

# Projet Telemeta - Transfert de technologie

Guillaume Pellerin (Parisson)

08/12/2011 - TGE Adonis



- 1 Introduction
  - Présentations
  - Objectifs
  - Historique
- 2 Présentation du logiciel
  - Technologies et communautés
  - Modèle de données
  - Interface web
  - Droits d'accès
  - Traitement et lecture audio
  - Moissonnage et sauvegarde
  - Roadmap
- 3 Installation
- 4 Développement
  - Contexte
  - Modélisation
  - Contrôleurs
  - Vues
- 5 Conclusion et perspectives

- 1 Introduction
  - Présentations
  - Objectifs
  - Historique
- 2 Présentation du logiciel
  - Technologies et communautés
  - Modèle de données
  - Interface web
  - Droits d'accès
  - Traitement et lecture audio
  - Moissonnage et sauvegarde
  - Roadmap
- 3 Installation
- 4 Développement
  - Contexte
  - Modélisation
  - Contrôleurs
  - Vues
- 5 Conclusion et perspectives

# Parisson : solutions média sur-mesure

Les domaines de recherche et développement :

- **Captation et enregistrement** : captation multi-canal audio et vidéo, enregistrements et diffusions automatisés (projet TeleCaster)
- **Traitement des signaux** : analyse multi-dimensionnelle de signaux musicaux et vocaux, indexation semi-automatique (projets TimeSide, GStreamer)
- **Architectures, réseaux et base de données** : gestion des archives numériques et des métadonnées structurées, flux de métadonnées, streaming en temps-réel (Projets Telemeta, DeeFuzzer)
- **Interfaces** : définition et production des outils de navigation dans les flux de données (Technologies AJAX, JSON, CSS3, HTML5)

# Parisson : solutions média sur-mesure

## Les domaines d'application :

- chaînes de production multimédia
- télé-conférence, e-learning
- archivage pérenne sémantique
- indexation collaborative
- gestion et hébergement de fonds (laboratoires, web radios, services audio)

**Clients** : Laboratoires, musées, médiathèques, centres de formation, industrie musicale

**Mots clés** : audio, vidéo, métadonnées, acoustique, enregistrement, transcodage, diffusion, indexation, analyse audio, archivage pérenne, broadcast, streaming, e-learning, réseaux sociaux.

**Contexte de développement** : 100% open source, logiciels libres, Linux, Python, Debian.

# Le Centre de Recherche en Ethnomusicologie

Un fond ethnomusicologique unique :

- 4291 collections (inédit et édité)
- Actuellement 22101 fiches documentaires comportant 50 champs en moyenne (60000 environ à terme)
- 3250 heures de sons inédits soit 4 To environ + montages et copies diverses
- 3500 heures de sons édités soit 4,5 To environ
- 300 heures environ de video ( $\approx$  4 To)

# 1 Introduction

- Présentations
- Objectifs
- Historique

# 2 Présentation du logiciel

- Technologies et communautés
- Modèle de données
- Interface web
- Droits d'accès
- Traitement et lecture audio
- Moissonnage et sauvegarde
- Roadmap

# 3 Installation

# 4 Développement

- Contexte
- Modélisation
- Contrôleurs
- Vues

# 5 Conclusion et perspectives

# Telemeta : open web audio CMS

Objectifs du projet :

- **Pérenniser** les archives audionumériques (logiciels et formats)
- **Valoriser** le patrimoine culturel par la consultation légale
- Faciliter et **optimiser** la transmission des méta-données
- Augmenter les capacités de recherche et de travail collaboratif (saisie en ligne, **géo-localisation**, **web sémantique**, **interopérabilité**, croisement de données)
- Définir une ergonomie d'indexation et de **publication collaborative**, un principe de sauvegarde des oeuvres musicales



# 1 Introduction

- Présentations
- Objectifs
- Historique

# 2 Présentation du logiciel

- Technologies et communautés
- Modèle de données
- Interface web
- Droits d'accès
- Traitement et lecture audio
- Moissonnage et sauvegarde
- Roadmap

