HAL
open science

Multimedia applications for playing with digitized theater performances

Marc Caillet, Cécile Roisin, Jean Carrive

To cite this version:

Marc Caillet, Cécile Roisin, Jean Carrive. Multimedia applications for playing with digitized theater performances. Multimedia Tools and Applications, 2013, 10.1007/s11042-013-1651-1 . hal-00855102

HAL Id: hal-00855102

https://inria.hal.science/hal-00855102v1

Submitted on 28 Aug 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

This is a pre-print version of the article published in :

Multimedia Tools and Applications, August 2013

DOI: 10.1007/s11042-013-1651-1

http://link.springer.com/article/10.1007%2Fs11042-013-1651-1

Multimedia Applications for Playing with Digitized Theater Performances

Marc Caillet

marc-caillet@orange.fr

Cécile Roisin

INRIA Rhône-Alpes, 655 avenue de l'Europe, F-38334 St-Ismier

+334 76 61 53 60

+334 76 61 52 07

cecile.roisin@inria.fr

http://wam.inrialpes.fr

Jean Carrive

INA, 4 avenue de l'Europe, F-94366 Bry-sur-Marne,

jcarrive@ina.fr

http://www.ina-sup.com/

Abstract: This article presents a multimedia production chain that specializes in semantically annotated digitized theater performances. Semantic annotations - we prefer the term descriptions - are expressed in a description language that combines object-oriented features, taxonomical inheritance and temporal aggregations. The descriptions that are produced from several types of content related to the same theater play are synchronized at several levels of granularity providing rich relationships between the narrative structure of the text of the play and the narrative structure of the digitized theater performances. Two applications for multimedia access and navigation are presented in this paper, namely *Dual Players*, a navigation application that allows to synchronously play acts and scenes of two recordings; and *Synthesizer* that produces a raw publication of a new audiovisual document on the basis of the recordings.

.

Keywords*: semantic annotation, temporal segmentation, theater performance, multimedia application*

1. Introduction

Recorded theater performances constitute a rich cultural heritage that presents high interest to various communities and users such as school pupils and literature teachers or stage actors learning for new characters. However, most available digital content about classical plays and their authors is provided through (hyper)textual web sites where few multimedia stuff is made available (for instance http://shakespeare.mit.edu or http://www.site-moliere.com/).

This particular domain nevertheless presents a high potential – rarely explored until now – for providing new ways of learning and experiencing these works through rich interactive applications: indeed a theater play is characterized by a unit of time, places and characters, the text is publicly available and its structure is fixed. The availability of multiple versions of recorded performances of the same play paves the way for the development of new multimedia applications as those explored in this paper.

For instance the collections of the French National Institute of Audiovisual (INA) include no less than 600 of full-length recordings of theater plays, including at least 43 plays by Molière. Several versions of the most famous plays are available and among them 7 versions of Le Misanthrope. INA is greatly interested in being able to publish this material with sophisticated means of access, for cultural or educational purposes. As it is now, INA web site (www.ina.fr) gives access to multimedia content that is organized by themes or personalities. But to go further in the access to its cultural heritage, it is necessary to investigate more attractive features in terms of navigation and content creation.

The purpose of multimedia applications considered in this article is to exploit and make the best use of the audiovisual heritage by means of a prospective exploration of virtual access to documents through semantic annotations – we will use the term descriptors. In [15], a model of multimedia productions is defined as a set of canonical processes such as premeditate, create media asset, annotate, construct message, organize, publish... The production chain presented in this paper illustrates some of these canonical processes, more precisely the annotate, package, query, organize, and publish steps. It also aims at showing how rich annotation structures can bring benefits to the automatic production of two categories of multimedia applications:

- Navigating in audiovisual documents through their descriptors. In this paper we make intensive use of temporal descriptors and references between descriptors to provide users with rich access modes inside related multimedia content, for instance two performances of the same theater play.
- Generating new audiovisual documents on the basis of existing documents and their descriptors.

