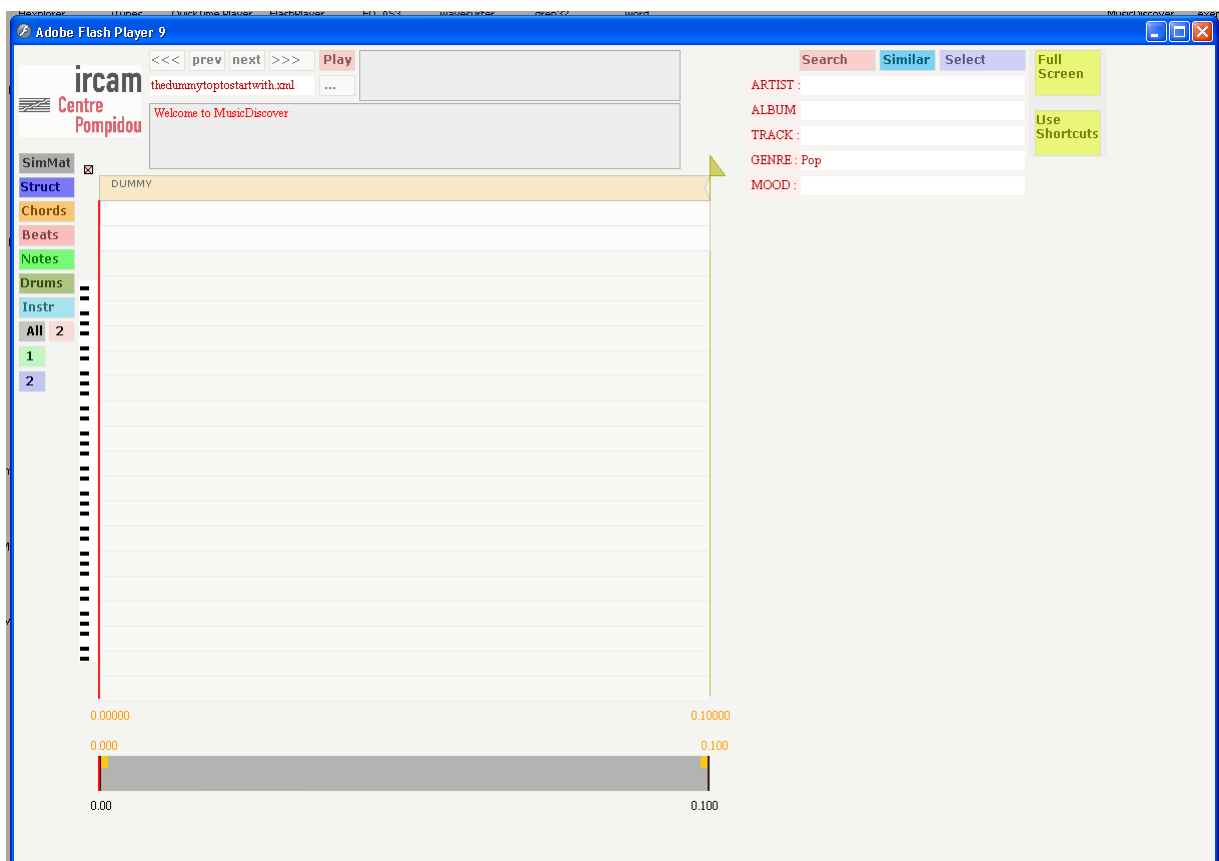


## MusicDiscover - Graphical interface documentation (tache 4 : description sémantique structurée)

L'interface de navigation MusicDiscover permet de synthétiser les différentes extractions faites sur les musiques, de les représenter graphiquement, et de permettre une navigation interactive afin d'explorer en direct ces différents paramètres.

### 1.1 Présentation de l'interface de navigation

L'interface de navigation MusicDiscover.swf a été développée en Flash9 / Actionscript 3 sous l'environnement de programmation FlashDevelop. Etant écrite en flash, on peut donc l'utiliser sous diverses plateformes (mac, pc, linux, etc) sous la forme d'un exécutable autonome ou bien depuis un navigateur internet de type firefox. La page d'accueil de l'interface graphique est comme suit :



on remarquera la colonne de gauche avec les boutons cliquables pour afficher diverses représentations (voir chapitres suivants), la timeline globale et zoomable en bas en gris, et la section de recherche dans une base de données à droite.

## 1.2 Structure XML des données

L'ensemble des données extraites par divers outils est interfacée dans le navigateur flash en utilisant le format standard XML. Ce format a l'avantage de pouvoir être lu et parsé facilement par flash, et est en quelque sorte un format pivot entre différents outils. Nous détaillerons ci-dessous les différentes sections de ces fichiers XML.

