Understanding Music Production and Content Management as an Emerging Digital Ecosystem

Karim Barkati IRCAM
1, place Stravinsky
75014 Paris, France
karim.barkati@ircam.fr

Francis Rousseaux CReSTIC
Reims University, EA 3804
Moulin de la Housse - BP 1039
51687 REIMS Cedex 2, France
francis.rousseaux@univ-reims.fr

ABSTRACT

Digital studios trace a great amount of processes and objects. The important flow of these traces calls for a system to support their interpretation and understanding. We have studied and developed such a system in the digital music production context – within the Gamelan research project – towards musical object and process reconstitution. We present the results we obtained from combining trace engineering, knowledge modeling and knowledge engineering, based on the differential elaboration of a strongly-committed ontology, standard formats and common knowledge management tools. We conclude by discussing some hypothesis about trace-based knowledge management, digital music preservation and reconstitution, opening on to some considerations about artistic style.

Categories and Subject Descriptors

D.2.13 [**Software Engineering**]: Reusable Software—*Domain engineering*; H.1.1 [**Models and Principles**]: Systems and Information Theory—*Value of information*; H.2.1 [**Models and Principles**]: Logical Design—*Data models*; H.2.3 [**Models and Principles**]: Languages—*Data description languages, Query languages*; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Retrieval models, Selection process*; H.4.1 [**Information Systems Applications**]: Office Automation—*Workflow management*; I.2.4 [**Artificial Intelligence**]: Knowledge Representation Formalisms and Methods—*Representation languages, Semantic networks*; J.5 [**Arts and Humanities**]: Architecture, Music; K.4.3 [**Computers and Society**]: Organizational Impacts—*Reengineering*

General Terms

Langages, Management, Reliability

Keywords

Digital Music Production, Reconstitution and Preservation, Trace Engineering, Knowledge Management, Digital Humanities, Style and Creativity, Digital studio.

MEDES'13 October 29-31, 2013, Neumünster Abbey, Luxembourg

1. INTRODUCTION

In the music production studio, everything is about creativity. Until now, music tools design has mainly focused on the making of the final product, because the very first aim of the studio is to provide the creator with efficient means to make and shape the musical object he or she came in the studio for. But this requisite priority on creativity has overshadowed another need that appears later: reconstitution.

1.1 Digital Music Production Stakes

Of course, creativity empowering raises in itself tough challenges to work out. For instance, on the conceptual side, bridging the gap between creative thinking and application interfaces remains a challenging issue [26, 5], while on the technical side, the heterogeneity of tools, systems, components, protocols and interfaces keeps renewing difficulties for the management of production environment [19, 11].

So today, a creator finishing his or her work in a studio marks the end of the production process: the so-awaited object is here at last, thus the creator, producer, sound engineer and all the people involved are happy or at least relieved; the goal is reached and the story reaches its end. However, at this very moment, because the final object is there, no one wonders about its reconstitution.

But – say in ten years' time – when "back-catalog" teams of music compagnies want to edit some easy to sell *Greatest Hits* at up-to-date audio formats, mining the musical archives is no longer easy. Back to the reachable recorded digital files, it may be painful to figure out which one of the bunch of files left is the one we are looking for. File dates are unreliable, because they may have change during copies. File names are clearly not more reliable than file dates.

Closer in time – say five years after the production – music companies face problems when dealing with another typical activity: repurposing (karaoke, spatialization, movie remix, game remix, etc.). Indeed, for such time laps, we cannot rely on the project file that professional Digital Audio Workstations (DAWs) provide, because any element of any layer of the digital production system is likely to prevent the project file from replaying. Obsolescence often leads to forbid replayability, whatever the elements: DAW, plugin, driver, operating system, machines, etc. Moreover, most professional DAWs project file formats are closed.

