TELEMETA, audio web Content Management System for ethnomusicological sound archives

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)

Abstract

- *Telemeta* is an open-source audio web Content Management System (CMS) dedicated to digital sound archives secure storing, indexing and publishing.
- The demonstration presents the features of this platform in the context of *ethnomusicological* research.
- It focuses on the enhance 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.
- *Telemeta* also provides integrated audio signal processing tools for automatic analysis of sound items.

KEYWORDS : Sound archives, Metadata, Ethnomusicology, Database, Audio labelling, Web platform

telemeta
http://telemeta.org/

Contact : guillaume@parisson.com

Introduction

Needs

- In social sciences like anthropology and linguistics, researchers have to work on multiple types of multimedia documents such as photos, videos, sound recordings or databases.
- The need to easily access, visualize and annotate such materials can be problematic given their diverse formats, sources and given their chronological nature.

The *Telemeta* project

- The CREM laboratory and Parisson have been developing an innovative, collaborative and interdisciplinary open-source web-based multimedia platform since 2007.
- Goal : fit the professional requirements from both sound archivists and researchers in ethnomusicology.
- Official Platform online since 2011 : *Archives sonores du CNRS, Musée de l'Homme* :

`http://archives.crem-cnrs.fr`

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, an efficient and dynamic management of the associated metadata is also required.
- 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) [2, 4].

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.

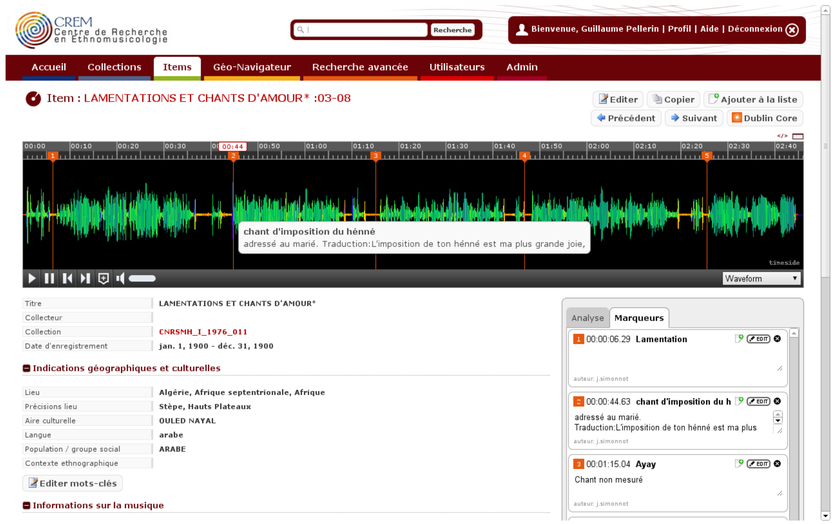
Annotations and segmentation

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.

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.

Telemeta web interface



TimeSide : open web audio processing framework

One specificity of the *Telemeta* architecture is to rely on an external component, *TimeSide*, that offers audio player integration together with low and high level audio signal processing capabilities.

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

Goals

- 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 fancy waveforms, spectrograms and other cool graphers.
- 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.

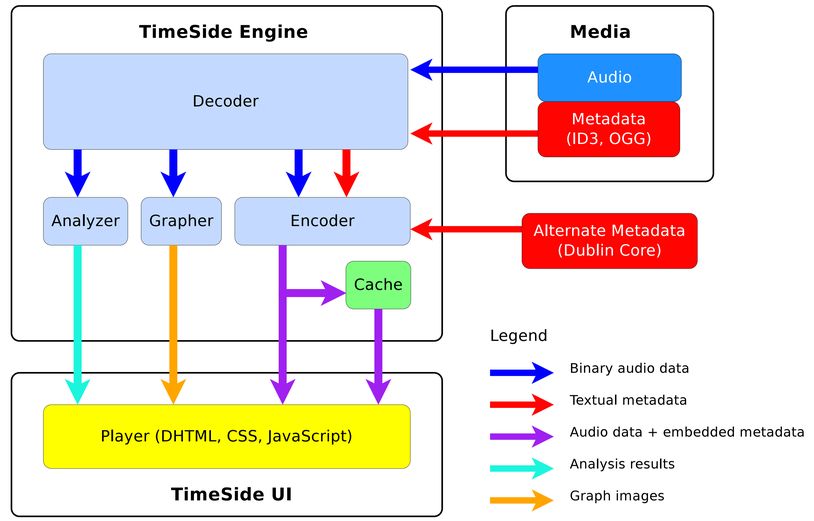
Audio features extraction

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

- **Aubio :** `http://aubio.org` [1]
- **Yaafe :** `http://yaafe.sourceforge.net` [3]
- **Vamp plugins :** `http://www.vamp-plugins.org` [5]

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. Further works lead by the DIADEMS project will incorporate advance Music Information Retrieval methods in order to provide automatic annotation, segmentation and similarity analysis.

TimeSide Architecture



Code Example (Python)

```
import timeside

# Define some processors:
decoder = timeside.decoder.FileDecoder('sweep.wav')
analyzer = timeside.analyzer.Level()
grapher = timeside.grapher.Spectrogram()
encoder = timeside.encoder.VorbisEncoder('sweep.ogg')

# Then, the magic pipeline:
(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

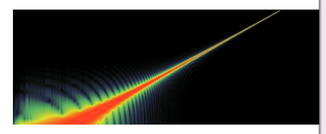


FIGURE: Spectrogram (sweep signal)

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

Références

[1] Brossier, P. : Automatic annotation of musical audio for interactive systems. Ph.D. thesis, Centre for Digital music, Queen Mary University of London, UK (2006)

[2] Dublin Core metadata initiative, http://dublincore.org/

[3] Mathieu, B., Essid, S., Fillon, T., Prado, J., Richard, G. : Yaafe, an easy to use and efficient audio feature extraction software. In : Proc. of ISMIR 2010, Utrecht, Netherlands. pp. 441–446. International Society for Music Information Retrieval (2010), http://ismir2010.ismir.net/proceedings/ismir2010-75.pdf

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

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