GAMELAN: A Knowledge Management Approach for Digital Audio Production Workflows

Karim Barkati [1] and **Alain Bonardi** [2] and **Antoine Vincent** [3] and **Francis Rousseaux** [4]

Abstract. The ongoing french research project GAMELAN aims at demonstrating how knowledge management principles and technics could serve digital audio production workflows management, analysis and preservation. In this position paper, we present both the production stakes of such an approach and the technical and scientific knowledge strategies we are developing, through the coupling of both knowledge and process engineerings.

1 INTRODUCTION

GAMELAN[5] is an ongoing french research project, which name means "An environment for management and archival of digital music and audio". It aims at answering specific needs regarding digital audio production – like interoperability, reusability, preservation and digital rights management – while striving to settle knowledge management issues, combining knowledge engineering with process engineering.

1.1 Digital audio production stakes

From a social standpoint, the large and growing number of users of audio environments for personal or applied production makes this field one of the richest in evolution. However, the complexity of the production management is a well known effect in the community and is often described as an inconsistency between the tools used. Indeed, the industry provides more and more powerful tools but regardless of global usage: users combine multiple tools simultaneously or constantly alternating from one tool to another.

From a legal standpoint, there is a real problem of contents tracking, given their multiple uses or changes in production. Till now, audio production systems keep no operational track that would allow following up the rights associated with each element.

Thereby, the GAMELAN project goals spread on four levels:

Production Environments — Keep track of all actions, since the starting material to finished product; organize production elements (files, software) in structures included as components of the environment; formalize the knowledge generated during the process.

Preservation Strategies — Use the production environment as a platform for preservation; extract structures and knowledge to simplify future access to the environment; apply OAIS[6] methodologies, allowing reuse of the environment and its components.

Reuse of productions — Restructure the production elements with new objectives, adding other materials and editing the links and the overall structure (*repurposing*, back-catalog rework); use subsets of the environment to generate new environments and facilitate the process deconstruction and reconstruction for intentions analysis.

Digital rights management — Enable traceability of content on a production to manage user rights. Detect and warn for missing DRM information (artists name, location, person in charge of the production) during the production process itself, from *creation patterns* specification.

1.2 Use cases and technical functionalities

The technical goal of GAMELAN research project is to create a software environment (also called GAMELAN), integrating musical and sound production softwares, and able to fully describe the workflow, from source to final product. GAMELAN conforms to Open Source Initiative criteria, is free, and defines guidelines to allow the meta-environment to extract specific software information.

We elaborated three use cases relying on the different expertises of project partners.

CD production at EMI Music — On digital audio workstations ("DAWs", like ProTools or Audacity). CD production workflow involves recording, mixing and mastering steps. The main related test case consists in removing a particular track from a song for *repurposing* (like the voice for karaoke edit), the second one in identifying all contributors name to ease rights management.

Acousmatic music creation at INA/GRM — Also on DAWs. Workflow involves at least mixing and editing steps. The main test case consists in identifying which file is the "final mix" for archiving, the second one in identifying eventual versions varying only in format (compression, number of channels, etc.).

Patch programming at IRCAM — On audio programming environments like Max/MSP, for real-time interactive works. The main test case consists in visualizing structural changes of patches (subpatches and abstractions calls) for genetic analysis, the second one in indexing control parameters values for centralized fine-tuning.

As a meta-environment, GAMELAN traces data during the production process and utilizes formalized knowledge upon collected

[1] Institut de Recherche et Coordination Acoustique/Musique (IRCAM), France; email: karim.barkati@ircam.fr

[2] Paris 8 University, EA 1572 – CICM & IRCAM, France; email: alain.bonardi@ircam.fr

[3] Université de Technologie de Compiègne, Laboratoire Heudiasyc, UMR 7253 CNRS, France; email: antoine.vincent@hds.utc.fr

[4] Reims Champagne-Ardenne University, CReSTIC EA 3804 & IRCAM, France; email: francis.rousseaux@univ-reims.fr

[5] http://www.gamelan-projet.fr, "*Un environnement pour la Gestion et l'Archivage de la Musique Et de L'Audio Numériques*".