Even sooner – say two months after the production – the simple task of collecting vital information on the contributors who actually worked on the project may turn into a real problem. A musician may have been replaced by another without logging his or her name. Or a name is missing because only the musician's nickname is given and there are no phone numbers either. There is a whole set of information on contributions (name, role, time spent, etc.) necessary to manage salaries, rights and royalties that regularly proves hard to collect afterwards. Obviously, this kind of information would be far easier to collect directly at the time of production.

1.2 Digital Music Production Knowledge Management Issues

Music production lacks means of reconstitution [15] both for its final object and its production process, to master authenticity, integrity and reusability current walls. What are the conditions for reconstitution? Can these conditions remain noninvasive regarding the creative priority of the studio? Which level of knowledge is needed both in quantity and in quality?

First of all, reconstitution requires collecting traces during the production process itself. Automatically-collected software traces differ from human-entered traces. The former can be seamlessly collected through automatic watching components, with interface traces and logs as heuristic material, while the latter inevitably requests a human contributor for information that cannot be automatically captured or inferred from automatic traces. A full production tracking environment would resemble *Living Labs* [4, 24], towards a "Living Studio" [23, 8].

Secondly, these traces call for an appropriate knowledge model. To stay as little invasive as possible, such a model should provide means to determine which information is worth asking to humans or not during the production compared to the creativity disturbing cost. More generally, without a knowledge model it would be impossible to give traces a meaning, nor to predetermine what kind of traces is worth capturing. To achieve this model, professional knowledge has to be identified, listed and characterized with domain experts, defining a digital music production "Knowledge Level" [21].

Lastly, on the technical side of the Symbol Level, the knowledge this model holds has to be shaped into a form exploitable by programs, following a proper formalism. Also, the overall environment, from tracking to knowledge base querying, is best designed upon standard technologies and formats, regarding the reconstitution aim. Both modeling (at Knowledge Level) and formalizing (at Symbol Level) entail their own pitfalls when dealt with separately, but adjusting their combination could hopefully ward off the curse.

2. MATERIAL AND METHODS

To address the reconstitution issue of digital music production, we developed a software meta-environment upon the production environment, by combining trace engineering, knowledge modeling and knowledge engineering.

2.1 Functional Overview

As a meta-environment, *Gamelan*[1] traces data during the production process and utilizes formalized knowledge to exploit collected data, both during and after production time.

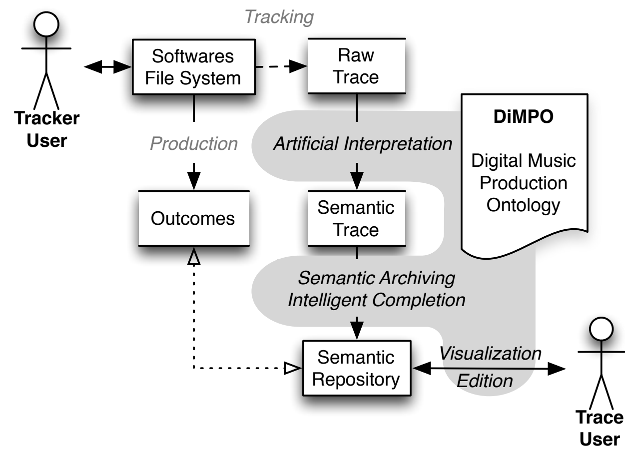

Figure 1: Architecture overview of the *Gamelan* meta-environment

Gamelan promotes two categories of users: users of the tracking system, who generate traces while they interact with the digital tools of the studio during the production process, and users of these traces, at the other end of the meta-environment. If a user of the tracker is also user of the trace, he or she simply get a feedback loop on the creative process, for example process evaluation.

The upper left region represents the trace engineering part of *Gamelan*, starting from the digital music production process of the tracker user. Tracking this process should respect the noninvasive constraint against creativity as much as possible (tracking dotted arrow on Fig. 1), and feeds the system with raw traces, which are precious but too difficult to exploit under this primitive form.