# 3 Installation

# 4 Développement

- Contexte
- Modélisation
- Contrôleurs
- Vues

# 5 Conclusion et perspectives

# Historique

- 2007 : rencontre du CREM avec Parisson et le LAM, contour des besoins mutuels, soutiens du projet Anthroponet
- 2008 : définition du cahier des charges, choix du framework, modélisation, migrateur 4D > MySQL
- 2009 : prototypage (v0.5), recherche de partenaires
- 2010 : corrections, développement agile (v0.9), soutien du TGE Adonis
- 2011 : migration définitive, v1.0 déployée sur serveur, mise en production



# Partenariats

## Partenaires participants :

- Centre de Recherche en Ethnomusicologie (**CREM**) du Laboratoire d'Ethnologie et de Sociologie Comparée (**LESC**), UMR 7186
- Equipe Lutheries, Acoustique et Musique (**LAM**) de l'Institut Jean le Rond d'Alembert (**IJLRA**), UMR 7190
- Médiathèque Eric-de-Dampierre de la **MAE**, Nanterre
- Museum National d'Histoire Naturelle (**MNHN**)
- Institut de Recherche et d'Innovation (**IRI**)
- Musée des Civilisations de l'Europe de la Méditerranée (**MuCEM**)
- Phonothèque de la Maison Méditerranéenne des Sciences de l'Homme (**MMSH**)

## Partenaires potentiels :

- Bibliothèque Nationale de France (**BNF**)
- Institut National de l'Audiovisuel (**INA**)
- Queen Mary University (**QMU**, Londres)

- 1 Introduction
  - Présentations
  - Objectifs
  - Historique
- 2 Présentation du logiciel
  - Technologies et communautés
  - Modèle de données
  - Interface web
  - Droits d'accès
  - Traitement et lecture audio
  - Moissonnage et sauvegarde
  - Roadmap
- 3 Installation
- 4 Développement
  - Contexte
  - Modélisation
  - Contrôleurs
  - Vues
- 5 Conclusion et perspectives

# Telemeta : un projet libre et ouvert

## Les fondamentaux du logiciel libre :

- Pérenniser les ressources informatiques
- Dynamiser le développement (partage, communautés internationales)
- Limiter les coûts de déploiement à grande échelle

## Briques 100% Open Source :

- Python et Django : langage côté serveur et framework web
- TimeSide : traitement des signaux audio, lecteur audio dynamique
- MySQL : base de données relationnelle
- Linux : noyau serveur
- CeCILL : licence libre conforme au droit français (CNRS, INRIA, CEA)

# Les standards et normes utilisés par Telemeta

- Web :
  - HTML5 : langage hypertextuel avec balises <audio> <video>
  - CSS : styles
  - JavaScript : langage côté navigateur (interfaces et lecteur dynamiques)
- Audio :
  - WAV : archivage audio brut
  - MP3, MP4 : compression avec pertes, largement utilisé, encapsulation partielle
  - OGG Vorbis : compression avec pertes, open source, encapsulation totale
  - FLAC : compression sans pertes, multi-pistes, open source, encapsulation totale
- Métadonnées :
  - XML (W3C)
  - DublinCore (OAI-PMH)
  - SQL : base de données

# Plateforme communautaire de développement

[telemeta.org](http://telemeta.org)

- Présentation, installation, documentation
- Blog, tickets, bugs
- Mailing list
- Wiki

[code.google.com/p/timeside](http://code.google.com/p/timeside)

- Présentation, installation, exemples, documentation
- tickets, bugs
- Wiki

# L'architecture de Telemeta

## Contrôleur

Le contrôleur reçoit les requêtes des utilisateurs, administrateurs, et systèmes distants, en général via HTTP, et met en oeuvre le Modèle, et la ou les Vue(s) adéquate(s).

**Vue  
HTML**

**Vue  
Dublin  
Core**

**Vue  
d'export**

**Vue de  
sauvegarde**

## Vues

Il y a plusieurs Vues, pour la génération HTML, les sauvegardes XML, les exports OGG/MP3, etc.. à partir du Modèle, comme requis par le Contrôleur.

