tabDiatobass` est de transformer chaque occurrence d'une note dans la variable basse en son équivalent selon le motif, et ce *sans tenir compte de la durée initiale de la note* en question.

En conséquence on ne doit plus écrire les accords comme ci-dessus, mais comme ceci :

```
Basses = {
    do4 do re re
    fa sol re mi
}
```

On a simplement écrit autant de notes que l'on a besoin de motifs leur correspondant, ce qui revient à les écrire avec une durée d'une noire, qui sera écrite même si elle n'est pas utilisée par la fonction (on verra plus loin pourquoi).

Pour une valse, où le motif rythmique est sur trois temps, Basse Accord Accord, on donnera à chaque indication de note une durée d'une blanche pointée (2.).

La transformation des accords en tablature se fait tout simplement en utilisant la fonction `tabDiatobass` de `FISARMONICA` pour générer une variable *TabBasses*

```
TabBasses = {\tabDiatobass \samePitch \motif \transpose \scoreIn \scoreOut \Basses}
```

où `motif` correspond au modèle d'écriture de l'accord.

IMPORTANT !

Les fonctions de traitement des basses ne fonctionnent qu'avec des notes, et non des accords. On n'utilisera donc pas le mode `\chord-mode` contrairement à l'habitude.

OPTION

L'affichage des noms de basses se fait habituellement en notation anglaise mais on peut modifier ce comportement en modifiant l'instruction suivante du fichier de configuration :

```
#(define bass-Names
noteNames_EN )
```

LANGAGE

En réalité tous les motifs sont écrits en langage anglais par souci d'internationalisation, mais sont *langage indépendant*.

En outre, étant donné que seule la structure rythmique sera utilisée et calculée sur les notes des basses, le nom de la note utilisée pour les motifs n'a strictement aucune importance.

TRANSPOSITION

On utilisera pour la génération de la variable *TabBasses*, et uniquement à cet endroit, l'instruction de transposition automatique définie par les indications de tonalité.

Cette variable sera ensuite affichée dans le *DrumStaff* correspondant :

```
\new DrumStaff \with { \diatoBassSetup } << \TabBasses >>
```

On utilise un *DrumStaff* et non plus un *Staff* standard, ce qui permet d'imposer à toutes les notes d'être écrites sur une unique ligne, ce qui ne serait pas le cas autrement.

Entre l'appel de la fonction `\tabDiatoBass` et l'appel du motif, on voit également un appel à une autre fonction : `\samePitch`. C'est cette fonction qui permet de répéter chaque note autant de fois que nécessaire dans le motif. Si on omet d'appeler cette fonction, les notes n'étant pas répétées, la basse utilisera une note, l'accord la suivante, et ainsi de suite. Cette particularité a son intérêt comme on va le voir plus loin.

RENDU GRAPHIQUE

Par défaut ces indications de tirette se placent sous la ligne de basses.

INDICATION DE TIERCE

Certains modèles disposent d'une ou deux tirettes à la main gauche, permettant d'occulter ou non la tierce des accords, ou une des voix de basses (basses profondes). Il est possible d'indiquer leur état avec les mêmes indications que pour les registres.

MOTIFS RYTHMIQUES

Un certain nombre de motifs sont prédéfinis dans le fichier de configuration de **FISARMONICA** mais chacun est libre de les modifier, d'en ajouter ou en supprimer (voir explications en annexe).

Voici la représentation graphique des différents motifs prédéfinis dans **FISARMONICA** ainsi que leur équivalent en notation dans la variable *Basses* et le nom du motif.

Noms de motifs

Ces noms sont purement subjectifs et modifiables dans le fichier de configuration.

Cependant le principe même de ces motifs est d'être le plus universels possible, ce qui suppose un consensus de la part de tous les utilisateurs.

Affaire à suivre donc...

Le motif *Step*, par exemple, sera utilisé pour toutes les danses où les pas se font sur deux temps, comme les scottishs, polkas, etc. Le motif *March* sera utilisé sur des danses plus lentes.

Pas	Marche	Variantes			Bourrée	Jig	Variantes			Valse
c4	c2	c4	c4	c4	c4.	c4.	c4.	c4.	c4.	c2.
Step	March	LNBin	SNBin	BcBin	Bouree	Jig	LNtern	SNtern	BcTern	Waltz

Certains motifs, binaires ou ternaires, indiquent un jeu soit plus appuyé, comme *LN* (long note) ou allégé : *SN* (short note), voire assimilable à un bourdon comme *Bc* (Bass + chord).

Ainsi qu'il est facile de le constater, l'écriture de la tablature des basses est extrêmement simplifiée grâce à l'utilisation des motifs, pour autant qu'ils soient réguliers et se répètent à l'identique sur tout le morceau. Si l'on doit changer de motif rythmique en cours de morceau cela se complique un peu mais pas trop.