[6] Open Archive Information System.

data, both during and after production time. Main technical functionalities are tracing, acquisition, ingestion, reasoning, requesting, browsing, file genealogy visualisation, integrity and authority checking, and archiving.

1.3 Knowledge management issues

On the way to GAMELAN's goals, we identified three main aspects:

Archival issue — How to work out a representation of the musical objects that allows to exchange, take back or reproduce them, while each musical environment is most of the time particular and contingent? Furthermore, this representation has to be abstract in that it must be general enough to be valid and usable in other contexts, and concrete enough to contain the information necessary for the reuse of objects.

Cognitive issue — How to characterize and explain the part of the knowledge necessary to understand and reuse tools and their settings, while the creation process mobilizes a set of intentions and knowledge rom the author that are only implicitly transcribed in its use and settings? This information will be included in the abstract representation elaborated in the previous archival issue.

Technical issue — How to make explicit the knowledge about the creative process as well as production tools despite the heterogeneity of abstraction levels of objects and the fragmentation of objects and tools that do not communicate with each other? Moreover, the worked out representations and models should to be exploitable in a technical environment that offers the user the ability to interact with data and structures.

So, the GAMELAN project addresses issues relevant to several knowledge fields:

- Digital archiving through intelligent preservation;
- Management and capitalization of knowledge;
- Process engineering and knowledge engineering.

The goal here is to define a trace engineering, that is to say intermediate objects built by the composer or producer and associated settings. To reach this goal, we divided the global work into three main tasks presented in the paper:

- Production process tracking, that should be agnostic;
- Professional knowledge models, that may be hierarchical;
- Work and process representation, that should allow both visualizing, querying and editing.

2 PRODUCTION PROCESS TRACKING

The first step we consider deals with gathering data, through logging user interaction events and collecting contextual information.

2.1 The GAMELAN meta-environment

The applicative objective of the project is to create a meta-environment for music and audio production, capable of integrating any type of production software, and able to fully describe the workflow, from source to final product.

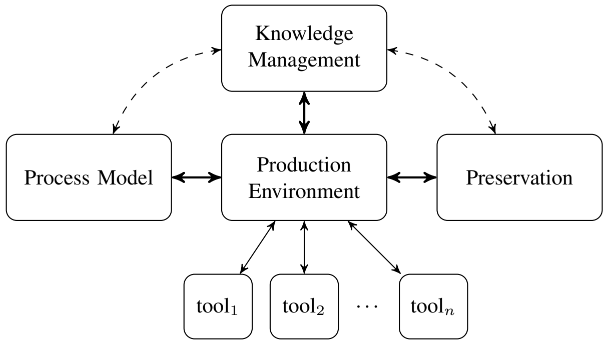

Figure 1. High-level technical architecture of GAMELAN.

2.1.1 High-level technical architecture

The technical architecture relies on the *production environment*, which includes various digital audio production tools at work in the process, as shown on Fig. 1.

It is based on predefined *process models* to measure and qualify the steps and operations performed during a particular process, related to a unit of *knowledge management* that provides methods for evaluating this process and provide at any time the user an evaluation of the current process and context sensitive help. Therefore, it aims at providing at all times an overview of the entire process in terms of progress, quality, and outcome.

Users should be able to control the interaction of this feedback with their own work, which implies non-intrusiveness and transparency for the meta-environment.

Finally, an *archive unit* will allow an smart preservation of digital objects, keeping the "footprint" of the entire process to allow full traceability. This unit will be based on the OAIS MustiCASPAR server developed within the CASPAR project [8], and adapted to the preservation of the production process.

2.1.2 Operational tracking

Considering the dynamic nature of knowledge is a key issue in knowledge engineering. Indeed, whatever the quality of the modeling process and the quantity of knowledge collection, resulting knowledge traces are then to be mobilized in contexts never completely predictable and these inscriptions will report a reality that has evolved by itself. This is the reason why we designed an operational tracking process as agnostic as possible, through messaging, tracing and logging.

The messaging part relies on an open-source standard commonly used in the computer music community, namely OSC[7], developed at UC Berkeley [15], which is a communication protocol for modern networking technology, with a client/server architecture (UDP and TCP).

