An open web audio platform for ethnomusicological sound archives management and automatic analysis

Thomas Fillon [1,2], Guillaume Pellerin[1], Paul Brossier[1], Joséphine Simonnot[3]

[1]Parisson, France
[2]LAM, Institut Jean Le Rond d'Alembert, UPMC Univ. Paris 06, UMR CNRS 7190, 11 rue de Lourmel, 75015 Paris, France
[3]CREM, LESC, UMR CNRS 7186, MAE, Université Paris Ouest Nanterre La Défense, 21 Allée de l'Université - 92023 Nanterre, France

This work was partially done inside the DIADEMS project funded by the French National Research Agency ANR (CONTINT)

Introduction

- Since 2007, ethnomusicologists from the *Center for Research in Ethnomusicology* (CREM) and engineers from Parisson have joined their efforts to develop *Telemeta*, a scalable and collaborative *open-source* web platform for management of and access to digital sound archives.
- A fully operational deployment of this platform is online since 2011 : « **Sound archives of the CNRS - Musée de l'Homme** » : `http://archives.crem-cnrs.fr`
- The design of Telemeta focuses on the enhanced and collaborative user-experience in accessing audio items and their associated metadata and on the possibility for the expert user to further enrich those metadata.
- It fits the professional requirements from both sound archivists and researchers in ethnomusicology.

`http://telemeta.org/`

Contact : {guillaume,thomas}@parisson.com

Telemeta architecture relies on *TimeSide*, an open audio processing framework written in Python which :

- Provides decoding, encoding and streaming methods for various formats together with a smart embeddable HTML audio player.
- Includes a set of audio analysis plugins and additionally wraps several audio features extraction libraries to provide automatic annotation, segmentation and musicological analysis

Web audio content management features and architecture

- *Telemeta* is a free and open source (CeCILL Free Software License Agreement) web audio content management system which introduces efficient and secure methods for backuping, indexing, transcoding, analysing and publishing any digitalized audio file with its metadata.
- *Telemeta* is ideal for professional collaborators who wants to easily organize, backup, archive and publish documented sound collections of audio files, CDs, digitalized vinyls and magnetic tapes over a strong database in accordance with open web standards.
- *Telemeta* architecture is flexible and can easily be adapted to particular database organization of a given sound archives.

The main features of *Telemeta* are :

- Pure HTML web user interface including high level search engine
- Smart workflow management with contextual user lists, profiles and rights
- Model-View-Controller (MVC) architecture
- Strong Structured Query Language (SQL) or Oracle backend

Beside database management, the audio support is mainly provided through an external component, *TimeSide*.

Metadata

- In addition to the audio data, dynamically handling metadata in a collaborative manner optimises the continuous process of knowledge gathering and enrichment of the materials in the database.
- Interoperability : integration of the metadata standards protocols Dublin Core and OAI-PMH (Open Archives Initiative Protocol for Metadata Harvesting) [3, 5].

Contextual Information

In ethnomusicology, contextual information could be geographic, cultural and musical. It could also store archive related information and include related materials in any multimedia format.

Annotations and segmentation

Metadata also consist in temporal information such as :

- a list of time-coded markers associated with annotations
- a list of of time-segments associated with labels (*in development*) .

The ontology for those labels is relevant for ethnomusicology (e.g. speech versus singing voice segment, chorus, ...). It should be noted that annotations and segmentation can be done either by a human expert or by some automatic signal processing analysis.