From this raw trace point, the DiMPO ontology becomes compulsory for any further operation, which is denoted on Fig. 1 by the gray semantized zone that surrounds every function coming next. This ontology, as a digital music production knowledge model, is the back-bone of this second part of the *Gamelan* meta-environment, directed to the trace user.

2.2 Use Cases and Technical Functionalities

The *Gamelan* project embraces various creative practices related to its partners core business and expertise, who defined three main reconstitution test cases.

IRCAM *Recovery assistance and synthesis of information from one phase to another of a record.* Follow 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.* Log every DAW operation of a composer during the composition of a jingle. — Ensure that the file called "Final-Mixdown"

[1] *Gamelan* is the name of both a French ANR research project and of the meta-environment developed during this project.

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.* Test the replacement of the drum from a recording made under *Gamelan*. — Accurately identify which tracks to replay; substitute an identified track to another; replay the final mix session with the replaced tracks.

To reach these use cases, the main technical functionalities of *Gamelan* meta-environment include at different levels: tracing, acquisition, ingestion, reasoning, requesting, browsing, file genealogy visualization, integrity and authority checking, and archiving. Moreover, *Gamelan* relies on standard formats, such as: OWL, OpenRDF, RDFS, SparQL and OSC.

2.3 Trace Engineering

The first step deals with raw production traces, through logging user interaction events and collecting additional contextual information, manually when necessary for this latter.

2.3.1 Operational Tracking

Traces are to be mobilized in never totally predictable contexts 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.

In order to produce usage data [20, 27], we hacked open-source domain production softwares, like Audacity[2] (written in C++), to send a complete OSC message [30] each time the user performs an action through a user-level function call of the software, built with: application name, application version number, time stamp, function name, function parameters.

We developed a tracing and logging application that logs every message received during the production from three sources:

- "gamelanized" applications for actions logs (`OSCMessages.txt`)
- File System for file movements and creation (`FolderState.txt`)
- Operating System for application swaps (`CurrentApplication.txt`)

This tracker adds a reception time stamp and keeps track of every version of modified files in a backup folder, for file genealogy analysis and preservation purposes. Fig. 2 shows excerpts of log files, reduced to fit here (some timestamps and/or other information are truncated).

2.3.2 Manual Informing

Several retained use cases, like contributors listing, require further information that cannot 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 rapidly appears: on the one hand, the more contextual information feeds the system, the more informative the knowledge management might be, but on the other hand, the more a system asks a user to enter data, the more the user may reject the system [3].

[2]`http://audacity.sourceforge.net`, an open-source software for recording and editing sounds.

```
———————————————— OSCMessages.txt ————————————————
2012-07-09 10:09:36546 +02 audacity 1.3 FileNew
2012-07-09 10:09:36553 +02 audacity 1.3 FileSaveAs  test.aup
2012-07-09 10:09:36560 +02 audacity 1.3 ImportAudio test.aup noise.wav
2012-07-09 10:09:36561 +02 audacity 1.3 ImportAudio test.aup clicks.wav
2012-07-09 10:09:36563 +02 audacity 1.3 Select "noise", "clicks"; Begin="1.931";
2012-07-09 10:09:36571 +02 audacity 1.3 ExportAudio test.aup mix.aif
2012-07-09 10:09:36581 +02 audacity 1.3 FileClosed test.aup

———————————————— CurrentApplication.txt ————————————————
2012-07-09 10:09:36544 +02 ApplicationActivated net.sourceforge.audacity
2012-07-09 10:09:36582 +02 ApplicationActivated com.apple.dt.Xcode
2012-07-09 10:09:36593 +02 ApplicationActivated com.apple.finder

———————————————— FolderState.txt ————————————————
folder-state 0       2012-07-10 16:22:58961547 +02
2012-01-20 18:07:65253 +01        noise.wav
2012-01-20 18:07:65253 +01        clicks.wav
...
folder-state 21       2012-07-10 16:23:59005107 +02
2012-01-20 18:07:65253 +01        noise.wav
2012-01-20 18:07:65253 +01        clicks.wav
2012-07-10 16:23:59005 +02        mix.aif
2012-07-10 16:23:58980 +02        test.aup

———————————————— Seances.txt ————————————————
Demo        file://localhost/Users/Barkati/Music/Demo/
Recording 2012-01-20 15:59:28783242 +02 2012-01-20 19:59:39583242 +02
        Alain Bonardi        Artistic director
        Emmanuelle Swiercz        Pianist
Mix        2012-07-10 15:04:39889830 +02
        Karim Barkati        Sound engineer
```