### 1.2.1 le fichier \_top.xml

Ce fichier est associé à chaque titre musical et contient des références vers les différents sous fichiers XML.

```
<?xml version="1.0" encoding="UTF-8"?>
<topleveldescription generator="ircamstructure" version="1.0"
date="2007/02">
  <media>Nirvana_.mp3</media>
  <struct>Nirvana_struct.xml</struct>
  <beat>Nirvana_correct_beat.xml</beat>
  <chord>Nirvana_chord.xml</chord>
  <notes>Nirvana_notes.xml</notes>
  <instruments>Nirvana_inst.xml</instruments>
  <drums>Nirvana_drums.xml</drums>
  <simmatimage>NirvanaSimMat.JPG</simmatimage>
</topleveldescription>
```

Les chemins vers les fichiers peuvent également être des chemins absolus.

### 1.2.2 le fichier \_struct.xml

C'est le fichier qui décrit le morceau en structure musicale (différentes sections, correspondant à des zones de musique similaire). On peut se servir de cette extraction pour représenter automatiquement les passages d'un morceau, correspondant intuitivement à la notion de couplet / refrain / pont, mais évidemment cela n'est qu'une vue de l'esprit.

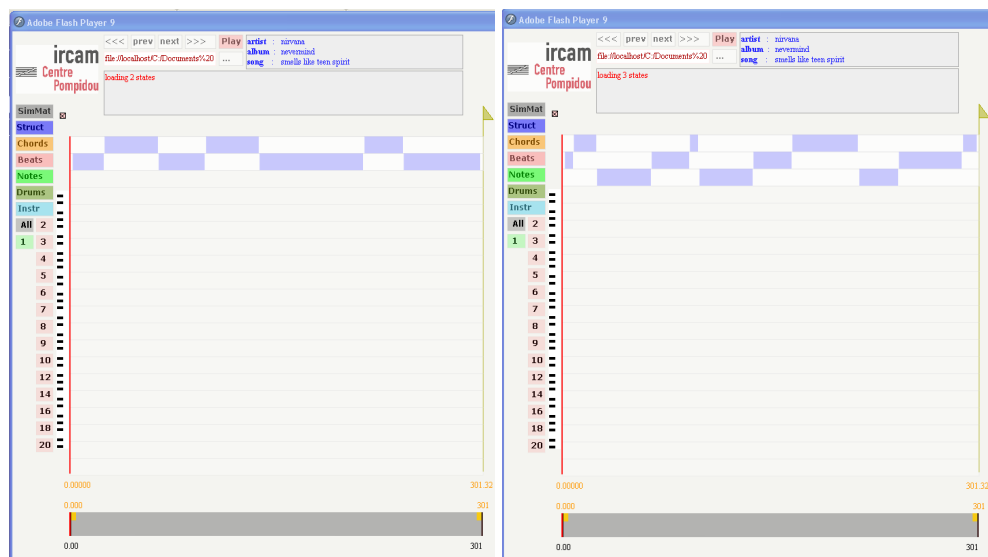
Le fichier XML est comme suit :

```
<structuredescription generator="ircamstructure" version="1.8"
date="2005/09">
  <media>Nirvana_.mp3</media>
  <description type="state" nbstate="2">
    <segment num="2" start="1.784" stop="24.6357"/>
    <segment num="1" start="25.2371" stop="64.3255"/>
    <segment num="2" start="64.9268" stop="98.603"/>
    <segment num="1" start="99.2044" stop="137.6915"/>
    <segment num="2" start="138.2928" stop="214.0643"/>
    <segment num="1" start="214.6656" stop="242.9296"/>
    <segment num="2" start="243.5309" stop="299.4575"/>
  </description>
  <description type="state" nbstate="3">
    <segment num="2" start="1.784" stop="7.7976"/>
    <segment num="1" start="8.399" stop="24.6357"/>
    <segment num="3" start="25.2371" stop="64.3255"/>
  </description>
</structuredescription>
```

```
<segment num="2" start="64.9268" stop="91.9881"/>
<segment num="1" start="92.5894" stop="98.603"/>
<segment num="3" start="99.2044" stop="137.6915"/>
<segment num="2" start="138.2928" stop="165.9554"/>
<segment num="1" start="166.5568" stop="214.0643"/>
<segment num="3" start="214.6656" stop="242.9296"/>
<segment num="2" start="243.5309" stop="289.2343"/>
<segment num="1" start="289.8357" stop="299.4575"/>
</description>
</structuredescription>
```

Les différentes descriptions permettent de choisir un niveau de finesse dans la décomposition du morceau en sous structures. C'est le paramètre nbstate (ici 2 ou 3) qui permet d'ajuster cette représentation.

Ce fichier est lu par l'interface graphique en flash et représenté comme suit :



représentation avec 2 , puis 3 états de structure.