CHANGEMENT DE MOTIF RYTHMIQUE

Si on écrit un morceau où la rythmique change en cours de morceau, comme par exemple une scottish-valse, où la première partie est en 4/4 et la seconde en 3/4, il faudra séparer la grille d'accords en deux sous-variables :

```
BassesScottish = { sol4 sol la la la la sol sol }
```

```
BassesValse = { sol2. la la sol }
```

On regroupe ensuite ces deux sous-variables dans la variable *Basses* :

```
Basses = {
  \ChordsScottish
  \ChordsValse
}
```

et on applique à chaque partie son motif rythmique pour créer la variable *TabBasses* :

```

TabBasses = {
    \tabDiatoBass \samePitch \Step \transpose \scoreIn \scoreOut
\BassesScottish
    \tabDiatoBass \samePitch \Waltz \transpose \scoreIn \scoreOut
\BassesValse
}

```

MOTIFS DISSOCIÉS

On peut par ailleurs avoir besoin de motifs où la note de la basse et celle de l'accord seront différentes. Dans ce cas on utilisera toujours le même motif mais sans la fonction `\samePitch` et on écrira toutes les notes nécessaires.

On devra donc écrire non pas `re4 re4. do8 do4` mais `re8 re re re re do do do` pour obtenir la ligne de basse `\D d D d D c C c`, avec une basse en ré et accord en do sur le troisième temps.

Il est fortement recommandé d'appliquer le motif sur une mesure complète, même si cela alourdit un peu le code, car sinon le risque est grand d'avoir des barres de mesures indésirables qui apparaissent ça ou là.

Exemple :

```

BassesI = {
    re4 re do do
}
BassesII = {
    re8 re re re re do do do
}
BassesIII = {
    fa sol la si
}
Basses = {
    \BassesI
    \BassesII
    \BassesIII
}
TabBasses = {
    \tabDiatoBass \samePitch \Step \transpose \scoreIn \scoreOut
\BassesI
    \tabDiatoBass \Step \transpose \scoreIn \scoreOut \BassesII
    \tabDiatoBass \samePitch \Step \transpose \scoreIn \scoreOut
\BassesII
}

```

MOTIFS COMPOSÉS

Les quelques motifs pré-définis peuvent sembler insuffisants pour couvrir les besoins. Par exemple il n'y a pas de valse à cinq temps. On peut bien évidemment en écrire un :

```
WaltzV = { c8^"Bs" r c^"Ch" r c^"Bs" r c^"Ch" r c^"Ch" r }
```

et l'utiliser dans `\tabDiatoBass`. Mais ce n'est absolument pas nécessaire, on obtient le même résultat en combinant `Step` et `Waltz` ainsi :

```
\tabDiatoBass \samePitch {\Step \Waltz} \transpose \scoreIn \scoreOut
\Basses
```

On peut aller plus loin en n'utilisant `\samePitch` que sur une partie du motif composé.

Si on écrit `{\samePitch \Step \Waltz}` le motif est composé de `Step` auquel est ap-

LANGAGE

Bien que placés dans le fichier de configuration, lequel peut être écrit dans le langage que l'on veut, les motifs le sont uniquement en langage standard, par souci d'internationalisation.

Ils sont encadrés par des commandes de changement / restauration de langage qui rendent ceci transparent (commandes qu'il ne faut en aucun cas modifier !).

pliqué `samePitch`, suivi de `Waltz` auquel on n'applique plus rien, puisque la paire d'accolades ne le joint plus à `Step`, il sera donc dissocié.

On pourrait même insérer des répétitions dans le motif mais ce n'est pas vraiment à conseiller, cela détruirait l'équilibre des différents affichages des basses.

ÉCRITURE MANUELLE DE LA TABLATURE DE BASSES

On voit qu'au prix d'une écriture un peu plus complexe on peut obtenir un résultat très précis. Le tout étant de savoir jusqu'où il est intéressant d'aller plutôt que d'écrire soit-même à la main tout ou partie de la tablature des basses qui se présenterait alors sous la forme :

```
TabBasses = {
    \tabDiatoBass \samePitch \Step \transpose \scoreIn \scoreOut
\BassesI
    d16^"D" r8 c16^"c" r16 c16^"c" r16
    \tabDiatoBass \samePitch \Step \transpose \scoreIn \scoreOut
\BassesII }
```

Le motif du milieu est volontairement différent de l'exemple précédent puisqu'on écrit ainsi des motifs inexistantes et peu intéressants à créer.

Le gros inconvénient de cette méthode est que la transposition automatique ne peut plus s'appliquer au passage écrit manuellement.