Figure 2: Log files excerpts

In our case, the balance between quantity and quality of information has to be adjusted in a close relationship with the strongly-committed ontology we have been incrementally developing with domain experts [28] and which is 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 as 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, as information integrity checking.

2.4 Knowledge Modeling

Digital music production knowledge modeling requires first constituting an analysis corpus, essentially because of the oral tradition of the domain. Then, the preservation aim of *Gamelan* drives to try and ensure the robustness of the model, which we addressed with a differential method.

2.4.1 Music Production Knowledge

Usually, the modeling phase begins with a corpus analysis from a collection of candidate-documents selected on their relevance [25]. But in the case of digital music production, such a corpus does not exist, *i.e.* no written document can provide sufficient support to terms selection. Indeed, vocabulary, and by extension all the production process, relies on musical practices that are acquired more by experience than by teaching.

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 "ideal and experimental" object:

this uniqueness leads to a possibly infinite number of ways to create [10]. Thus, to achieve this essential phase of study, we needed to make up our own 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 "music production 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 [21].

2.4.2 Production Process Modeling

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

Our music production process modeling 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.

2.5 Knowledge Engineering

The knowledge engineering part of *Gamelan* spreads on several operational implementation stages: interpreting raw traces into semantic traces, managing a server for these semantic traces, deploying a semantic repository with reasoning capabilities from the ontology, prototyping use case queries.

2.5.1 High-Level Technical Architecture

The technical architecture is seen as a meta-environment relying on the *production environment*, which includes various digital audio production tools at work in the process.

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 an evaluation of the current process and a context sensitive help to the user at any time. 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-invasiveness and transparency for the meta-environment.

Finally, an *archive unit* will allow a 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 [16], and adapted to the preservation of the production process.

2.5.2 Artificial Interpretation

Interpretation of raw traces according to the ontology yields semantic traces as ontological individuals. A smart implementation of a converter checks for uniqueness of these individuals against a knowledge base when necessary (Fig. 3).

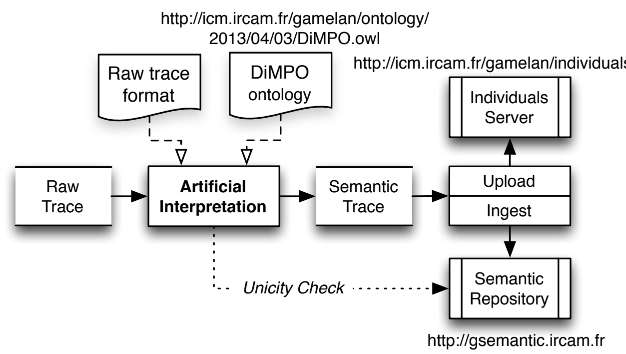

Figure 3: Trace interpretation and management

Raw Traces Interpretation.

Raw traces are not directly informative nor exploitable under this raw form of log files (see Sec. 2.3). The converter interprets theses traces according to DiMPO ontology and OWL language in order to convert them in ontological individuals, *i.e.* in semantic traces. A few interconnected DiMPO individuals are shown on Fig. 4 as "owl:NamedIndividual" elements, identified by a unique URI that ensures relations between individuals.