This article presents two applications that have been targeted to digitized theater performances. They are called JAM! (JAM! stands for *Jouons Avec le Misanthrope!*, which means *let us play with the Misanthrope!*): *JAM! Dual Players* falls into the first category of applications while *JAM! Synthesizer* falls into the second one. These applications are use cases of our FERIA experimental framework which provides multimedia applications developers with the following tools and services: a description language called FDL (Feria Description Language) of which a detailed description can be found in [11], several descriptor-based graphic user interface tools, and content, document, analysis and description servers.

This article is organized as follows. Section 2 describes *JAM! Dual Players* and *JAM! Synthesized* with a focus on how descriptors are organized to provide innovative features in terms of hierarchical temporal description, flexibility and reusability. Section 3 presents the background of this work and related work. Finally, section 4 provides some evaluation through the first experimentations of these applications and points at some directions for future work.

2. JAM! Applications

JAM! is a set of experimental multimedia applications that rely on six different broadcast performances of *The Misanthrope*. Low-level automatic analysis is performed over the text of the play and over the performances in order to get synchronization at multiple granularity levels between the textual and audiovisual structures of the play. Two applications have currently been developed on the basis of these descriptions: namely (i) *JAM! Dual Players*, a navigation application that allows one to synchronously play acts and scenes of two recordings; and (ii) *JAM! Synthesizer* that produces a raw publication of a new audiovisual document on the basis of the

recordings.

In this section we first describe the *JAM!* corpus and the two targeted applications. Then the descriptors are presented, with a particular focus on the temporal descriptors and cross references required for temporal navigation and new audiovisual document generation. Finally we demonstrate how these structures have been used to realize the navigation, synchronization and generation services.

2.1.JAM! Corpus

The six performances were chosen because they have been archived, digitized and annotated for documentary purposes by INA. The files have been demultiplexed and decoded in order to extract the audio track. As a result, six wave files were also supplied.

The Misanthrope is composed of about 1800 rhyming alexandrine verses. We picked up the text of the play on the Gallica web site (http://gallica.bnf.fr) and then transformed it into the Text Encoding Initiative standard format.

JAM! relies on the results of automatic speech recognition tools that are run over each of the six recordings. While generally these techniques encounter multiple issues (background noise, musical interludes, mistaken/forgotten words, etc.) that contribute to degrade the accuracy of automatic transcripts, the specificity of a theater play is more favorable since the text of the play is available, which can dramatically improve the results [13], turning the recognition problem into a much simpler alignment (or synchronization) problem.

The transcription resources have been produced at GET/ENST through an alignment process that runs in two successive steps: (i) a metrical and phonetic analysis is performed over the text thanks to the Metrometer [8], a tool which implements the verse specific pronunciation rules on top of a full-fledged rule based text-to-phoneme system; (ii) on the basis of the results of the first step, the alignment process is performed over the audio tracks of each of the six recordings with the use of an enhanced version of the Sirocco speech decoding engine [14].

JAM! also benefits from additional resources:

- Manual segmentation of each of the recordings at both act and scene level.
- Documentary descriptive note of each of the recordings. These notes are those that are produced by INA librarians. They notably supply broadcasting information and the names of the television director, the stage director and the performers.
- Documentary descriptive note of the text of the play which supplies information about the characters and their role.

2.2.JAM! Applications

JAM! Dual Players is a navigation application that considers two recordings at a time and allows one to synchronously play acts and scenes of both recordings thanks to the direct access provided by the text alignment feature. Act and scene segmentations are displayed by means of a rectangular control that is located just under each of the two video players. User-friendly keyboard short-cuts make it easy to:

- linearly navigate from act to act, or from scene to scene,
- hierarchically navigate from one act to its scenes, or from one scene to the act it is part of,
- zoom in and out.

Figure 1 shows, the three scenes of the first act (note that the third scene is very short) followed by the remaining four acts. As the two performances are being played, the text is synchronously displayed at act or scene level depending on which segment type is selected in the segmentation control. Finally the name of the stage performers being playing are also synchronously displayed.

This tool aims at providing services for comparative studies among performances: multilevel navigation and text synchronization for each performance are offered but also automatic positioning with the second play.