AFFICHAGE DES NOMS D'ACCORDS AU-DESSUS DE LA PORTÉE

Ces accords sont par défaut les mêmes que ceux contenus dans la variable `Basses` mais ils seront malgré tout recopiés dans une variable `Chords`, dont on verra ci-dessous l'utilité.

On écrira donc juste après la variable `Basses` la variable `Chords` ainsi ;

```
Chords = { \transpose \scoreIn \scoreOut \Basses }
```

L'affichage se fera de façon classique avec un `Staff ChordNames` placé au-dessus de la portée :

```
\new ChordNames { \Chords }
```

ce qui n'affichera le nom des accords avec la bonne durée puisqu'on a pris la précaution d'indiquer celle-ci correctement dans la variable `Basses`.

Cependant le nom de l'accord ne sera pas répété sur tous les temps tout simplement parce qu'on aura pris la précaution d'ajouter l'instruction suivante :

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

et dans notre exemple on a de plus utilisé :

```
\frenchChords
```

qui affiche le nom des accords en français, mais ceci est une option laissée à l'appréciation de chacun.

Si vous souhaitez afficher des accords plus précis à l'intention par exemple d'un guitariste accompagnateur, il suffira de les écrire de la façon habituelle dans la variable `Chords`, en gardant bien entendu les instructions de transposition et en prenant garde de travailler en mode `\chordmode` :

```
Chords = { \transpose \scoreIn \scoreOut \chordmode { do2:sus4 re2:5
fa4:m ... } }
```

De même, si vous aimez répéter le nom de l'accord au début de chaque mesure et que l'écriture des notes dans la variable `Basses` ne le permet pas, il faudra écrire à nouveau vos accords dans cette variable.

SORTIE MIDI

Étant donné que les tablatures sont des aides à la lecture pour les musiciens qui ne savent pas lire la musique, la génération d'un fichier Midi correct peut aussi s'avérer une aide précieuse pour travailler d'oreille.

Par conséquent, dans la même optique que la fonction `tabDiatoBass`, `PISARMONICA` vous propose également `midDiatoBass` qui vous permet, toujours avec le principe des motifs, de remplacer un `do4` non plus par `do16 r do r` mais par `do,16 r <do sol> r`, avec la première note descendue d'une octave et la seconde agrémentée de sa quinte, ce qui permet une restitution sonore assez convaincante (pour autant que le midi puisse l'être...).

L'utilisation de cette fonction nécessitera deux motifs au lieu d'un, le premier servant à fabriquer la rythmique de la basse, le second celle de l'accord.

Pour utiliser ces motifs il suffit de créer une variable `MidBasses` ainsi :

```
MidBasses = \copyVoltaRepeats \Basses {
\ midDiatoBass \StepMidBs \StepMidCh \transpose \scoreIn \scoreOut
\Basses
}
```

L'instruction `\copyVoltaRepeats \ Basses` permet de récupérer la structure des répétitions et fins alternatives écrite dans la variable `Basses` afin de la réutiliser dans la génération des basses midi.

Ensuite on applique les motifs de basse et d'accord dans l'ordre à chaque sous-variable.

On pourra également utiliser des motifs composés (attention à bien regrouper les motifs `Bs` ensemble et les `ch` ensemble) comme pour la tablature.

```
\ midDiatoBass {\StepMidBs \WaltzMidBs } {\StepMidCh \WaltzMidCh }
\transpose \scoreIn \scoreOut \Basses
```

Il suffit maintenant d'utiliser `MidBasses` et non plus `Basses` pour générer le Midi.

```
\score {
  \unfoldRepeats {
    \new StaffGroup
      <<
        \new Staff{
          \set Staff.midiInstrument="accordion"
          \Melody
        }
        \new Staff{
          \set Staff.midiInstrument="accordion"
          \MidBasses
        }
      >>
    }
  % \layout {} %% dé-commenter pour vérification visuelle de la sortie
  midi
  \midi{}
}
```

ACCORDS CREUX

Sur ces modèles la tierce a été ignorée, comme bien souvent sur les instruments eux-mêmes.

On pourrait écrire des modèles où elle existe toujours mais cela obligerait à compliquer encore plus l'écriture de la partition pour un résultat Midi guère plus intéressant.

Mais ceci est parfaitement possible.

CLÉ DE BASSE

Si vous êtes pointilleux sur l'écriture des clés, vous pouvez ajouter l'instruction `\clef bass` devant `\MidBasses` ce qui affichera une clé de fa sur la portée contenant ces notes. Le son produit sera strictement le même.

AFFICHAGE

On remarquera la commande commentée `\layout {}` qui permet, une fois dé-commentée, l'affichage de la sortie midi sous forme de partition à deux portées, ce qui permet de vérifier que le résultat est conforme à ce qui est attendu, ce qui est particulièrement utile dans le cas d'utilisation de motifs complexes ou d'alternance de motifs.