Uniqueness Checking.

If a DiMPO individual produced by the converter is intended to be ingested into an existing semantic repository, then the converter shall check whether this individual is already there, to ensure individuals uniqueness. Moreover, a mechanism of index attribution recovers current indexes for each DiMPO class existing in the semantic repository before individuals numbering.

Individuals and Ontology Servers.

Before being ingested into the semantic repository, semantic traces are uploaded on the dedicated individuals server `http://icm.ircam.fr/gamelan/individuals` in order to provide an internet location to loaded DiMPO individuals. New versions of the ontology are also uploaded on a server, at addresses like `http://icm.ircam.fr/gamelan/ontology/2013/04/03/DiMPO.owl`.

2.6 Semantic Repository

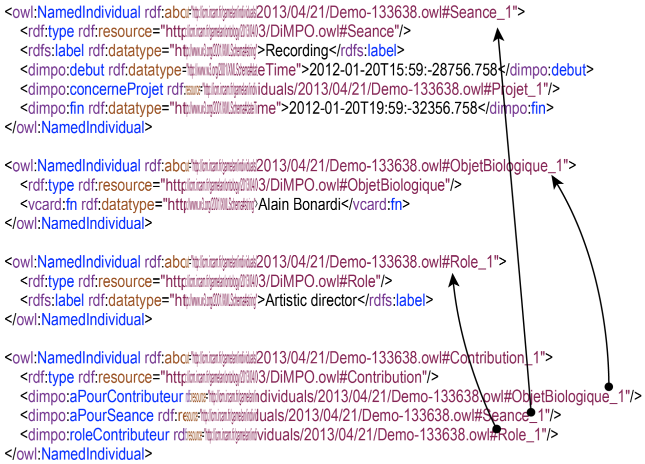

Figure 4: Some interrelated DiMPO individuals

A Sesame OpenRDF semantic repository has been installed from an Ontotext OWLIM-Lite version. It handles structured data storage and management, reasoning and querying.

Data Storage and Management OWL/RDF data ingestion on the semantic repository is triggered by a short Java program using Sesame API. An online graphical interface allows repositories management at `http://gsemantic.ircam.fr`.

Reasoning The OWLIM inference engine performs completion of facts through "total materialization" at load time. This reasoning strategy slows down upload but speeds up retrieval and querying.

Querying The semantic repository embeds a query engine accessible through HTTP. We use the SPARQL[3] language to write RDF queries against the semantic repository, with triple patterns syntax.

3. RESULTS

Results spread on several levels: an operational meta-environment with production tracking, a strongly-committed ontology for digital music production domain, a raw trace interpreter, a query manager, and a set of queries.

3.1 Production Tracking

The first result is a production tracking system, at technical level. The development affects three layers:

- software and file system operation tracing, based on application messaging;
- production file monitoring and back-up recording;
- manual entry information logging, through an ontological-conform interface.

This part integrates musical and sound production softwares and has its own non-invasive interface through a menu for manual informing.

[3]SPARQL is a recursive acronym that stands for "SPARQL Protocol and RDF Query Language".

3.2 Database Querying

The digital archival issue of provenance should be avoided or at least diminished upstream the ingest step. The *Gamelan* meta-environment allows to detect crucial missing information by reasoning on the combination of software traces and user information, from expert knowledge. This important features, dedicated to the trace user, are partially carried out through production tracking and common knowledge management tools, such as domain ontology, query engine, and semantic repository.