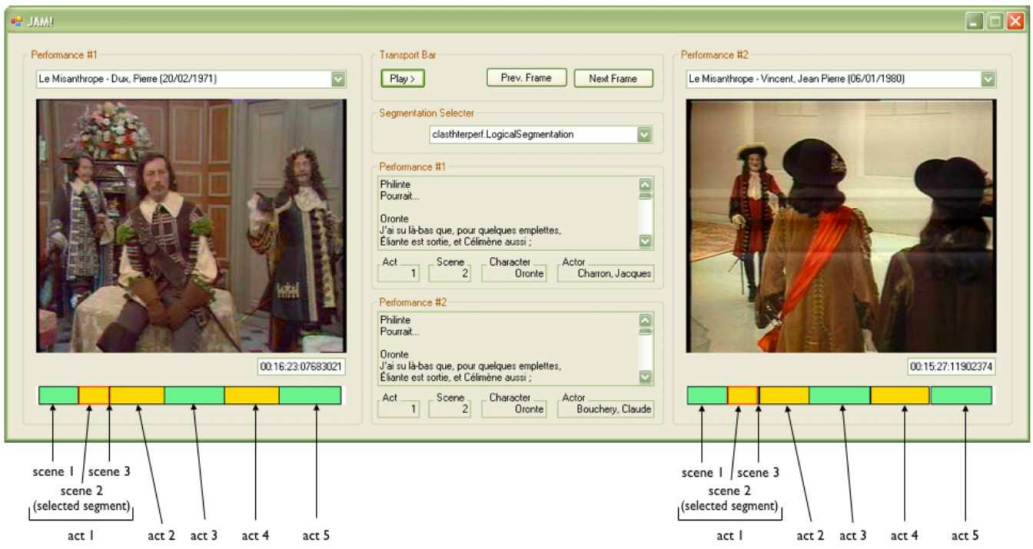

Figure 1: *JAM! Dual Players* application.

JAM! Synthesizer synthesizes a virtual play as a new audiovisual document on the basis of the six recordings available in our corpus. A new document is created by selecting, for each character, one stage performer among the six possible ones (left part of Figure 2).

The resulting virtual play is a sequence of excerpts of the existing recordings. The new audiovisual document that is shown on the right part of figure 2 has been created by picking Philinte in Dux's 1971 recorded play, Alceste in Vincent's 1980, and so forth. This document abstracts a virtual content whose dynamic creation is conducted by the text of the play and the line segmentation of each recording. The begin and end time information of each picked segment is directly extracted from the speaker segmentation of the audio tracks. This audio-driven play recombination gives a quite rough and ready virtual play (the shot video segmentation is not taken into account). However it brings amusing views of the play and open the door to richer document generations.