In order to produce usage data [10, 13], we hacked open-source domain production softwares, like Audacity[8], to send a complete OSC message each time the user performs an action through the software, build with:

- Application name

[7] http://opensoundcontrol.org/, *Open-sound control*.
[8] http://audacity.sourceforge.net, an open source software for recording and editing sounds.

- Application version number
- Time stamp
- Function name
- Function parameters

We developed a tracing and logging application (the tracker) that both logs every message received during the production, only adding a reception time stamp, and keeps track of every version of modified files, tracking also file system messages, for file genealogy analysis and preservation purposes.

Hereafter are excerpts of log files, reduced to fit here (timestamps and/or other information are removed).

OSCMessages.txt

```
audacity 1.3 FileNew
audacity 1.3 FileSaveAs  test.aup
audacity 1.3 ImportAudio test.aup noise.wav
audacity 1.3 ImportAudio test.aup clicks.wav
audacity 1.3 Selection mix.aif "noise", "clicks"; Begin="1.9314"; End="10.0114"
audacity 1.3 ExportAudio test.aup mix.aif
audacity 1.3 FileClosed test.aup
```

CurrentApplication.txt

```
2012-07-09 10:09:36544633 +02 ApplicationActivated net.sourceforge.audacity
2012-07-09 10:09:36582045 +02 ApplicationActivated com.apple.dt.Xcode
2012-07-09 10:09:36593654 +02 ApplicationActivated com.apple.finder
```

FolderState.txt

```
folder-state 0      2012-07-10 16:22:58961547 +02
2012-01-20 18:07:65253000 +01      noise.wav
2012-01-20 18:07:65253000 +01      clicks.wav
...
folder-state 7      2012-07-10 16:23:58981517 +02
2012-01-20 18:07:65253000 +01      noise.wav
2012-01-20 18:07:65253000 +01      clicks.wav
2012-07-10 16:23:58980000 +02      test.aup
...
folder-state 21      2012-07-10 16:23:59005107 +02
2012-01-20 18:07:65253000 +01      noise.wav
2012-01-20 18:07:65253000 +01      clicks.wav
2012-07-10 16:23:59005000 +02      mix.aif
2012-07-10 16:23:58980000 +02      test.aup
```

2.1.3 Manual informing

Knowledge management as defined in our project requires further information that can not be inferred from the software activity logging. Indeed, a set of primary contextual information must be given by a human operator, like the user's name and the title of the work being produced. But a design dilemma immediately appears: on the one hand, the more contextual information feeds the system, the more informative might be the knowledge management, but on the other hand, the more a system asks a user to enter data, the more the user may reject the system [4].

In our case, the balance between quantity and quality of information has to be adjusted in a close relationship with the strongly-commited ontology we are incrementally developing with domain experts [14] and presented thereafter.

Temporal modalities have also to be anticipated in the information system, since the operational manual informing phase can be entered either at the same time that the production phase or temporally uncoupled, either by the producing user (e.g. a composer) or by an external agent (e.g. a secretary). Moreover, crucial missing data detection by the knowledge management system is a key feature of the project, as information integrity checking.

2.2 Managing knowledge flows

2.2.1 Ontology-driven KM

As we saw, the manual informing part of the system strongly depends of the domain ontology, but this is not the only part. Indeed, knowledge management depends on the ability to transform data and information into knowledge, according to Ackoff's model [1], and it turns out that ontologies are key tools in this transition process [9, 6]. We incrementally developed a strongly-commited differential ontology dedicated to audio production, dipping in productions with experts, in the OWL formalism.

In our system, except for the operational tracking that has to remain agnostic, the ontology drives all functional modules:

Data — The informer module we saw previously for contextual user data, especially for the entry interface design;

Information — The preprocessing module that prepares raw data (both usage data and user data) according to the ontology;

Knowledge — The semantic engine reasoning on the preprocessed information, and allowing requests;

Understanding — The browser module for data browsing and edition, and the viewer module that provides global graphical representations, like file genealogy trees.

The central position of the ontology comes from its semantic capabilities and justifies deep research toward professional knowledge modeling in music.

2.2.2 Production process tracing

We distinguish between *user data* and *usage data*. The former corresponds to the manual informing data and the latter to the automatic tracking data. In the computer music field, this production process tracing has never been done yet. We asked for use cases to domain professionals (cf. Section 1.2) in order to reproduce relevant user interaction with the production meta-environment.