## Modèle

Le Modèle fournit une couche d'abstraction des données, à travers une API bien conçue, pour les méta données et fichiers audio.

## Système de Fichiers

Stocke les fichiers audio WAV sources

## Base de données relationnelle

Stocke les méta données



# 1 Introduction

- Présentations
- Objectifs
- Historique

# 2 Présentation du logiciel

- Technologies et communautés
- **Modèle de données**
- Interface web
- Droits d'accès
- Traitement et lecture audio
- Moissonnage et sauvegarde
- Roadmap

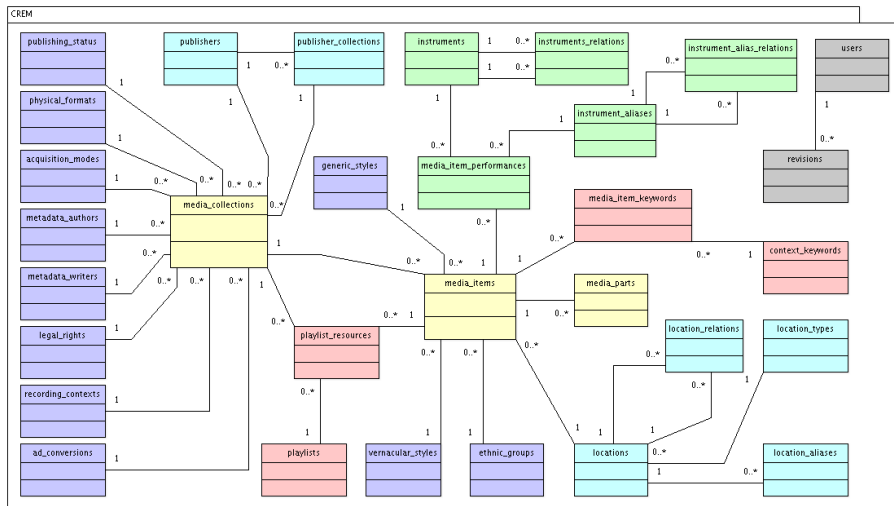
# 3 Installation

# 4 Développement

- Contexte
- Modélisation
- Contrôleurs
- Vues

# 5 Conclusion et perspectives

# Modèle de données du CREM



# 1 Introduction

- Présentations
- Objectifs
- Historique

# 2 Présentation du logiciel

- Technologies et communautés
- Modèle de données
- **Interface web**
- Droits d'accès
- Traitement et lecture audio
- Moissonnage et sauvegarde
- Roadmap

# 3 Installation

# 4 Développement

- Contexte
- Modélisation
- Contrôleurs
- Vues

# 5 Conclusion et perspectives

# Edition des objets

- Saisie documentaire
- Collections, Items
- Enumérations (Thésaurus)
- Lieux
- Instruments
- Compositions
- Mots clés
- Marqueurs temporels

## 1 Introduction

- Présentations
- Objectifs
- Historique

## 2 Présentation du logiciel

- Technologies et communautés
- Modèle de données
- Interface web
- **Droits d'accès**
- Traitement et lecture audio
- Moissonnage et sauvegarde
- Roadmap

## 3 Installation

## 4 Développement

- Contexte
- Modélisation
- Contrôleurs
- Vues

## 5 Conclusion et perspectives

# Droits d'accès

- **Utilisateurs** : profils, droits, gestion des mots de passe
- **Groupe**s : administrateur, documentaliste, chercheur, membre, anonyme
- **Droits** : ajouter, supprimer ou modifier les objets selon les règles
- Pour tous les utilisateurs : **listes de lecture** personnelle, ajout de **marqueurs temporels**
- **Droits particuliers** pour la lecture audio (date glissante de 50 ans, paramètre “public access”)