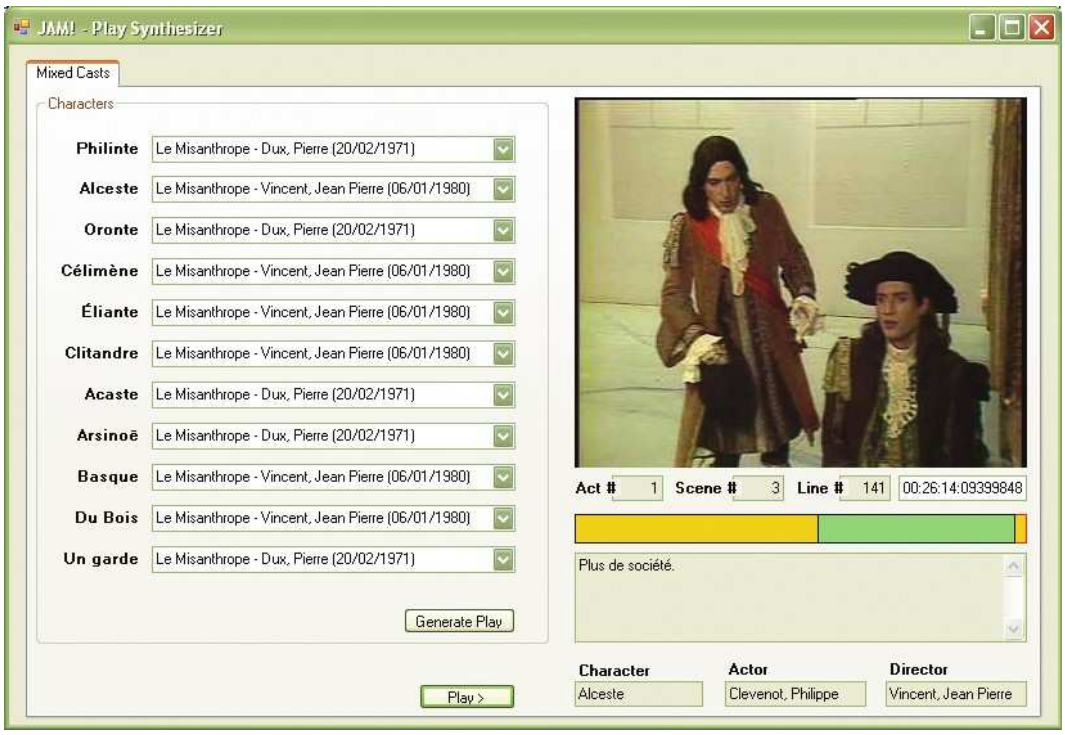

Figure 2: A virtual play generated by J*AM! Synthesizer*

Needs for descriptors for these applications

While we need to handle different types of resources, the descriptors may result from the analysis of different modalities: video, audio, text or any combination of them. Some are temporal, such as the ones that result from the temporal alignment; others are not, such as a documentary descriptive note. The composition relation between two descriptors may be complex and temporally constrained: for instance, a play may be defined as exactly five temporally successive acts; an act may, in turn, be defined as at least two temporally successive scenes. This temporal composition is defined with Allen [1] time relations combined with quantitative data (to define the number of acts for instance as can be seen in Figure 4).

Some temporal descriptors bring virtual access to documents through a specific viewpoint: for instance, a speaker segmentation of a recorded play enables navigation from one verse to another of the same character. Moreover, as we work with six different recordings of *The Misanthrope*, we may navigate from one speaker in a given recording to another speaker in another recording as is done in J*AM! Synthesizer*. This is obtained thanks to the reference operations that are defined between structured descriptors that describe the audiovisual content.

Next sections present the main descriptors (section 2.3) with the reference operations (section 2.4) that have been defined for the *JAM!* applications.

2.3.Descriptors for the JAM! Applications

Text Descriptors

The text of the play has been transformed (using XSLT) from the TEI format to a set of descriptors that represent a logical segmentation of the text at act, scene and line level. The descriptor classes of this segmentation are organized in the hierarchy of Figure 3 where the bottom level contains the text descriptors (using the *txt* namespace) and the top level are the non temporal descriptor classes of the FDL description language on top of which the applications are built (using the *fdl* namespace). For instance *fdl:D* is the higher level descriptor class from which every descriptor class inherits. It is composed of an identifier, a reference towards the upper level descriptor which is useful in case of hierarchical segmentation and a list of URNs of other descriptors that refer to the current descriptor.