For instance, one can query the semantic repository in order to check whether expected contributors and their roles on the project are well informed or not. Sets of queries are designed and managed in a dedicated interface developed for *Gamelan*.

```
PREFIX scenario01:<http://icm.ircam.fr/gamelan/
  individuals/2013/04/06/scenario01-123240.owl#>
PREFIX rdf:<http://www.w3.org/1999/02/
  22-rdf-syntax-ns#>
PREFIX rdfs:<http://www.w3.org/2000/01/rdf-schema#>
PREFIX vcard:<http://www.w3.org/2006/vcard/ns#>
PREFIX dimpo:<http://icm.ircam.fr/gamelan/
  ontology/2013/04/03/DiMPO.owl#>

SELECT ?Name ?Role
WHERE {
  ?subject rdf:type dimpo:BiologicalObject .
  ?subject vcard:fn                ?Name .
  ?contrib dimpo:hasContributor ?subject .
  ?contrib dimpo:hasRole         ?roleID .
  ?roleID  rdfs:label            ?Role .
}
```

Name	Role
Emmanuelle Swiercz	Pianist
Alain Bonardi	Artistic Director
Nicolas Thelliez	Sound Engineer

Figure 5: *Contributors* Query and Results

3.3 Time Axis Reconstruction

In every selected use case, the reconstitution aim implies a time axis reconstruction capability.

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* use case aims of EMI Music France for instance.

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.

For example, a typical query can retrieve and order audio files movements (imports and exports) in a project (results on Fig. 6).

3.4 Timeline Visualization

Besides, a dedicated timeline visualization tool developed by INA brings a global view to help understand query re-

```
SELECT ?SoundfileName ?MoveID ?Timestamp
WHERE {
  ?FileID   rdf:type          dimpo:Soundfile .
  ?FileID   dimpo:fileName    ?FileName .
  {?MoveID  dimpo:hasSource   ?FileID . } UNION
  {?MoveID  dimpo:exportsFile ?FileID . }
  ?MoveID   dimpo:timestamp   ?Timestamp .
}
ORDER BY ?Date
```

FileName	MoveID	Timestamp
rec1.wav	ng:ImportAudio_1	2012-04-05T12:20:52
rec2.wav	ng:ImportAudio_2	2012-04-05T12:20:59
mix1.wav	ng:ExportAudio_1	2012-04-05T15:26:14

Figure 6: Retrieval and ordering of file movements

sults, 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 later, as pointed out on INA/GRM use case.

3.5 Creation Patterns

When DiMPO ontology reached a decent level and stabilized, we entered a second phase of our ontological research: *creation patterns* design. Creation patterns define audio creation acts, such as editing, shown on Fig. 7 (in UML). The use of these patterns allows to represent a set of actions with a musical meaning, incorporating the vocabulary developed in the ontology.

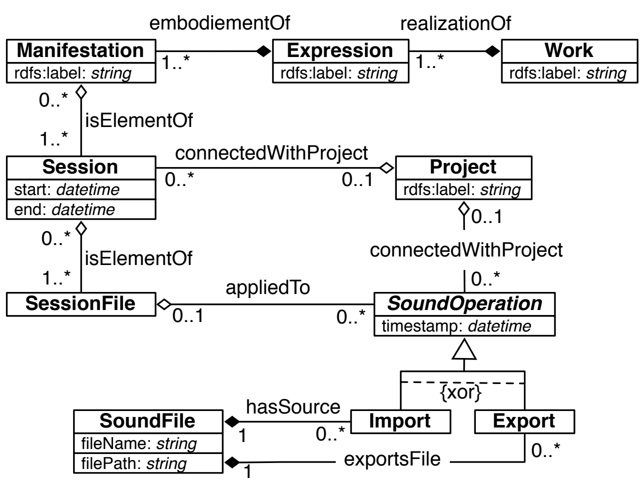

Figure 7: *Import/export* Pattern

Query patterns (cf. Sec. 3.2) are grounded on these creation patterns. Reuse of ontology vocabulary in creation patterns eases their translation into query patterns, especially when using RDF compliant query languages like SPARQL, as we do onto a Sesame repository containing OWL individuals.

Here, knowledge can be viewed as bilocalized: on the semantic repository side for objects of the trace database, and on the query manager side for the formalized relations of the query patterns base.