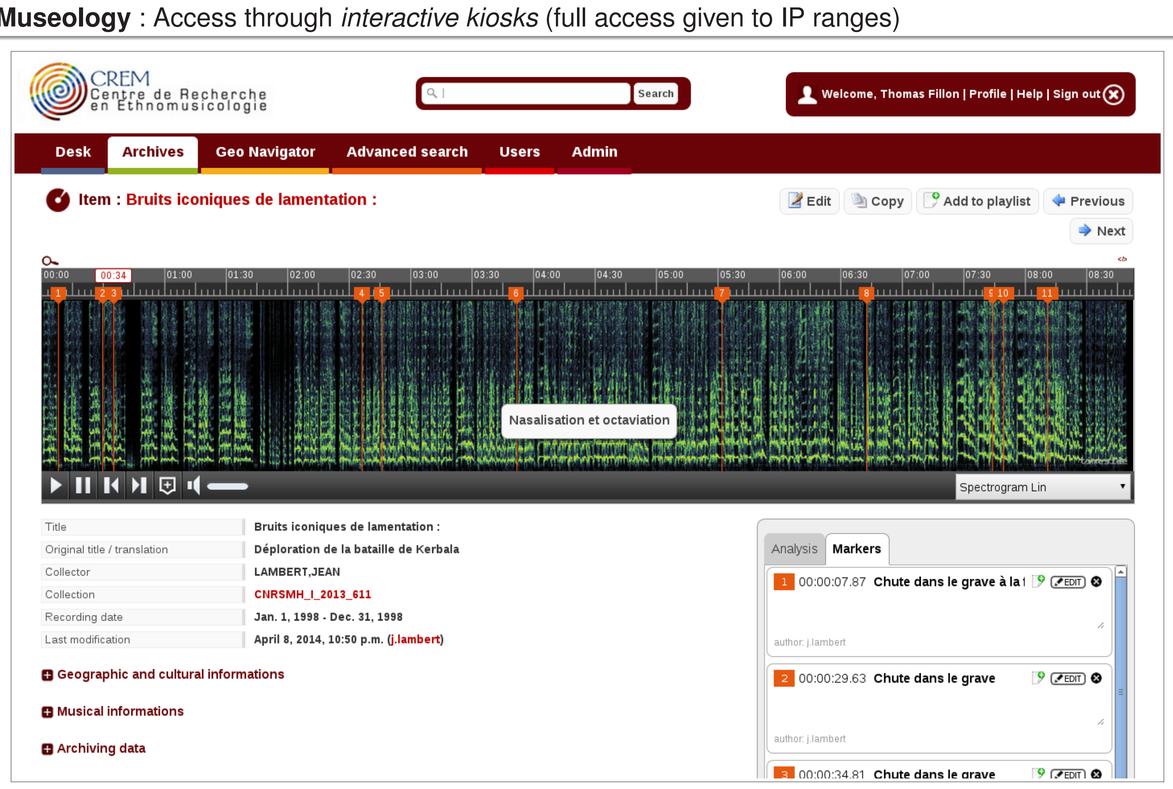
Sound archives of the CNRS - Musée de l'Homme

The ressources are available to researchers and to the extent possible, the public, in compliance with the intellectual and moral rights of musicians and collectors. These archives are the most important in Europe :

- Nearly 3,500 hours of recordings of unpublished field.
- Approximately 3700 hours of material published (more than 5000 discs, many of which are very rare).

Usages

- **Research** :
  - Publish collected ressources together with research work.
  - Exchange data online with other researchers or communities producing their music in their home countries through collaborative tools like markers and comments.
- **Teaching** : Ressources for teachers in *anthropology* or *ethnomusicology* as it provides the students an access to materials for lessons, academic works and exams.
- **Museology** : Access through *interactive kiosks* (full access given to IP ranges)



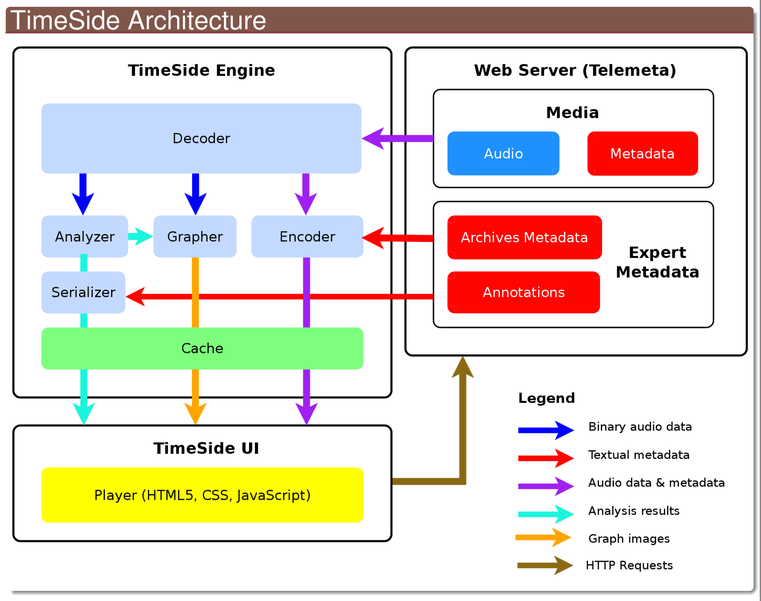
TimeSide : open web audio processing framework