# Détail des droits des groupes

- **Anonyme** : parcours du site, lecture selon les autorisations de chaque objet
- **Membre** : ajout de listes de lectures personnelles, ajout de marqueurs
- **Chercheur** : ajout et édition des fiches documentaires
- **Documentaliste** : suppression d'objets, tous droits sur la base de données (hors utilisateurs), lecture audio de tous les items
- **Administrateur** : ajouter des utilisateurs

+ droits spéciaux au cas par cas :

- Lecture audio de tous les items
- Téléchargement de tous les items

## 1 Introduction

- Présentations
- Objectifs
- Historique

## 2 Présentation du logiciel

- Technologies et communautés
- Modèle de données
- Interface web
- Droits d'accès
- **Traitement et lecture audio**
- Moissonnage et sauvegarde
- Roadmap

## 3 Installation

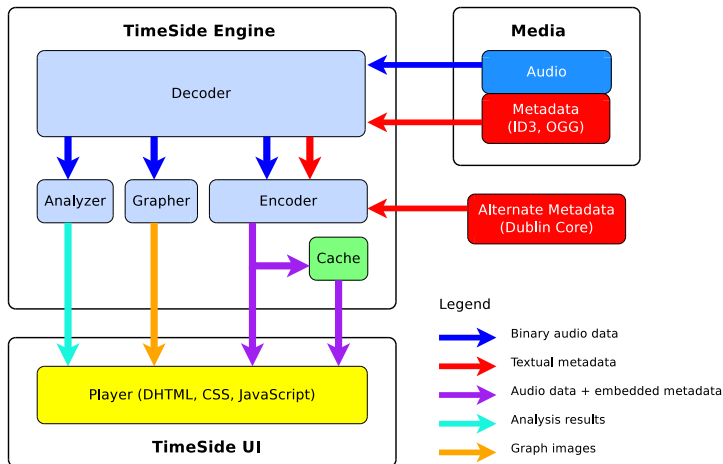
## 4 Développement

- Contexte
- Modélisation
- Contrôleurs
- Vues

## 5 Conclusion et perspectives



# TimeSide : Web Audio Components



# TimeSide UI : lecteur audio dynamique (R. Zaccarelli)

- Lecture audio
- Affichage audio
- Indexation temporelle
- Portabilité
- Principes du modèle de développement
- [code.google.com/p/timeside/wiki/UiGuide](http://code.google.com/p/timeside/wiki/UiGuide)

- 1 Introduction
  - Présentations
  - Objectifs
  - Historique
- 2 Présentation du logiciel
  - Technologies et communautés
  - Modèle de données
  - Interface web
  - Droits d'accès
  - Traitement et lecture audio
  - **Moissonnage et sauvegarde**
  - Roadmap
- 3 Installation
- 4 Développement
  - Contexte
  - Modélisation
  - Contrôleurs
  - Vues
- 5 Conclusion et perspectives

# Moissonnage et sauvegarde

- mapping Dublin Core
- serveur OAI-PMH intégré
- flux RSS (revisions)
- sauvegarde sérialisée : XML + WAV
- upload accumulatif

- 1 Introduction
  - Présentations
  - Objectifs
  - Historique
- 2 Présentation du logiciel
  - Technologies et communautés
  - Modèle de données
  - Interface web
  - Droits d'accès
  - Traitement et lecture audio
  - Moissonnage et sauvegarde
  - Roadmap
- 3 Installation
- 4 Développement
  - Contexte
  - Modélisation
  - Contrôleurs
  - Vues
- 5 Conclusion et perspectives

# Roadmap

Objectifs pour Telemeta 1.x :

- **Documents attachés** (images, textes, vidéos, etc...)
- **Compatibilité HTML5** totale du lecteur audio
- **Accès spéciaux** par adresses IP
- Intégration et MAJ des **ontologies** (langues, géographie)

Objectifs pour Telemeta 2.x :

- **Modèle générique** de données
- **Fonctions d'analyse** augmentée (reconnaissance, recoupement statistique) : ANR CONTINT 2011 **DIADEMS** et ANR CORPUS 2011 **DicA2Ref**
- **Déploiement mutualisé** (CNRS IN2P3)
- Définition d'une **API**

## 1 Introduction

- Présentations
- Objectifs
- Historique

## 2 Présentation du logiciel

- Technologies et communautés
- Modèle de données
- Interface web
- Droits d'accès
- Traitement et lecture audio
- Moissonnage et sauvegarde
- Roadmap

## 3 Installation

## 4 Développement

- Contexte
- Modélisation
- Contrôleurs
- Vues

## 5 Conclusion et perspectives

# Installation de Telemeta

- voir INSTALL.rst
- exemple !