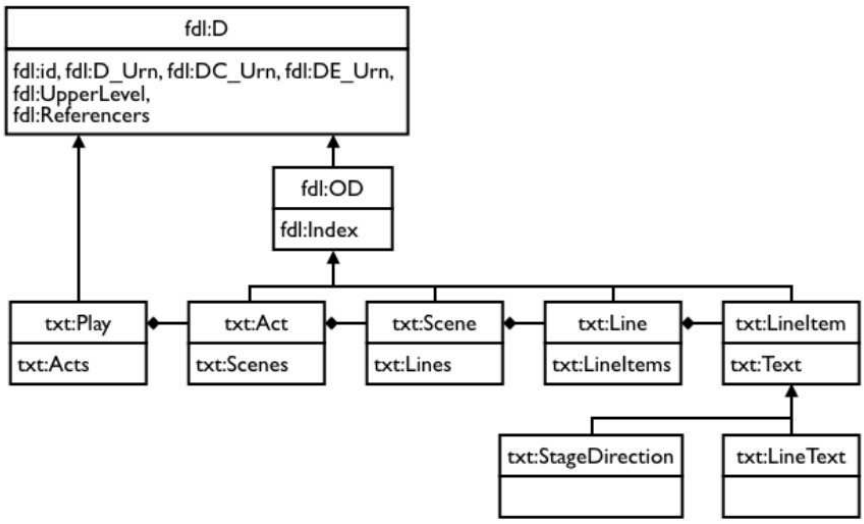

Figure 3: Logical segmentation of the text of the play

Documentary descriptors

The documentary descriptive note of the play results from a manual annotation. It states both name and role of each character. In the same way, the documentary descriptive note of each of the recorded play has been transformed into a descriptor on the basis of a text file that comes from the INA archive system. Such a note is partly composed of credits which, in particular, gather director and performers; each performer notably contains a reference towards a character that is defined in the documentary descriptive note of the text.

Audiovisual descriptors

The act and scene segmentation of each of the digitized plays, that notably contains a reference towards its related text (*ref txt:Act*, *ref txt:Scene*), results from a manual annotation directly expressed in the description language and constitutes the audiovisual descriptors hierarchy (with the namespace *av*) given in Figure 4.

The top level of this hierarchy uses basic FDL temporal descriptors *fdl:TD* and *fdl:pmTS*:

- *fdl:TD* is a simple Temporal Descriptor defining a temporal segment with an inclusive lower bound and an exclusive upper bound (close to OWL-Time *ProperInterval* definition [17]) .
- *fdl:pmTS* class defines compound descriptors in which the components are temporally related with the disjunction of the previous and meet Allen relations ([p ∨ m]). Note that this class is similar to the temporal sequence of [25] defined as a temporal aggregate with the *ibefore* relation.

More precisely, the logical structure of a recorded play — *av:PlaySeg*. — is defined by a property that holds temporal segments whose type is a temporal aggregation composed of an opening credit segment — *O* — that precedes the play itself — *P/Play* — which is followed by a closing credit segment — *C*. Both *O* and *C* are specializations of the simple *fdl:TD* descriptor class while *av:PlaySeg*., *av:Play* and *av:Act* are compound temporal segments and therefore are specialization of the *fdl:pmTS* descriptor class.

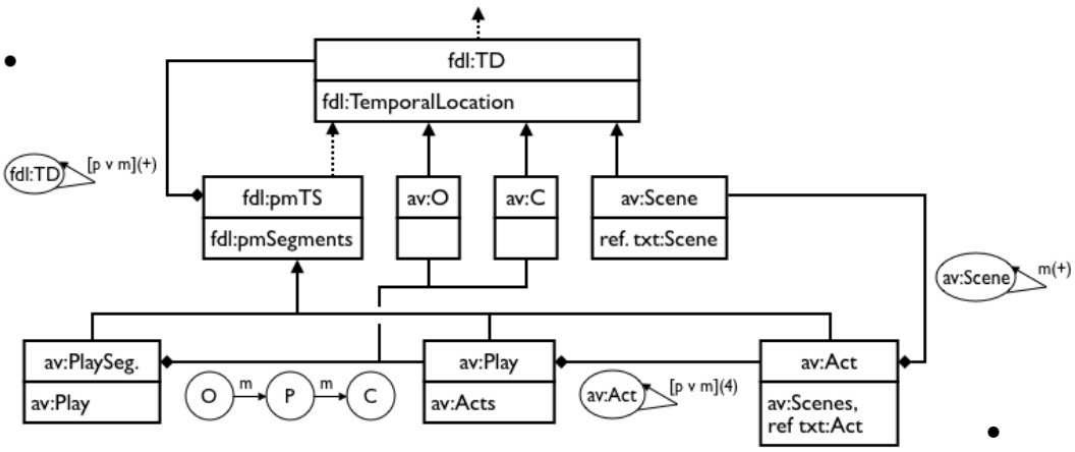

Figure 4: Logical segmentation of the recorded plays.