HISTORIQUE, REMERCIEMENTS ET DÉDICACE

Pourquoi mélanger ces trois concepts dans une même section ? Parce tout est imbriqué...

Au départ, en 2008/2009, j'étais parti sur l'idée de détourner les tablatures de guitare, avec un script externe en *Python* pour générer une pseudo partition avec des notes décalées des valeurs nécessaires afin que leur affichage représente bien le numéro du bouton de la rangée de clavier. Mais il y avait beaucoup de limitations et j'ai fini par perdre de mon ardeur.

Peu après, PENBEUZ a repris l'idée, s'est (légèrement) inspiré de mes travaux mais est parti dans une autre direction, l'utilisation de **DrumStaff** et de *markup* au lieu de mes **TabStaff**, et toujours avec un script externe en *Python*, mais accompagné de quelques fonctions en *Scheme*. Et il a fini par perdre lui aussi de son ardeur.

Poussé par le besoin d'écrire des tablatures correctes pour ma fille, je m'y suis remis début 2013, avec un nouvel axe : scripts uniquement en *Scheme* pour ne plus avoir à sortir de LILYPOND, et abandon des **TabStaff** (donc un grand merci à PENBEUZ pour cette idée).

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) : elmamyra ; Jean-Charles Mahieude ; Seventies ; Gilles (notamment pour les idées des motifs de basses et notamment de basses Midi), Gérard (pour ses idées que j'ai développées dans la section Accessibilité, et pour sa relecture attentive de cette documentation) et m'excuser d'en oublier probablement encore beaucoup d'autres.

En ce qui concerne la dédicace, je ne me suis pas posé la question une seule seconde, j'ai tout de suite pensé à celui des différents formateurs et musiciens que j'ai côtoyés qui a le plus contribué à me faire apprécier l'accordéon diatonique, de par son charisme et sa chaleur humaine, sans parler de ses talents de musicien et de pédagogue, je veux parler de Marc PERRONE.

ANNEXES

A) LICENCE

Sauf mention contraire explicite, le contenu de l'ensemble des fichiers composant **FISARMONICA** est placé sous la licence **Creative Commons BY NC ND**, y compris la présente documentation.

En outre, merci de bien vouloir également considérer **FISARMONICA** comme un *e-PostCard-Ware* : si vous appréciez ce greffon merci d'envoyer une carte postale électronique à l'auteur :

daniel@cartron.org

ABONNEMENT

Profitez de l'envoi de votre *e-card* pour indiquer si vous souhaitez être abonné à la liste de discussion de **FISARMONICA**.

C'est de toutes façons le seul mode d'inscription existant ;-)

B) RÉFLEXIONS SUR LES PLANS DE CLAVIER

Il a été fait plus haut mention des plans de clavier assez fantaisistes fournis par certains fabricants qui sont manifestement plus doués pour la fabrication d'instruments que pour la musique.

Les erreurs sont de deux sortes : le nom des altérations supplémentaires situées sur les boutons 1 (ou décalées sur la troisième rangée) mais aussi parfois la gamme complète d'une rangée sont notés de façon aberrante.

Cette dernière étant (heureusement) moins répandue nous allons en parler en premier.

GAMMES ABERRANTES

L'exemple le plus frappant est la gamme de do #, laquelle n'est pas rare sur les modèles irlandais. Cette gamme a la particularité d'avoir des dièses sur toutes les notes. Or comment sont-elles indiquées ? On trouve un fa #, certes, mais aussi un fa # (et sur le même bouton !) à la place du mi # qui devrait s'y trouver. Idem pour le do # au lieu du si #.

Certes le son est le même, et on n'est pas habitué à parler de mi # ou de si #, mais il n'en demeure pas moins qu'il s'agit d'une belle hérésie en termes de théorie musicale. On ne peut avoir une gamme avec deux fa et deux do, et aucun mi ni si.

Il est probable que les fabricants ne peuvent se résoudre à parler de telles notes puisque toutes les tonalités qui en comportent présentent les mêmes caractéristiques : si # au lieu du do # pour une gamme de si majeur, pour ne prendre qu'un exemple de plus.

ALTÉRATIONS SUPPLÉMENTAIRES

Commençons par les tonalités que nous avons choisies comme références pour notre dictionnaire de notes.

DO/FA

RANGÉE EXTÉRIEURE

Cette rangée est en do majeur. Si on veut passer en mineur, le la mineur pour être précis, on a besoin d'un sol #. Il est bien indiqué, en poussant.

Pour pouvoir moduler dans les tons voisins, le fa et le sol, seule l'altération caractéristique à la tonalité de sol est nécessaire, celle de fa se trouvant déjà sur la deuxième rangée. Cette note est un fa #, qui est présente en tirant.