This strategy aims several beneficiaries and time horizons:

In the immediate time of production — The composer, audio producer, may turn back its own work during the production, to explore various options or correct the undesirable consequences; it can be for example a selective "undo" instruction given to cancel an operation; it is also, for the composer or the sound engineer an opportunity to see and understand the overall work of composition or production.

In the intermediate time of collection — The composer, or the institution that manages its works, may return on a given work to recreate or reuse the content components.

In the long term preservation — The work becomes a memory and a relic, the challenge is to preserve the artistic and technical information to understand, interpret and re-perform.

3 PROFESSIONAL KNOWLEDGE MODEL

3.1 Modeling context

It is common to begin the modeling phase by a corpus analysis, usually from a collection of candidates-documents selected depending on their relevance [12]. But in our case study, we have no written document that can provide support to terms selection: vocabulary, and by extension, all production work relies on musical practices that are acquired more by experience than by teaching. Indeed, every musical work is a *prototype* in the sense of Elie During, as "the most perfect example, the more accurate", where each creation is an object "ideal and experimental": this uniqueness leads to a possibly infinite number of ways to create [5]. Thus, to achieve this essential phase of study, we design ourselves our corpus, which is rather unusual, by following several musical productions to find out invariants.

We do not seek to explain sound nor music (the *what*, like MusicXML kind of languages) but the way it is produced (the *how*), *i.e.* a formal language for audio production process. This language is devoted to the representation of what we might call the "musical level", referring to the "knowledge level" of Allen Newell: we want to represent the work at the right abstraction level, neither too concrete because too technology dependent and therefore highly subject to obsolescence, nor not enough because information would be too vague to be usable [11].

3.2 Main test cases

The GAMELAN project embraces various creative practices, related to the partners core business who defined three main test cases:

IRCAM *Recovery assistance and synthesis of information from one phase to another of a record.* We followed the recording and editing situation of the piece "Nuages gris" of Franz Liszt in the "Liszt as a Traveler" CD played by pianist Emmanuelle Swiercz. — Identify and represent the work of the sessions in two dimensions by time and by agent, all the events of one session (creation, update, export), and the dependencies of import and export files between sessions.

INA/GRM *Identification of files that have contributed to the final version of a work.* We log every DAW operation of composer Yann Geslin during the composition of a jingle. — Ensure that the file called "Final-Mixdown" is actually the one that produced the last audio files of the work; identify possible format changes (stereo, 8-channel, mp3); identify the intermediate versions.

EMI Music *Recovery and edit of past productions.* We plan to test the replacement of the drum from a recording made under GAMELAN. — Accurately identify which tracks to replay; substitute to an identified track for another; replay the final mix session with the replaced tracks.

3.3 Production process modeling

To create the representation language of the production process, we apply the *Archonte*[9] method of Bachimont [2].

Our production process modeling work followed three steps:

1. Normalization of the meanings of selected terms and classification in an ontological tree, specifying the relations of similarity between each concept and its father concept and/or brothers concepts: we then have a *differential* ontology;
2. Formalization of knowledge, adding properties to concepts or constraining relation fields, to obtain an *referential* ontology;
3. Operationalization in the representation language, in the form of a *computational* ontology.

After a phase of collection of our corpus and the selection of candidate terms, we took the first step in the form of a taxonomy of concepts, in which we strived to maintain a strong semantic commitment in supporting the principles of the differential semantics theory presented thereafter. This taxonomy has been performed iteratively, since it is dependent on our participation in various productions. Thus, at each new integration to the creation or the updating of a work, we flatten and question our taxonomy and term normalization, in order to verify that the semantic commitment is respected.

[9] ARCHitecture for ONTological Elaborating.

For incremental development and testing, we divided the ontology in two parts: the one as a model, with classes and properties, uploaded on a dedicated server `icm.ircam.fr`[10], the other as conform data sets, with individuals, uploaded on an OWL server (OpenRDF Sesame plus the OWLIM-Lite plugin) `gsemantic.ircam.fr`[11]. For common features, we import standard ontologies, like vCard[12] for standard identity information, so we will only detail the making of the domain ontology in this article.