`https://github.com/yomguy/TimeSide/`

Functionality

- Do asynchronous and fast audio processing with Python.
- Decode ANY audio or video format into numpy arrays thanks to Gstreamer.
- Analyze audio content with some external audio feature extraction libraries.
- Organize, serialize and save analysis metadata through various formats.
- Draw various waveforms, spectrograms and other representations from audio analysis.
- Transcode audio data in various media formats and stream them through web apps.
- Playback, index, tag and interact on demand with a smart high-level HTML5 extensible player.

TimeSide Architecture



Audio features extraction

TimeSide incorporates some state-of-the-art audio feature extraction libraries such as :

- **Aubio :** `http://aubio.org` [2]
- **Yaafe :** `http://yaafe.sourceforge.net` [4]
- **Vamp plugins :** `http://www.vamp-plugins.org` [1]

Given the extracted features, every sound item in a given collection can be automatically analyze. The results of this analysis can be displayed as a support to ethnomusicological studies.

Code Example (Python)

```
import timeside
from timeside.core import get_processor

# Define some processors:
file_decoder = get_processor('gst_dec')('sweep.wav')
analyzer = get_processor('level')()
grapher = get_processor('spectrogram_log')()
encoder = get_processor('gst_vorbis_enc')('sweep.ogg')

# Then, the magic pipeline:
(file_decoder | analyzer | grapher | encoder).run()

# Get the results:
grapher.render(output='image.png')
for key in analyzer.results.keys():
    print '%s in %s : %s' % (analyzer.results[key].name,
                             analyzer.results[key].unit,
                             analyzer.results[key].data)
```

Results

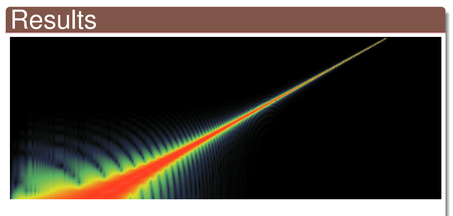


FIGURE: Spectrogram (sweep signal)

```
Level Analyzer Max:[-6.021]
Level Analyzer RMS:[-9.856]
```

Ongoing developments

- Annotate multimedia items by time segments supporting both free annotations and ontology-based annotations
- Efficiently visualize results from analysis from various data types together with X-Y zoom capability and audio synchronization by using state-of-the-art multimedia JavaScript libraries : *D3.js*, JQueryUI and audio libraries from the Wave project (`http://wave.ircam.fr/`)
- Design, manage and run analyses on large audio corpus and serialize the results through a REST web API with the TimeSide server
- Increase the analysis functionality with various automatic analysis and annotation tools for speech, audio, Music Information Retrieval and ethnomusicology (DIADEMS project).

Références

[1] The Vamp audio analysis plugin system. `http://www.vamp-plugins.org`.

[2] Paul Brossier. *Automatic annotation of musical audio for interactive systems*. PhD thesis, Centre for Digital music, Queen Mary University of London, UK, 2006.

[3] Dublin Core metadata initiative. `http://dublincore.org/`.

[4] Benoît Mathieu, Slim Essid, Thomas Fillon, Jacques Prado, and Gaël Richard. Yaafe, an easy to use and efficient audio feature extraction software. In *Proc. of ISMIR 2010, Utrecht, Netherlands*, pages 441–446. International Society for Music Information Retrieval, 2010.

[5] Open archives initiative protocol for metadata harvesting. `http://www.openarchives.org/pmh/`.