RANGÉE INTÉRIEURE

Le relatif mineur est le ré mineur, qui nécessite un do #. Il est bien indiqué, toujours en

DES NOMS !

Désolé, il ne sera donné aucun nom. Il suffit de chercher un peu sur le Web pour trouver tout seul quels fabricants sont concernés.

C'EST PAREIL !

Encore une fois désolé, mais : non, un do # et un ré b ce n'est pas la même chose.

On entend certes la même chose, un peu comme le son s peut venir du s ou du c suivi d'une voyelle.

Il s'agit d'une question de syntaxe et non de sonorité.

Exemple : se et ce se prononcent de la même façon, alors que le s du premier indique qu'il s'agit d'un prénom possessif, et le c du second qu'il s'agit d'un pronom démonstratif.

En musique c'est pareil, il y a des règles : une gamme doit comporter obligatoirement une note et une seule de chaque degré (do ré mi...) quelle que soit son altération.

C'est la raison pour laquelle les plans de clavier cités sont incorrects.

poussant.

La modulation en si \flat nécessite un mi \flat qui est également à sa place en tirant.

LIMITES

Ces considérations sont bien entendu limitées aux claviers à deux rangées ou à ceux dont les trois rangées suivent le schéma continental standard.

Les modèles dont la troisième rangée ne contient que des altérations ont été conçus dans l'optique d'atteindre à un pseudo-chromatisme, ce qui rend la dénomination des notes moins rigoureuse.

Pendant une bonne partie de ce qui est écrit ci-contre s'applique probablement.

NATUREL ?

Dans une partition, les altérations (hormis celles dues au mode mineur) seront indiquées une bonne fois pour toutes à la clé et ne seront nullement visibles dans la partition.

Il sera alors peu évident de déterminer si le fa que vous voyez est le fa naturel ou le fa dièse de votre plan de clavier.

Sauf si vos connaissances en théorie musicale sont suffisantes pour faire la différence aisément, auquel cas ces considérations ne vous servent à rien...

PARTAGEZ

Pensez à communiquer votre configuration à l'auteur de **FISARMONICA** afin qu'elle puisse ensuite être partagée avec le plus grand nombre.

LOGIQUE DES ALTÉRATIONS

Quelle que soit la rangée, l'altération permettant de jouer en mineur est poussée, celle permettant de moduler est tirée.

Simple, non ? Pas tant que ça, il suffit d'éplucher les plans de clavier de différents fabricants pour constater que ce n'est pas toujours le cas, mais ceci sort du propos de ce document.

Et maintenant regardons le clavier d'un Sol/Do...

SOL/DO

RANGÉE INTÉRIEURE

Commençons par cette rangée qui est commune à la configuration précédente.

Comme pour le modèle en Do/Fa, le sol \sharp permettant de passer en mineur est bien présent en poussant, et pour la modulation en fa on a bien le si \flat en poussant.

RANGÉE EXTÉRIEURE

Pour la modulation en ré : le do \sharp est bien à sa place, mais pour le relatif mineur du sol majeur qui est le mi mineur et nécessite un ré \sharp , on trouve souvent un mi \flat ...

Pourquoi un mi \flat et non un ré \sharp ? Est-ce un souci esthétique, pour obtenir un équilibre visuel avec le si \flat ? Quoiqu'il en soit ce n'est musicalement pas correct non plus.

AUTRE TONALITÉS

Plus on s'éloigne du Do/Fa et plus on trouve d'altérations illogiques, non seulement mal nommées, mais de plus non adaptées à un jeu mineur ou à des modulations. Nous en resterons donc là. De plus non seulement il est amusant de constater qu'on trouve non seulement des divergences entre fabricants (certains nomment correctement les notes, d'autres non), mais en outre certains proposent des modèles dont la logique diffère d'une tonalité à l'autre, pour le même modèle... L'aspect auto-transpositeur de l'accordéon diatonique n'est réel qu'en mode majeur, et encore sans modulation...

CONCLUSION

Ce qui a motivé cette réflexion est que l'on ne peut décemment pas construire un dictionnaire de configuration de clavier ni écrire une partition en se fiant aux plans trop souvent erronés des fabricants.

En conséquence si d'aventure la partition que vous venez de saisir vous indique des \color{red} rouges alors que vous savez que la note existe, vérifiez que vous n'êtes pas victime d'un plan fantaisiste...

Et surtout si vous voulez créer une nouvelle configuration de clavier vérifiez que le plan que vous avez en main est correctement écrit à la lumière de ce qui précède.