3.4 The differential approach

In short, the differential approach for ontology elaboration systematically investigates the similarity and difference relations between each concept, its parent concept and its sibling concepts. So, in developing this structure, we tried to respect a strong ontological commitment by applying a *semantic normalization*, that is to say that for each concept, we ask the four differential questions of Table 1.

S. w/ P.	– Why does this concept inherit from its parent concept?
D. w/ P.	– Why is this concept different from its parent concept?
S. w/ S.	– Why is this concept similar to its sibling concepts?
D. w/ S.	– Why is this concept different from its sibling concepts?

Table 1. The four differential questions.

To realize practically this semantic normalization task, we used softwares DOE[13] [3] and Protégé[14], for both concepts and relations taxonomies building, refining and exporting (RDFS, OWL, etc.). At the end of this recursive process, we obtained a domain-specific differential ontology (see excerpt on Fig. 2), where the meaning of all terms have been normalized and that allows to develop the vocabulary needed for the next steps to reach the development of the representation language of the audio production process.

As a result of the differential method, domain vocabulary mostly lies in leaves of such an ontological tree: *work, performance, version; connection, graphical object, track, region; association, enterprise, institute; musical score, instrument, brass, strings, percussions, winds; effect box, synthesizer; file, session file, program; create, delete, edit; content import, content export; listen, play, work session, current selection*; etc. A large set of properties completes this domain ontology.

4 WORK AND PROCESS REPRESENTATIONS

The idea of such a meta-environment as GAMELAN, viewed as a trace-based system (cf. Fig. 3) meets clear needs in the community, as mentioned before. Moreover, while the operational meta-environment is still under development, our ontological work already points to the solution of various scientific challenges:

- Representation language for managing the creation process;
- Description language for representing the content of a work, in the diversity of its components;
- Integration of both languages in a single control environment.

[10] http://icm.ircam.fr/gamelan/ontology/2012/07/10/SoundProduction.owl
[11] http://gsemantic.ircam.fr:8080/openrdf-workbench/repositories/
[12] http://www.w3.org/Submission/vcard-rdf
[13] http://www.eurecom.fr/~troncy/DOE/, *Differential Ontology Editor*.
[14] http://protege.stanford.edu/