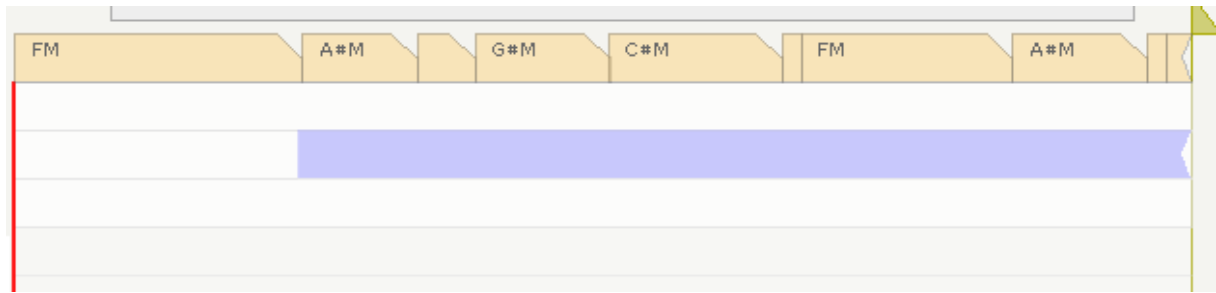
Chaque objet rectangulaire bleu est cliquable afin de permettre l'écoute de la section correspondante.

### 1.2.3 le fichier \_chords.xml

Ce fichier XML contient les informations extraites de suite d'accords. Il est comme suit :

```
<structuredescription generator="ircamchords" version="1.0" date="2007/10">
<description type="chords">
<segment start="0.000000" stop="1.812245" chord="FM"/>
<segment start="1.812245" stop="2.537143" chord="A#M"/>
<segment start="2.537143" stop="2.899592" chord="D#M"/>
<segment start="2.899592" stop="3.745306" chord="G#M"/>
<segment start="3.745306" stop="4.832653" chord="C#M"/>
<segment start="4.832653" stop="4.953469" chord="fm"/>
<segment start="4.953469" stop="6.282449" chord="FM"/>
<segment start="6.282449" stop="7.128163" chord="A#M"/>
<segment start="7.128163" stop="7.248980" chord="D#M"/>
<segment start="7.248980" stop="8.940408" chord="G#M"/>
<segment start="8.940408" stop="9.061224" chord="A#M"/>
<segment start="9.061224" stop="10.390204" chord="FM"/>
</description>
</structuredescription>
```

Les accords sont représentés sous la forme d'un texte affiché dans une zone avec un début et une fin. La représentation graphique correspondante est la suivante :



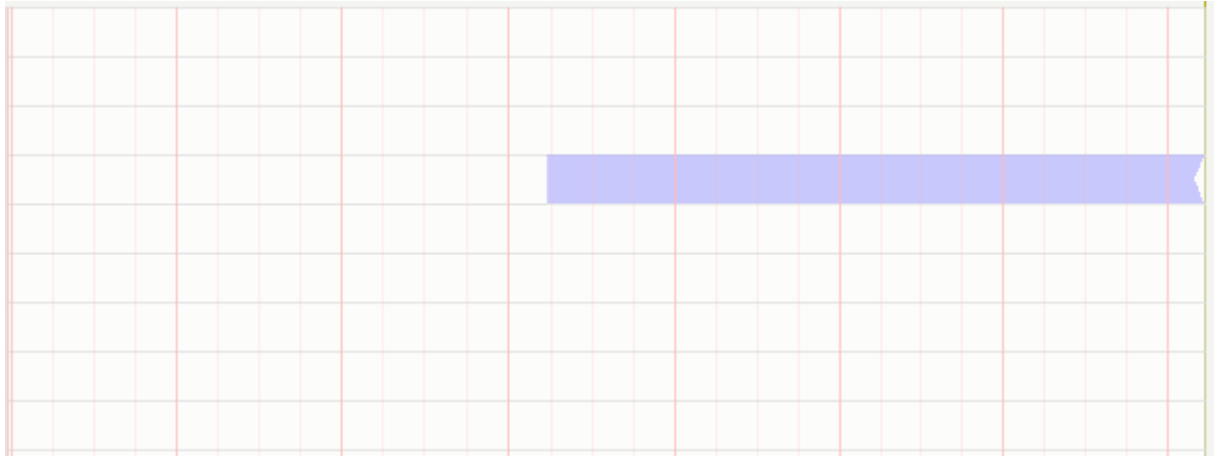
On peut également cliquer sur chaque élément graphique pour entendre la section correspondant à chaque accord.