C) NOUVELLE CONFIGURATION DE CLAVIER

Afin de faciliter la conception d'une nouvelle configuration de clavier, **FISARMONICA** est accompagné d'un fichier nommé *keyboard-layout.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 de clavier 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 nom_de_configuration
  ( ( "kbd-ref" ."cte" )
    ( (note octave altération) . (sens rang "bouton") )
    ( (note octave altération) . (sens rang "bouton") )
    ( (note octave altération) . (sens rang "bouton") )
    ( (note octave altération) . (sens rang "bouton") )
  )
)
```

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

Le terme en rapport avec `kbd-ref` est le code de la configuration, qui sera utilisé pour retrouver son nom dans la table de traduction correspondante.

La référence de la note est un chiffre, allant de 0 pour le do à 6 pour le si.

L'octave est égale à :

- -1 pour les notes les plus graves du clavier, celles qui sont notées sans ' ni , en syntaxe LILYPOND, ou qui sont écrites sous la portée, pour simplifier ;
- 0 pour les notes qui sont situées aux environs des boutons 3 à 6 (sauf pour les rangées 3 partielles, bien entendu), écrites dans la partie inférieure de la portée, et notées avec une ' en syntaxe LILYPOND ;
- 1 pour les notes de l'octave suivante ;
- 2 pour les notes les plus aiguës du clavier.

L'altération est égale à :

- 0 pour un bécarre (pas d'altération) ;
- 1/2 pour un dièse ;
- -1/2 pour un bémol.

Le sens est égal à :

- 5 pour poussé ;
- 6 pour tiré.

Le rang est le numéro de la rangée de clavier, en partant de 1 pour le clavier extérieur , puis 2 voire 3.

Et bouton est égal au numéro de bouton tout simplement...

Voici à titre d'exemple le début du dictionnaire du modèle par défaut (les termes remplacés ont été laissés en italique pour la facilité de compréhension), et les lignes sont commentées ;

```
#(define cte ;
' ( ( "kbd-ref" . "cte" )
  ( ( 0 0 1/2 ) . ( 5 1 "1" ) ) = do / octave 0 / dièse / poussé / rangée 1 / bouton 1
  ( ( 1 0 1/2 ) . ( 6 1 "1" ) ) = ré / octave 0 / dièse / tiré / rangée 1 / bouton 1
  ( ( 6 0 -1/2 ) . ( 5 2 "1" ) ) = si / octave 0 / bémol / poussé / rangée 2 / bouton 1
  ( ( 4 0 1/2 ) . ( 6 2 "1" ) ) = sol / octave 0 / dièse / tiré / rangée 2 / bouton 1
)
)
```

En outre, étant donné que l'on retrouve souvent les mêmes éléments d'une configuration à l'autre (les boutons 2 à 10 ou à 11 sont très souvent les mêmes), ceux-ci sont définis dans des dictionnaires de base (matrices) lesquels sont ensuite utilisés comme briques pour définir les configurations tout simplement en leur ajoutant ce qui fait leur spécificité. On trouvera donc les dictionnaires de base suivants :

- les boutons 2 à 11 de la rangée de do sont dans une matrice nommé `base_c` ;

QUEL CODE ?

Vous pouvez utiliser n'importe quel code, pour autant qu'il soit utilisé à l'identique dans la table des noms de configuration, mais il est judicieux de garder le concept actuel de formation des noms, surtout si cette configuration va être partagée.

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.

ÉCRITURE

Il est nécessaire d'écrire le numéro de bouton entre deux *double quote* afin qu'il soit interprété comme du texte.

- les boutons 2 à 10 de la rangée de fa sont dans une matrice nommé `base_f` ;
- les matrices contenant les boutons correspondant à une extension de ces matrices de base sont préfixés `x_` suivi du nom de la configuration qu'ils servent à créer.

RECYCLAGE

Il se peut tout-à-fait que votre configuration puisse se contenter de quelques boutons à ajouter aux matrices de base pré-définies, notamment s'il s'agit d'une configuration de deux rangées et demi.

Ces matrices sont ensuite assemblées ainsi :

```
(define cte
  (append
    base_c
    base_f
    x_cte
  )
)
```

La configuration `cte` est construite à partir de `base_c`, `base_f` et `x_cte`.

LANGAGE

Comme indiqué plus haut il convient d'écrire les motifs en langage par défaut.

Étant donné qu'une seule note est utilisée ce n'est pas trop gênant.

De plus, la note en elle-même n'a aucune importance. On pourrait aussi bien écrire `d` ou `f`, le motif resterait valide.

D) NOUVEAU MOTIF RYTHMIQUE

La syntaxe des modèles est relativement simple eu égard à la puissance de la fonction.

Prenons un exemple :