4. DISCUSSION

In this section, we discuss modeling, reconstitution support, and time horizons of production process tracing.

4.1 Model Pervasiveness and Design Heuristics

As suggested in the architecture overview schema on Fig. 1 with the pervasive grey zone, in our system, except for the operational tracking that has to remain agnostic, the ontology drives all functional modules at each level:

Data — The manual informing module for contextual user data, especially for the entry interface design;

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

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

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

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 [18, 12]. The key position of the ontology comes from its semantic capabilities and justified deep research toward professional-knowledge modeling in music.

Yet, despite their power and thus their pervasiveness, ontologies remain human artifacts, reflecting human vision, and never elaborated without design heuristics. We developed a strongly-committed ontology incrementally, dipping into music productions with domain experts and submitting them ontology drafts. This incremental approach continued during the next phases: during software development – with developers feedback –, and during tests and validation – with users groups feedback.

The differential approach we applied along the ontology design cycles balances the random part brought by heuristics but cannot eliminate it in any way. Ontology-driven knowledge management should be aware of this contingency dimension.

4.2 A Reconstitution Support Language

The descriptive approach is not about keeping 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 [14], 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.

The combination of these semiotic and logic approaches are key concepts to unlock the reconstitution possibility of both the work as an object and the creation as a process.

4.3 Horizons of Production Process Tracing

In the computer music field, production process tracing has never been done yet. We distinguish between *user data* and *usage data*; the former corresponds to the manual informing data and the latter to the automatic tracking data. Domain professionals drew use cases in order to help select relevant user interaction with the production meta-environment.

This production process tracing strategy aims for 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.

5. CONCLUSION

Traditional places of creation generate final objects or art works that are closed: creativity is emphasized but the creation process is most often lost, locking both object reconstitution and process reconstitution. In this context, Living Labs often attempt to trace the creation process, by recording actions for usage study. But, be they objects or process recordings, how to understand digital studio outputs?

5.1 Synthesis: Unlocking Reconstitution

We presented how a knowledge management approach for digital music 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 detailed how we combined a trace-based architecture and an ontology-driven knowledge management system, the latter being built upon differential semantics theory for sustainability. Technically, semi-automatic production process tracking feeds a semantic engine driven by production process ontology levels. Clearly, this requires both trace engineering and knowledge engineering, and also digital preservation methods awareness.

The idea of such a meta-environment like *Gamelan*, viewed as a trace-based system, meets clear needs in the community. Moreover, 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, with the diversity of its components;

- Integration of both languages in a single control environment.

Digital studios and Living Labs produce a great amount of production process traces [13] that could be better understood – and thus more easily exploitable – using semantic trace strategies such as those developed for the case of digital music production by *Gamelan*, combining semi-automatic job process tracing and content and process modeling.

5.2 Perspectives: Meeting Style

Currently, our system can support trace interpretation only up to a certain point, which is style [22, 6]. Meeting style reconstitution would need further modeling effort at higher level, which should be partially eased by our lower level creation patterns and the object and process reconstitution methods we developed. Further studies shall evaluate to what extend creation process style can be modeled.

To envision style modeling from semantic traces will require to rely on experts of art humanities at least, typically in our music production case on musicologists and composers.

Of course, approaching style reconstitution is of great interest [7, 9]. However, it may be perceived by creators as a provocative attempt to unraveling the mystery of art and creation. Then, we are entitled to wander if art objects opacity regarding their making is not a consequence of a mystery will from makers. If it is the case, new reconstitution capabilities could be perceived both as a cure and a poison.

This is probably a first class concern of future Digital Humanities culture [17, 29], but from our point of view, the advent of style pattern reconstitution would not reduce creative processes nor creativity potentials, but rather most likely shift them.

Acknowledgments 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.

6. REFERENCES

[1] R. 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] H. Barki and J. Hartwick. Measuring user participation, user involvement, and user attitude. *Mis Quarterly*, pages 59–82, 1994.