#### 1.2.4 le fichier `_beat.xml`

Ce fichier XML représente le rythme sous la forme de temps. Le premier temps est distingué. Le fichier est comme suit :

```
<beatdescription generator="ircambeat" version="1.9" date="2007/05">
  <media>
    Nirvana_##_Smells_Like_Teen_Spirit.wav
  </media>
  <segment start="-0.1" stop="298.2841">
    <markerDS>
      <marker id="1" num="1">
        -0.088531
      </marker>
      <marker id="2" num="2">
        0.420487
      </marker>
      <marker id="3" num="3">
        0.931917
      </marker>
      <marker id="4" num="4">
        1.443007
      </marker>
      <marker id="5" num="1">
        1.950916
      </marker>
      <marker id="6" num="2">
        2.457816
      </marker>
      <marker id="7" num="3">
        2.965624
      </marker>
      <marker id="8" num="4">
        3.476488
      </marker>
    </markerDS>
  </segment>
</beatdescription>
```

Ici nous avons un morceau en 4 temps avec le premier temps spécifié avec l'attribut `num=1`. La représentation graphique de cette information est comme suit :



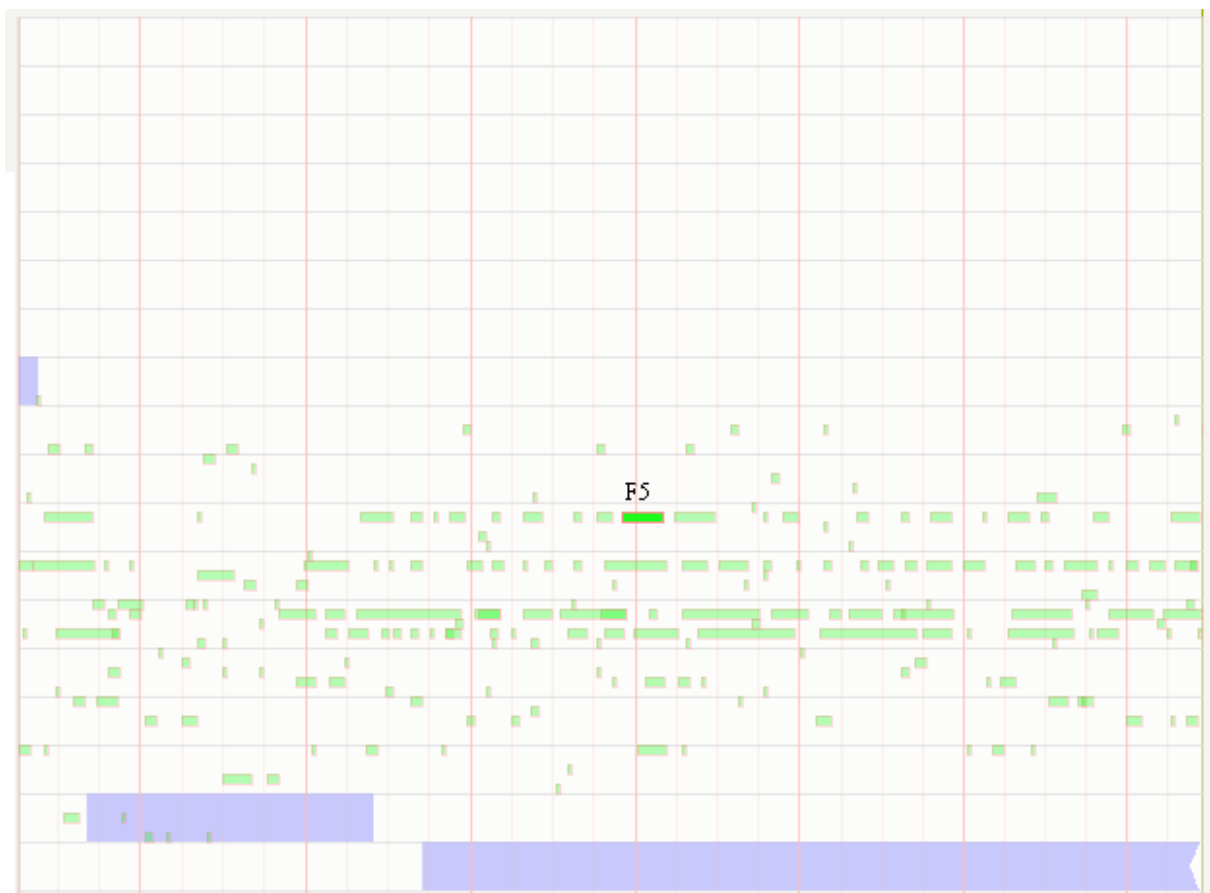
On voit apparaître le premier temps en gras et les 3 temps suivants en trait fin.