## 1 Introduction

- Présentations
- Objectifs
- Historique

## 2 Présentation du logiciel

- Technologies et communautés
- Modèle de données
- Interface web
- Droits d'accès
- Traitement et lecture audio
- Moissonnage et sauvegarde
- Roadmap

## 3 Installation

## 4 Développement

- Contexte
- Modélisation
- Contrôleurs
- Vues

## 5 Conclusion et perspectives

# Contexte de développement

- utilisation de Git (versionnement décentralisé)
- dépôts (vcs.parisson.com, vcs.parisson.org, github, vous!)  
`git clone http://vcs.parisson.com/git/telemeta.git`
- branches (crem, lam, generic, production, video, etc...)
- wikis : <http://telemeta.org>

## 1 Introduction

- Présentations
- Objectifs
- Historique

## 2 Présentation du logiciel

- Technologies et communautés
- Modèle de données
- Interface web
- Droits d'accès
- Traitement et lecture audio
- Moissonnage et sauvegarde
- Roadmap

## 3 Installation

## 4 Développement

- Contexte
- **Modélisation**
- Contrôleurs
- Vues

## 5 Conclusion et perspectives

# Modélisation

Voir models/

## 1 Introduction

- Présentations
- Objectifs
- Historique

## 2 Présentation du logiciel

- Technologies et communautés
- Modèle de données
- Interface web
- Droits d'accès
- Traitement et lecture audio
- Moissonnage et sauvegarde
- Roadmap

## 3 Installation

## 4 Développement

- Contexte
- Modélisation
- **Controlleurs**
- Vues

## 5 Conclusion et perspectives

# Controlleurs

Voir web/ et url.py

- 1 Introduction
  - Présentations
  - Objectifs
  - Historique
- 2 Présentation du logiciel
  - Technologies et communautés
  - Modèle de données
  - Interface web
  - Droits d'accès
  - Traitement et lecture audio
  - Moissonnage et sauvegarde
  - Roadmap
- 3 Installation
- 4 Développement
  - Contexte
  - Modélisation
  - Contrôleurs
  - Vues
- 5 Conclusion et perspectives

# Vues

Voir `templates/telemeta_default/`



# 1 Introduction

- Présentations
- Objectifs
- Historique

# 2 Présentation du logiciel

- Technologies et communautés
- Modèle de données
- Interface web
- Droits d'accès
- Traitement et lecture audio
- Moissonnage et sauvegarde
- Roadmap

# 3 Installation

# 4 Développement

- Contexte
- Modélisation
- Contrôleurs
- Vues

# 5 Conclusion et perspectives

# Conclusion et perspectives

- Technologie et ergonomie prometteuse pour la **sauvegarde et la valorisation** du patrimoine musical
- **Déploiement** et **pérennité** optimisés avec les briques open source
- **Intégration souple** de données métiers hétérogènes (sciences humaines et sciences informatiques)
- **Plateforme collaborative** à un niveau international
- Exemple au CREM du 18/05 au 25/06 : plus de **450** fiches en moyenne éditées par semaines !
- Plateforme de **développement** ouverte

# Merci !



[http ://telemeta.org](http://telemeta.org)

Telemeta 1.3 “Bell”

[http ://pypi.python.org/packages/source/T/Telemeta/Telemeta-1.3.tar.gz](http://pypi.python.org/packages/source/T/Telemeta/Telemeta-1.3.tar.gz)

Ce document est mise à disposition selon un [contrat Creative Commons](#)