[4] B. Bergvall-Kåreborn, C. Ihlström Eriksson, A. Ståhlbröst, and J. Svensson. A milieu for innovation–defining living labs. In *2nd ISPIM Innovation Symposium, New York*, 2009.

[5] N. Bonnardel and E. Marmèche. Evocation processes by novice and expert designers: Towards stimulating analogical thinking. *Creativity and Innovation Management*, 13(3):176–186, 2004.

[6] N. Bonnardel and F. Zenasni. The impact of technology on creativity in design: An enhancement? *Creativity and Innovation Management*, 19(2):180–191, 2010.

[7] J. D. Carney. The style theory of art. *Pacific Philosophical Quarterly*, 72(4):272–289, 1991.

[8] J. S. Downie, A. F. Ehmann, and X. Hu. Music-to-knowledge (m2k): a prototyping and evaluation environment for music digital library research. In *Digital Libraries, 2005. JCDL'05. Proceedings of the 5th ACM/IEEE-CS Joint Conference on*, pages 376–376. IEEE, 2005.

[9] S. Dubnov, G. Assayag, O. Lartillot, and G. Bejerano. Using machine-learning methods for musical style modeling. *Computer*, 36(10):73–80, 2003.

[10] E. During. Entretien avec Franck Madlener. In Ircam, editor, *L'Étincelle*, Paris, 2010.

[11] E. A. Edmonds, A. Weakley, L. Candy, M. Fell, R. Knott, and S. Pauletto. The studio as laboratory: combining creative practice and digital technology research. *International Journal of Human-Computer Studies*, 63(4):452–481, 2005.

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

[13] A. Følstad. Living labs for innovation and development of information and communication technology: a literature review. *The Electronic Journal for Virtual Organizations and Networks*, 10(7):99–131, 2008.

[14] 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*, pages 21–30. ACM, 2007.

[15] A. Galey, R. Cuningham, B. Nelson, R. Siemens, P. Werstine, et al. Beyond remediation: The role of textual studies in implementing new knowledge environments. *New Knowledge Environments*, 1(1), 2009.

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

[17] F. Giraud, F. Jauréguiberry, S. Proulx, et al. *Usages et enjeux des technologies de communication*. Liens Socio, 1970.

[18] T. 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.

[19] T. Lubart. How can computers be partners in the creative process: classification and commentary on the special issue. *International Journal of Human-Computer Studies*, 63(4):365–369, 2005.

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

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

[22] F. Pachet. The future of content is in ourselves. *Computers in Entertainment (CIE)*, 6(3):31, 2008.

[23] K. R. Page, B. Fields, D. De Roure, T. Crawford, and J. S. Downie. Reuse, remix, repeat: the workflows of mir. *Proceedings of the International Society for Music Information Retrieval (ISMIR)*, 2012.

[24] M. Pallot, B. Trousse, B. Senach, D. Scapin, et al. Living lab research landscape: From user centred design and user experience towards user cocreation. In *First European Summer School'Living Labs'*, 2010.

[25] F. Rousseaux and A. Bonardi. Parcourir et constituer nos collections numériques. In *CIDE Proceedings*, pages 133–142, Nancy (France), 2007.

[26] D. A. Schön. *The reflective practitioner: How professionals think in action*, volume 5126. Basic books, 1983.

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

[28] A. Vincent, B. Bachimont, A. Bonardi, et al. Modéliser les processus de création de la musique avec dispositif numérique: représenter pour rejouer et préserver les œuvres contemporaines. *Actes des 23ès Journées Francophones d'Ingénierie des Connaissances (IC 2012)*, 2012.

[29] F.-Y. Wang. Is culture computable? *Intelligent Systems, IEEE*, 24(2):2–3, 2009.

[30] 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*, pages 153–160. National University of Singapore, 2003.