### 1.2.5 le fichier \_notes.xml

Ce fichier XML contient l'extraction des informations de notes, via un algorithme de détection de FO multiples. Ce fichier est comme suit :

```
<structuredescription generator="matlab" version="1.0" date="2007/010">
<description type="midinote">
<segment start="0.139286" stop="0.185726" midinote="54" track="1" velocity="7.306315e+00" />
<segment start="0.325045" stop="0.417925" midinote="62" track="1" velocity="2.885359e+01" />
<segment start="0.557245" stop="0.882324" midinote="41" track="1" velocity="5.810454e+01" />
<segment start="0.557245" stop="1.346723" midinote="53" track="1" velocity="5.894776e+01" />
<segment start="0.557245" stop="0.743005" midinote="58" track="1" velocity="4.607130e+01" />
<segment start="0.650125" stop="1.021644" midinote="48" track="1" velocity="7.086617e+01" />
<segment start="0.650125" stop="0.743005" midinote="58" track="1" velocity="2.202337e+01" />
<segment start="0.696565" stop="1.021644" midinote="65" track="1" velocity="3.140779e+01" />
<segment start="0.743005" stop="0.882324" midinote="67" track="1" velocity="3.321819e+01" />
</description>
</structuredescription>
```

La représentation graphique correspondante est sous forme classique de piano roll. Chaque note affichée est cliquable et permet d'écouter une note de piano de référence pour comparer l'extraction avec le morceau joué, en direct. Les informations de notes se superposent aux autres paramètres déjà évoqués ci-dessus (struct, beat).



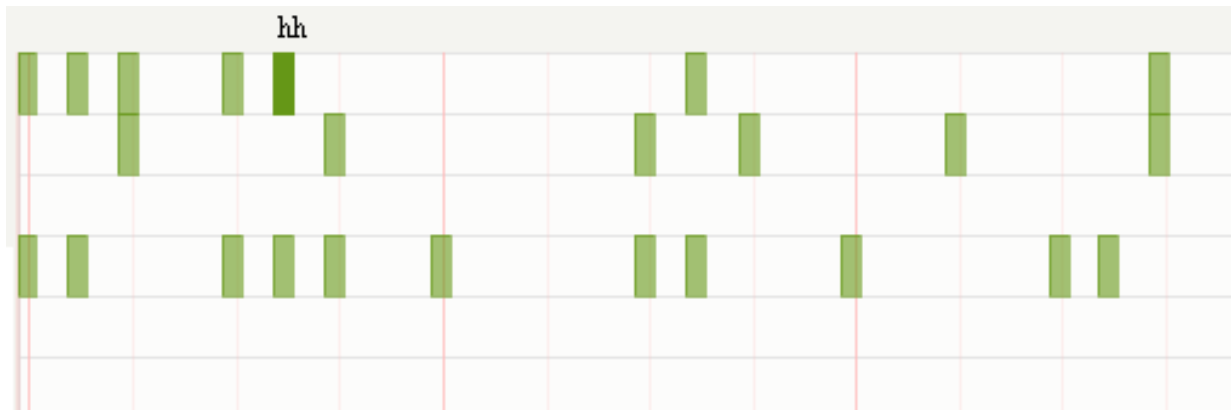
### 1.2.6 le fichier \_drums.xml

Ce fichier contient l'extraction sous forme d'éléments de batterie (BD : bass drum, LT : Low Tom, SD :Snare Drum, HH : High Hat). Le fichier a la syntaxe suivante :

```
<structuredescription generator="awk_script" version="1.0" date="2007/10">
  <description type="drumcodes">
    <segment start="0.494" drumcode="hh" />
    <segment start="1.075" drumcode="hh" />
    <segment start="1.458" drumcode="hh" />
    <segment start="1.589" drumcode="hh" />
    <segment start="1.841" drumcode="hh" />
    <segment start="2.146" drumcode="hh" />
    <segment start="2.410" drumcode="hh" />
    <segment start="2.685" drumcode="hh" />
    <segment start="3.074" drumcode="hh" />
    <segment start="3.224" drumcode="hh" />
    <segment start="3.487" drumcode="sd" />
    <segment start="3.487" drumcode="hh" />
    <segment start="3.613" drumcode="hh" />
    <segment start="3.762" drumcode="hh" />
    <segment start="3.882" drumcode="hh" />
    <segment start="4.014" drumcode="hh" />
    <segment start="4.295" drumcode="hh" />
    <segment start="4.565" drumcode="hh" />
  </description>
</structuredescription>
```

La représentation graphique correspondante reprend le paradigme visuel d'une programmation de patterns de boîte à rythmes.





le nom de l'élément représenté est affiché lorsque l'on pointe sur l'objet à la souris (ici : HH pour Hi Hat). La position des éléments suit un ordre logique de bas en haut (BD,LT,SD,HH) allant du grave vers l'aigu.