In this descriptor hierarchy, we can also see how Allen relations (and cardinalities) are used to express the temporal structure of the play. For instance the *[p v m](4)* expression between the *Play* and the *Act* descriptor classes denotes that a play is composed of exactly 5 acts organized in a previous or a meet relation. Similarly, the *m(+)* expression between the *Act* and the *Scene* descriptor classes denotes that an act is composed of at least 2 scenes set in a meet relation.

Speaker descriptors

Thanks to the previous audiovisual descriptors and their relationships through attributes *ref text:Act* and *ref text:Scene*, the temporal alignment process aligns the text with the recorded plays at various granularity level, from line level up to the phoneme level. We can then deduce a temporal segmentation at speaker level from the line level (under the assumption that no speak-over occurs) as represented in Figure 5.

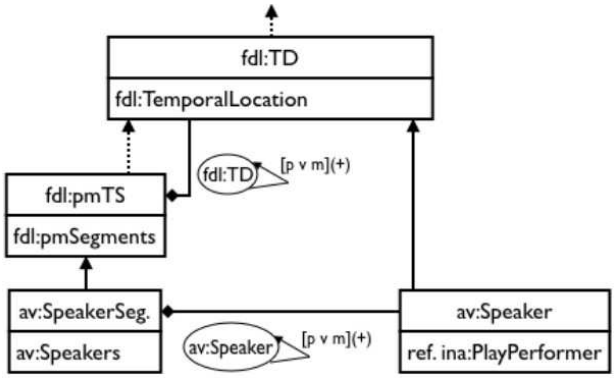

Figure 5: Speaker segmentation of the recorded plays.

Synthesis

This set of descriptors has been defined in order to cope with the needs of the two *JAM!* Applications as shown in sections 2.5 and 2.6. Therefore they do not cover broader semantic descriptions such as OntoMedia [19].

Moreover, nothing is done at shot level although automatic tools could have provided this decomposition level. Similarly no character recognition tools have been used. The consequence is that temporal synchronization between text and play can only be obtained from the audio track (see section 2.5). This however has proven sufficient to get multimedia applications for theater plays that show the relevance of the approach.

Most of these descriptors (logical segmentation of the text, documentary descriptive notes, speaker, act and scene segmentations of the recorded plays) are used in both *JAM! Dual Players* and *JAM! Synthesizer* descriptions. We can notice that this ability for reusing description classes and instances from one application to another is due to the modularity of the FDL object hierarchies.

2.4.Reference Operations Between Descriptors

Both *JAM! Dual Players* and *JAM! Synthesizer* applications are basically built with the two main functional steps: (i) selecting audiovisual documents and a set of their descriptors; and (ii) putting together graphic user interface components that display the descriptors and the associated content.

Regular multimedia applications (Flash-, Director- or SMIL-based) need references between their components being statically specified at authoring time. Conversely, *JAM!* applications dynamically link their components at runtime depending on user interactions. References are thus not defined at authoring time. This feature makes the development of these applications quite flexible because one can add or delete any component without having to worry about defining new links or deleting dead links.

The two *JAM!* applications make use of three reference types between descriptors:

- *hierarchical cross-references* between temporal descriptors that are involved in a temporal segmentation; these references are computed when the descriptors are being loaded thanks to attributes *UpperLevel* and *Referencers* of the *fdl:D* descriptor class from which all descriptors inherit. The main application of these references is to ease navigation within a hierarchical segmentation, for instance between an act and its inner scenes.
- *temporal cross-references* between temporal descriptors; these references are resolved by an enhanced event manager according to the current instant to be displayed. This instant is computed from the component currently selected and it allows to determine which descriptors must be displayed at each moment. Temporal cross-references can be used for instance to retrieve the stage performers who are playing in a given scene. Figure 6 shows a temporal cross-reference between the (partial) act and scene temporal segmentation and the (partial) temporal segmentation at speaker level. It is heavily used in *JAM! Synthesizer* to provide the virtual play resulting from the selection of performers from different performances.