```
motif = { c16^"Bs" r c^"Ch" r }
```

Explications :

- Le contenu de `{ }` définit la façon dont seront transcrites les occurrences des notes de *Basses*. Le nom des notes, `c` en l'occurrence, n'a aucune importance, il sera remplacé par celui de la note réelle écrite dans *Basses*.
- La durée des différents éléments du modèle est libre, et totalement indépendante de celle de la variable *Basses*, rappelons-le, ce qui laisse une très large possibilité d'expression. Il est ainsi possible d'avoir une basse longue et un accord court ou le contraire...
- La partie `"Bs"` ou `"Ch"` du modèle indique à la fonction quelle indication textuelle sera affichée à l'aplomb de la note : `Bs` signifie Basse, soit le nom de la note (en notation anglo-saxonne) en majuscule, et `ch` ce même nom en minuscule pour indiquer l'accord. Il existe également une notation `Bsch` qui renvoie comme on peut s'en douter les deux lettres accolées. Ces dénominations ne doivent pas être modifiées.

MODIFICATION D'AFFICHAGE

Si on écrit `c16^"Bs" r c^"Ch" r` dans le motif la lettre représentant l'accord sera écrite sous la ligne de basses et non au-dessus.

Il est judicieux avant de vous lancer dans la confection de vos propres motifs, d'étudier ceux qui se trouvent dans *fisarmonica-setup.ly*.

Vos motifs personnalisés devront donc également être placés à cet endroit, ou bien directement dans le fichier de votre morceau de musique si :

- ce motif ne sera jamais réutilisé ;
- vous envisagez de partager votre fichier, auquel cas il faut aussi transmettre le nouveau motif.

Bs Ch

L'écriture de ces indications textuelles ne doit absolument pas être modifiée, faute de quoi *FISARMONICA* ne retrouverait plus ses petits...

Le rendu de cette indication textuelle sera bien évidemment automatiquement calculé en fonction de la note de la grille d'accord à laquelle elle s'applique.

E) NOUVEAU MOTIF MIDI

Comme expliqué plus haut les motifs midi fonctionnent par paire, le premier servant à fabriquer le son de la basse et le second celui de l'accord.

Exemples :

```
patIMidBs = { do16 r r8 }
```

```
et patIMidCh = { r8 do16 r }
```

```
patIIMidBs = { do8 r r2 }
```

```
et patIIMidCh = { r4 do8 r do r }
```

Leur syntaxe est exactement la même que celle des motifs de basse, excepté qu'il faut remplacer une partie des notes par des silences.

Et bien entendu les indications textuelles `^"Bs"` et `c^"Ch"` n'ont plus lieu d'être.

À partir de ces motifs la fonction `midDiatoBass` effectuera les opérations suivantes :

- elle ajoutera à toutes les notes correspondant à `motifMidBs` la même note transposée d'une octave vers le bas ;
- elle y ajoutera ensuite toutes les notes correspondant à `motifMidCh` ;
- elle y ajoutera à nouveau toutes ces notes transposées d'une quinte vers le haut.

Le code qui effectue ceci est tout simplement :

```
\changePitch $patI { $music s }
\changePitch $patI \transpose do' do { $music s }
\changePitch $patII { $music s }
\changePitch $patII \transpose do sol { $music s }
```

auquel on pourrait ajouter

```
\changePitch $patII \transpose do mi { $music s }
```

pour avoir également la tierce majeure, par exemple. Les termes `patI` et `patII` sont les variables qui contiennent le nom des motifs passés en argument lors de l'appel de la fonction.

F) MODIFICATION DES DIAGRAMMES DE VOIX

Ces notations sont enregistrées à la fin du fichier *fisarmonica.ly*, sous forme de markup :