## 1.2.7 le fichier \_instr.xml

Ce fichier contient les extractions automatiques d'instruments de musique détectés. Son style est comparable au fichier \_drums.xml, sauf que les évènements ont aussi un temps de fin. Il a la forme suivante :

```
<structuredescription generator="awk_script" version="1.0" date="2007/10">
<description type="instrumentcodes">
<segment start="0.522426" stop="1.044875" instrumentcode="po"/>
<segment start="0.928776" stop="1.277075" instrumentcode="vc"/>
<segment start="1.149365" stop="2.089773" instrumentcode="vc"/>
<segment start="2.252313" stop="3.285601" instrumentcode="vc"/>
<segment start="2.832812" stop="4.504649" instrumentcode="po"/>
<segment start="2.937302" stop="3.401701" instrumentcode="vc"/>
<segment start="3.273991" stop="3.575850" instrumentcode="po"/>
<segment start="3.552630" stop="3.808050" instrumentcode="po"/>
<segment start="5.479887" stop="6.002336" instrumentcode="vc"/>
<segment start="20.770227" stop="21.153356" instrumentcode="bs"/>
<segment start="21.164966" stop="21.536485" instrumentcode="bs"/>
</description>
</structuredescription>
```

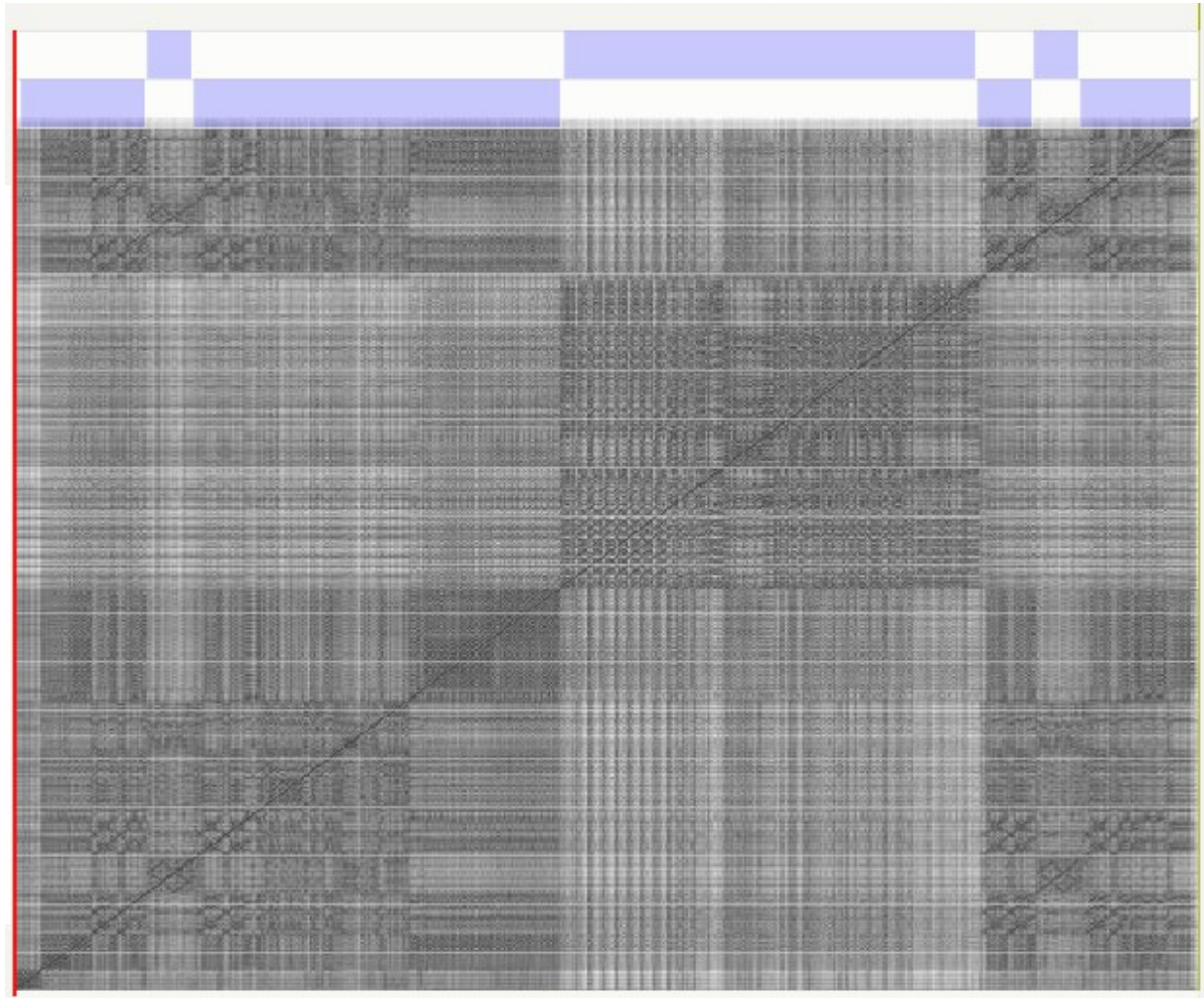
Les instruments sont représentés sous la forme de mnémoniques comme po (piano), vc (violoncelle), etc,etc. La représentation graphique de ce fichier dans l'interface est comme suit :



les blocs en bleu turquoise sont des objets cliquables (le nom de l'instrument apparaît en survolant l'objet avec la souris). Chaque ligne correspond à un instrument de musique détecté.