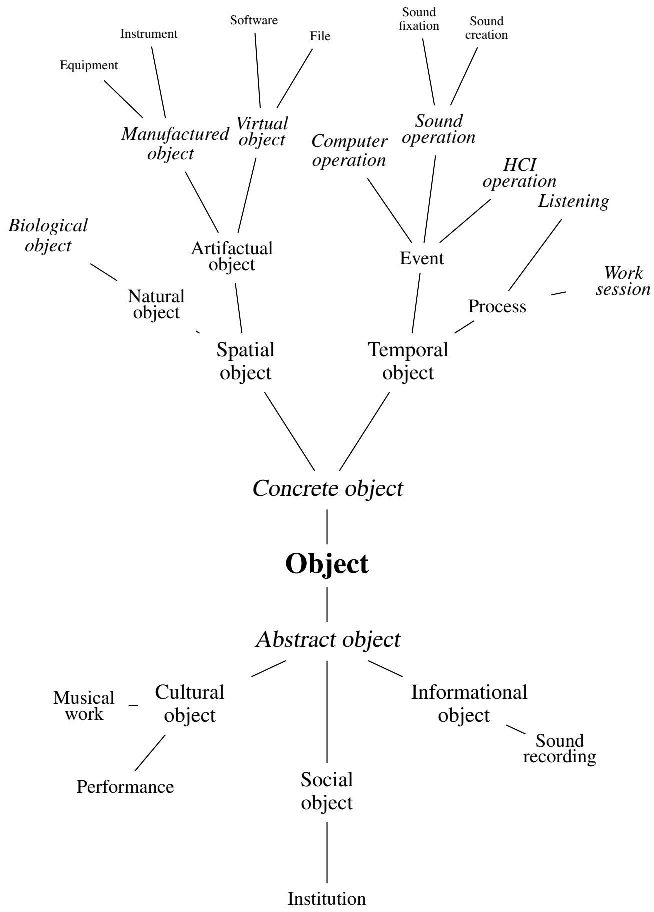

Figure 2. Excerpt from the differential taxonomy

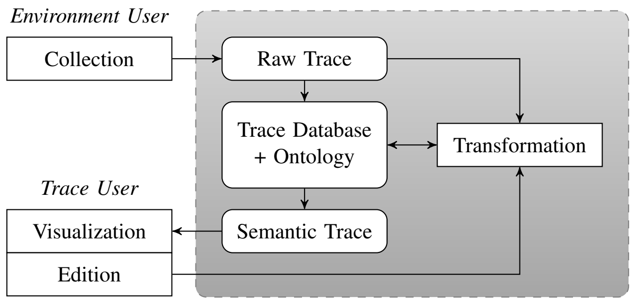

Figure 3. Schema of GAMELAN trace-based system.

4.1 Content description language

The descriptive approach is not to keep the content stored, because content is usually partial, incomplete or poorly defined (*ad hoc* formats, imperfect knowledge of it, etc.). Rather, it is better to retain a description of the content that enables to reproduce it. The description may include the main points to reproduce, the author's intention to comply [7], the graphical appearance, etc.

So, the description of the content of a work is an approach increasingly adopted in response to the technical complexity (mostly digital) of content: instead of maintaining a technical object that we may no longer know how to handle, we shall construct a description to reinvent this object with the tools we will have at hand when the time comes. Such a description necessarily introduces a deviation from the original: the challenge being that this difference does not affect the integrity nor the authenticity of the work.

The main question is how to determine such a description language. The score used in the so-called classical music, is a good example of such a language. Instead of stepping on the impossible task to keep a musical object before recording techniques, musicians preferred to keep *the instructions to create it*. Now, the complexity of the works, the mutability and fragility of digital imply that it is impossible to guarantee that a technical object will still be executable and manipulated in the future.

Several approaches are possible, but some semiotic and logic work has to be conducted to identify such a description stage:

- Semiotic, because it is necessary to characterize the objects mobilized in a production, define their significance and propose an associated representation;
- Logical, since this representation must be enrolled in a language for control actions in the proposed meta-environment.

4.2 Time axis reconstruction

The proposed description must also be temporal and allow browsing of the different states of the work. The representation of time must be done in terms of versions, traces of transformations, to offer the user the ability to revert to previous states and build new states by reusing some earlier versions of objects composing his work.

Indeed, in the final stage of production, archiving of music and sound production is generally confined to the archiving of a final version ("mastered"). Whereafter it is clearly impossible from this single object to trace the production history, nor to take back and modify the process in a different perspective, while informed musical remix is a clear identified need with *repurposing* aims.

This lead us to ensure strong timing properties through our trace-based system, not only time stamping user events from the production tools when emitting messages, but also independently time stamping a second time these events in the logging module when receiving messages. This allows us to reconstruct the time axis of the production safely.

4.3 Database browsing and timeline visualization

Here, the digital archival issue of provenance should be avoided or at least diminished upstream the ingest step, thanks to knowledge management. The GAMELAN meta-environment is intended to be able to detect crucial missing information by reasoning on the combination of software traces and user information. This important features, dedicated to the trace user, are carried out through common knowledge management tools, namely:

- Domain ontology;
- Trace database;
- Query engine;
- Semantic repository;

Besides, a timeline visualization tool brings a global view to help the answers understanding, typically showing the genealogy of the files used during the production. For example, GAMELAN can infer which files were used to compose a mixed file, hierarchically, and also deduce which is the "last mix" in a set of file; this kind of knowledge is of prime importance when a composer or a producer decides to remix a work years latter.

4.4 Creation patterns

Now that our ontology has reached a decent level and stabilizes, we enter a second phase of our ontological research: *creation patterns* design. These patterns will define audio creation acts. The use of these patterns will allow to represent a set of actions with a musical meaning, incorporating the vocabulary developed in the ontology.

Technically, we chose to stick to the OWL formalism, instead of switching to process languages like BPMN, to describe and analyse some relations between ontology objects with domain experts, especially for relations that they stressed as being of prime importance regarding test cases.

From these creation patterns, we intend to derive query patterns, in an automated way as much as possible. Indeed, a common formalism between the ontology and the creation patterns ease both reuse of the vocabulary during the pattern design phase and the translation into query patterns, especially when using compliant query languages like SPARQL as we do on our Sesame repository.