```
oxo = ^\markup\box\center-column {
  \draw-circle #0.3 #0.1 ##f
  \raise #1.5
  \draw-circle #0.3 #0.1 ##t
  \raise #3
  \draw-circle #0.3 #0.1 ##f
}
```

Le `##f` signifie que le centre du cercle n'est pas coloré, donc pour nous il signifie ouvert.

Vous pouvez créer d'autres diagrammes si votre accordéon le nécessite.

G) EMPTYTAB.LY

Le fichier modèle est abondamment commenté, étant entendu que chacun l'adaptera à sa manière.

La totalité des options modifiables localement y sont écrites et commentées, il sera probablement judicieux d'en retirer une grande partie, en fonction de vos besoins.

Toutes les variables et la configuration des rendus Pdf et Midi s'y trouvent également, vous pouvez les modifier si vous savez ce que vous faites...

POSITIONNEMENT

Il faut une instruction `\raise` entre chaque cercle pour les décaler verticalement. Si vous trouvez les écartements trop importants vous pouvez essayer de modifier ces valeurs.

VERSION TABLATURE

Une version dédiée à ce mode de saisie est également disponible sous le nom *EmptyTab2.ly* comportant en plus la variable *Tablature* et les lignes des symboles décrites plus haut. Cette version ne sera correctement lisible qu'après installation de la police adéquate.

EMPLACEMENT DES FICHIERS

Comme pour le greffon **FISARMONICA**, 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 **FISARMONICA**, sinon les configurations de celui-ci seraient écrasées.

FIN DE JAZZ

L'instruction `\jazzOn` a son pendant, `\jazzOff`, si vous souhaitez sortir de ce mode avant la fin de votre partition, ce qui est peu probable.

ÉCRITURE DE TABLATURES

Les explications qui suivent sont également valables si vous écrivez directement votre tablature à l'aide des symboles contenus dans la police *fisarmonica*.

WYSIWYG

LILYPOND n'est pas du tout *Wysiwyg*, mais ne nous leurrions 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 accordéon diatonique.

H) 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 complètement ré-écrite et s'appelle *FreeJazz.ttf* dans cette version. C'est celle qu'il faudra installer pour bénéficier du greffon.

De même pour le fichier *LilyJAZZ.ily*, où un certain nombre d'options ont été rajoutées, modifiées ou commentées, et qu'il faudra donc utiliser 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 là où il est souhaité, probablement partout, ce qui signifie la mélodie, les tablatures et les basses, en ajoutant dans le bloc `\score` l'instruction `\jazzOn` avant l'indication de contexte, comme dans l'exemple suivant :

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

Enfin il faut modifier l'arbre des polices en ajoutant les instructions suivantes dans le bloc `\paper` (à créer au besoin) :

```
TailleGlobale = #20
```

```
#(define fonts (make-pango-font-tree "FreeJAZZ Text" "FreeJAZZ Text" "FreeJAZZ Text" (/ TailleGlobale 20)))
```

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

I) DÉCOUVRIR LILYPOND AVEC FISARMONICA

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 *EmptyTab.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 (ou acceptant un greffon adapté), 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. Nous allons donc maintenant entrer dans le vif du sujet.

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 *EmptyTab.ly*. Il suffit de remplacer l'explication entre " " par le texte adéquat en prenant bien garde à ne pas supprimer ces guillemets.

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 *Melody* ou *Basses*.

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 ici, soit directement, soit en insérant d'autres variables, voire 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 à l'anglaise (**a b c d e f g**) qui a l'avantage de ne demander qu'une seule frappe. 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 *EmptyTab.ly*, mais on peut la supprimer ou la transformer en commentaire.

L'altération en français se note simplement **a** pour dièse et **b** pour bémol. En anglais c'est respectivement **es** et **is**.

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

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...

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.

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.

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.

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 `sol''` qui suit le mot clé `relative` indique la note centrale de la musique (ici en fait la note centrale du rang extérieur du clavier, 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.

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

```
Melody = relative sol'' {
  la2|i\1-2 do4-3 | la blanche, tiré, rangée 1, doigt 2 ; do noire, doigt 4
  si2\o-2 sol4\1-1 | si blanche, poussé, doigt 2 ; sol noire, rangée 1, doigt 1
  la2.\i\1-2 | la blanche pointée, tiré, rangée 1, doigt 2
}
```

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 inévitablement des erreurs.

Lesquelles seront facilement évitées si la première note de la mesure a une indication de durée explicite, et identifiées si les | préconisés 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).

LES RÉPÉTITIONS

La musique pour diatonique est souvent conçue comme un ensemble d'une ou plusieurs phrases musicales répétées mais avec des fins différentes. On écrira ceci sous forme de plusieurs expressions musicales imbriquées :

```
\repeat volta 4 { On indique le début de la partie commune et le nombre de
  reprises de celle-ci
  la4|i do8 la | La mélodie commune à toutes les reprises est elle-même
  si4\o si8 do | une expression musicale et donc entre accolades
} Fin de la partie commune
\alternative { Début des fins alternatives (expression musicale)
  { re8\o mi re si | La première fin, sur deux mesures donc deux lignes, autre
  do4|i la | } expression musicale donc entre accolades aussi
  { re8\o mi re do | La deuxième fin...
  la4|i la | }
  { re8\o mi re si | Etc.
  do4|i la | }
  { re8\o mi re do |
  la2|i | }
} Fin des fins alternatives
```

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 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 BASSES

Comme expliqué plus haut, les basses sont écrites dans leur propre variable (avec éventuellement des sous-variables). Leur structure devra être la même que celle de la mélodie (au niveau des reprises).

LES ACCORDS

Contrairement aux notes des basses, 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.

Bon travail avec LILYPOND !

NOTATION ABSOLUE

Il n'est pas utile d'utiliser une notation relative pour les basses, les fonctions de traitement de celles-ci rendent les indications d'octave superflues.

ALLER PLUS LOIN

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

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