### 1.2.8 la matrice de similarité

Cette matrice de similarité peut être fournie à l'interface sous deux formats : un fichier JPG ou bien un binaire SDIF matriciel. La représentation graphique correspondante est comme suit :

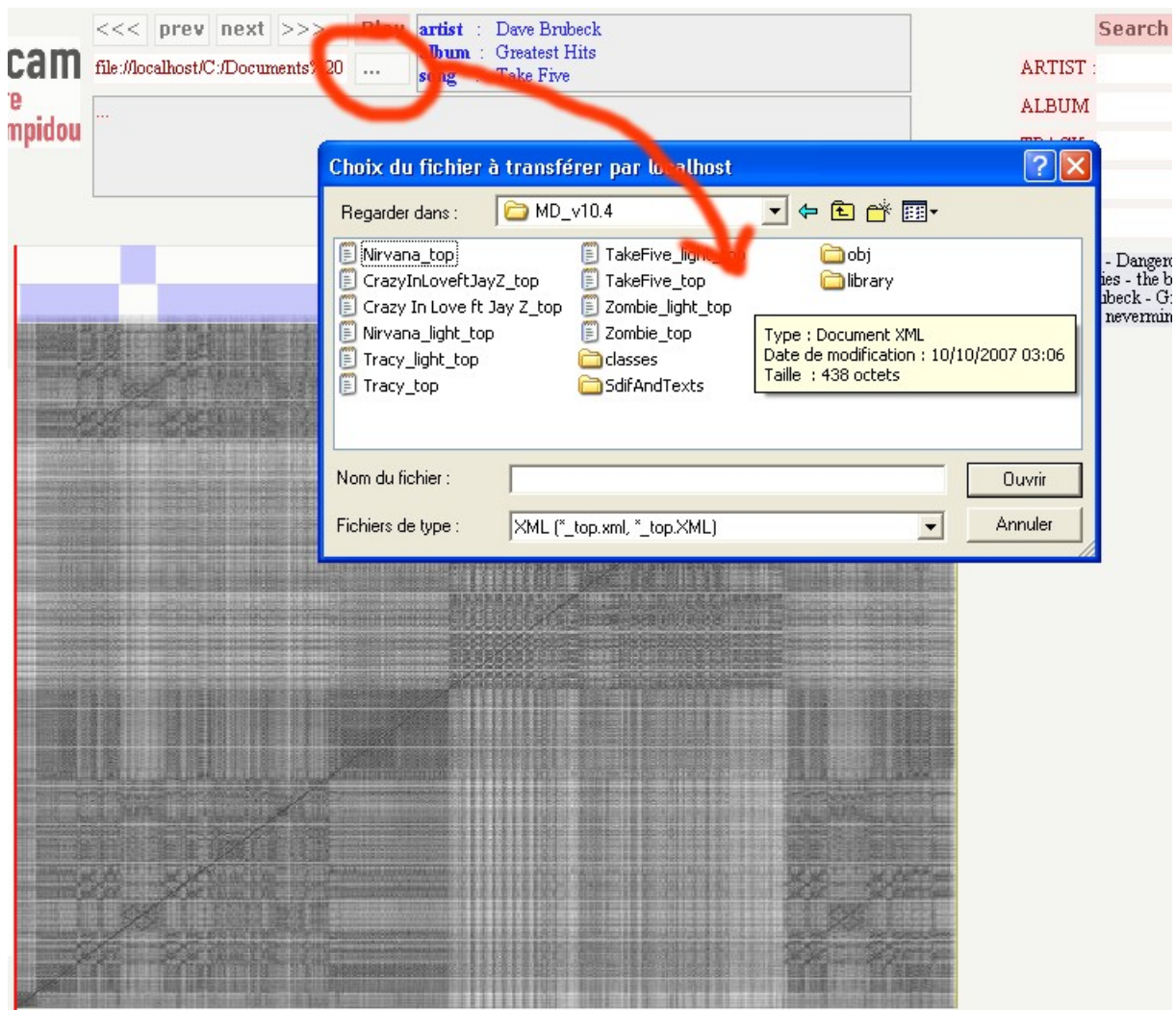


On voit sur l'image ci-dessus que cette matrice de similarité corrèle bien avec les informations de structure affichées pour ce morceau.

### 1.2.9 le fichier mp3

Les morceaux de musique lisibles dans cette interface doivent être des fichiers encodés en mp3 , avec un Bitrate Variable ou non. La seule contrainte importante est l'obligation de les tagger avec des tags ID3 v2.