Knowledge will be then bilocalized: on the semantic repository side for objects of the trace database, and on the integrity and authenticity checker side for the formalized relations of the query patterns base.

5 CONCLUSION

Along this position paper of the GAMELAN research project, we presented how a knowledge management approach for digital audio production workflows could be of great utility at several time horizons: in the immediate time of production, in the intermediate time of collection, and in the long term preservation.

We also detailled how we are combining a trace-based architecture and an ontology-driven knowledge management system, the latter being build upon differential semantics theory. Technically, semi-automatic production process tracking feeds a semantic engine driven by production process ontology levels. Clearly, this requires both knowledge engineering and process engineering but also digital preservation methods awareness.

At last, the project will provide the following results:

- A software environment, published as free software, used to drive selected production tools and capable to accommodate to other tools later on, thanks to its openness;
- A representation and description language of manipulated content, including their temporal variation and transformation;
- A representation language of the digital audio creation process.

ACKNOWLEDGEMENTS

The GAMELAN project is funded by French National Research Agency. It started in 2009 and will end in 2013. The partners are:

- IRCAM (*Institut de Recherche et Coordination Acoustique/ Musique*),
- Heudiasyc – UMR 7253 CNRS UTC,
- INA (*Institut National de l'Audiovisuel*),
- EMI Music France.

REFERENCES

[1] R.L. Ackoff, 'From data to wisdom', *Journal of applied systems analysis*, **16**(1), 3–9, (1989).

[2] B. Bachimont, 'Ingénierie des connaissances', *Hermes Lavoisier, Paris*, (2007).

[3] B. Bachimont, A. Isaac, and R. Troncy, 'Semantic commitment for designing ontologies: a proposal', *Knowledge Engineering and Knowledge Management: Ontologies and the Semantic Web*, 211–258, (2002).

[4] H. Barki and J. Hartwick, 'Measuring user participation, user involvement, and user attitude', *Mis Quarterly*, 59–82, (1994).

[5] Elie During, 'Entretien avec Franck Madlener', in *L'Étincelle*, ed., Ircam, Paris, (2010).

[6] D. Fensel, F. Van Harmelen, M. Klein, H. Akkermans, J. Broekstra, C. Fluit, J. van der Meer, H.P. Schnurr, R. Studer, J. Hughes, et al., 'On-to-knowledge: Ontology-based tools for knowledge management', in *Proceedings of the eBusiness and eWork*, pp. 18–20, (2000).

[7] L. Gaillard, J. Nanard, B. Bachimont, and L. Chamming's, 'Intentions based authoring process from audiovisual resources', in *Proceedings of the 2007 international workshop on Semantically aware document processing and indexing*, pp. 21–30. ACM, (2007).

[8] D. Giaretta, 'The caspar approach to digital preservation', *International Journal of Digital Curation*, **2**(1), (2008).

[9] T.R. Gruber et al., 'Toward principles for the design of ontologies used for knowledge sharing', *International journal of human computer studies*, **43**(5), 907–928, (1995).

[10] I. McLeod, H. Evans, P. Gray, and R. Mancy, 'Instrumenting bytecode for the production of usage data', *Computer-aided design of user interfaces IV*, 185–195, (2005).

[11] A. Newell, 'The knowledge level', *Artificial intelligence*, **18**(1), (1982).

[12] F. Rousseaux and A. Bonardi, 'Parcourir et constituer nos collections numériques', in *CIDE Proceedings*, pp. 133–142, Nancy (France), (2007).

[13] S. Smith, E.D. Schank, and B.M. Tyler. Instrumented application for transaction tracing, March 29 2005. US Patent App. 11/092,428.

[14] A. Vincent, B. Bachimont, and A. Bonardi, 'Modéliser les processus de création de la musique avec dispositif numérique : représenter pour rejouer et préserver les œuvres contemporaines', in *Journées francophones d'ingénierie des connaissances (accepted)*, (2012).

[15] M. Wright, A. Freed, and A. Momeni, 'Opensound control: state of the art 2003', in *Proceedings of the 2003 conference on New interfaces for musical expression*, pp. 153–160. National University of Singapore, (2003).