### 1.3 Chargement manuel d'un morceau de musique



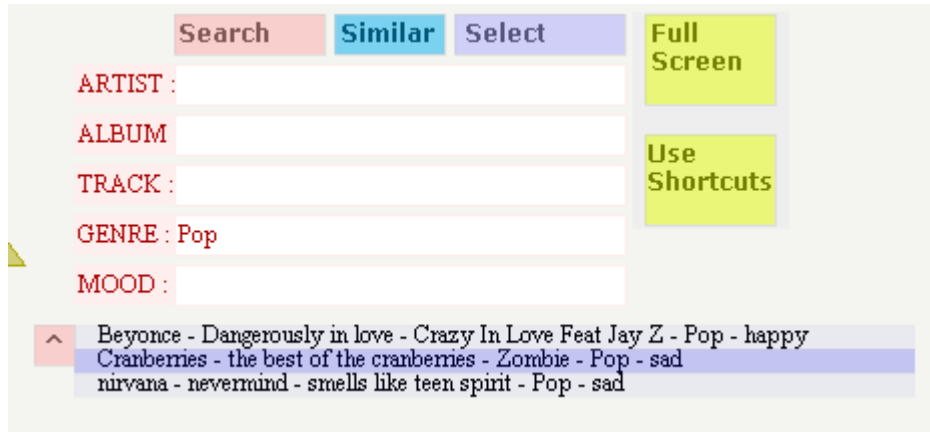
cette opération s'effectue en cliquant sur le bouton « ... » comme expliqué ci-dessus. Une fenêtre de dialogue permet de sélectionner le fichier `_top.xml` voulu. Après chargement, les différentes informations contenues dans les XMLs présents sont représentées. Si un XML manque, les éléments graphiques correspondants ne sont pas affichés (par exemple : il se peut qu'il n'y ait pas de batterie dans un morceau. On peut alors se dispenser simplement de fichier `_drums.xml`).

## 1.4 Base de donnée de titres

Le bouton de droite permet la recherche dans une base de données de titres.

### 1.4.1 recherche par critères

Entrer les critères comme auteur, titre, album, genre, humeur et une liste de résultats se crée.

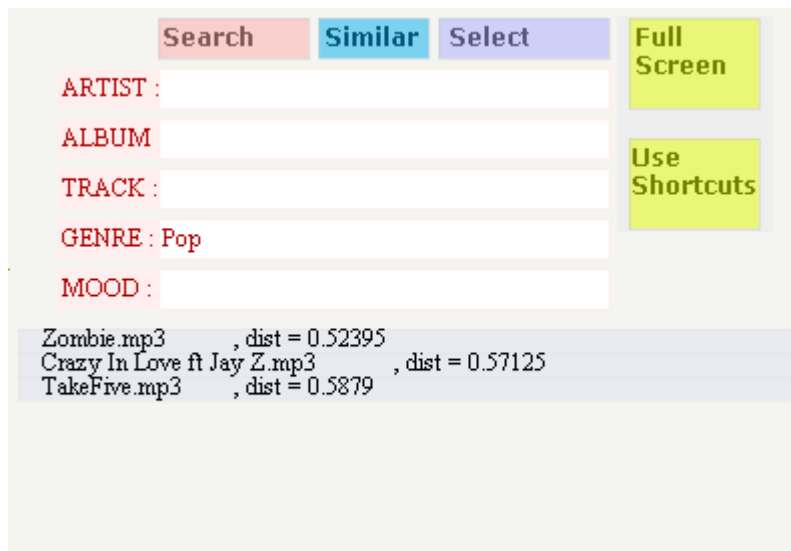


The screenshot shows a search interface with the following elements:

- Buttons: Search (red), Similar (blue), Select (purple), Full Screen (yellow), Use Shortcuts (yellow).
- Search criteria:
  - ARTIST :
  - ALBUM :
  - TRACK :
  - GENRE : Pop
  - MOOD :
- Results list:
  - ^ Beyonce - Dangerously in love - Crazy In Love Feat Jay Z - Pop - happy
  - Cranberries - the best of the cranberries - Zombie - Pop - sad
  - nirvana - nevermind - smells like teen spirit - Pop - sad

### 1.4.2 recherche par similarité

lorsqu'un titre est en train de jouer, cliquer sur similar permet de trouver des titres similaires sur la base de la tonalité, du timbre, du tempo et du rythme.



The screenshot shows a search interface with the following elements:

- Buttons: Search (red), Similar (blue), Select (purple), Full Screen (yellow), Use Shortcuts (yellow).
- Search criteria:
  - ARTIST :
  - ALBUM :
  - TRACK :
  - GENRE : Pop
  - MOOD :
- Results list:
  - Zombie.mp3 , dist = 0.52395
  - Crazy In Love ft Jay Z.mp3 , dist = 0.57125
  - TakeFive.mp3 , dist = 0.5879

le résultat est une liste de fichier avec leur distance respective envers le fichier couramment sélectionné.