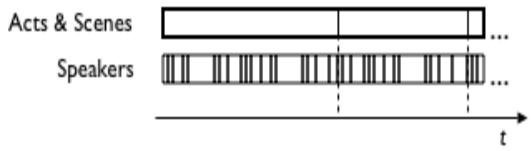

Figure 6: Temporal cross-reference between two temporal descriptors.

- *semantic cross-references*; these references are resolved by an enhanced event manager. They are computed between descriptors of different audiovisual contents that share the same structure but with different act and scene and durations (figure 7). *JAM! Dual Players* is a great example of the use of these references.

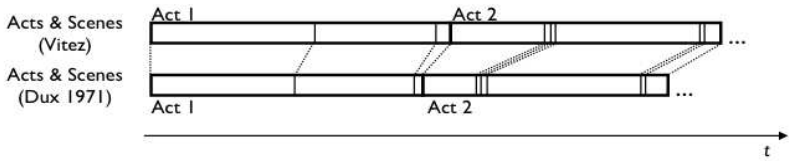

Figure 7: Synchronizing two different recordings at scene level.

2.5.Temporal and Semantic Navigation

In this section, we show how *JAM!* applications provide valuable semantic navigation features in video content thanks to their straightforward temporal visualization tools that take advantage of the temporal and semantic structures provided by the description classes.

Both *JAM! Dual Players* and *JAM! Synthesizer* synchronize the narrative structure of the text and the narrative structure of the recorded plays in the same manner. As soon as an act or a scene is selected, whether by a mouse click or through the course of a player, the text of the play is temporally aligned with the recordings at act and scene level. This feature is directly provided by means of the references that are held by the act and scene temporal segmentation towards their related text.

The applications handle multiple temporal cross-references that make it possible to know, at any moment, which actor is speaking and which are the current act, scene and line (see Figure 6). A reference from the actor in the documentary descriptive note of a recorded play towards the related character in the documentary descriptive note of the text makes it possible to display the name of the character that is being played by the speaking actor.

In *JAM! Dual Players*, act and scene segmentations of both recordings semantically cross-refer to one another (as shown in Figure 7). This results in the ability to synchronously move from one segment to another: each action that is being made on one of the segmentation controls is immediately reflected on the other (see the rectangular segmentation selector in the bottom part of Figure 1). This interface component has been defined in our framework to display temporal segmentations of type *pmTS* . Since act and scene segmentations both specialize *pmTS*, it has not been necessary to develop a new component for the segmentation selector of *JAM! Dual Players*, the *pmTS* interface component has directly been used.

2.6.Dynamic Creation of New Audiovisual Documents

As illustrated in Figure 2, *JAM! Synthesizer* creates a new audiovisual document by selecting, for each character, one stage performer among the six possible ones. This new document abstracts a virtual content whose dynamic creation is driven by the text of the play and makes use of the respective line segmentation of each recording that serves as a basis of the new document.

Basically, this application uses the same description tools as does the *JAM! Dual Players*, namely: act and scene segmentations for both the text and the audiovisual content, speaker segmentation of the audio content and descriptive notes. Besides, the *JAM! Synthesizer* description adds line segmentations of the recorded plays that are necessary to generate the excerpts of the new

synthesized document. This segmentation results from the audio segmentation tools.

A virtual content is composed of a list of excerpts of contents. An excerpt is defined by the URN of the full-length content together with a temporal location. For each line of the text, J*AM! synthesizer* first gets the current character by getting the descriptor that is referred to by the current text line (this is a reference of a non temporal descriptor towards another non temporal descriptor). It then gets the line temporal segment, from the corresponding line segmentation, that refers to the current line of the text. *JAM! Synthesizer* finally adds a new excerpt to the virtual content defined by:

- the URN of the content that is referred to by the document which is described by the current line segmentation (Dux's 1971 when Philinte is the current character, Vincent's 1980 when the current character is Alceste, in our example);
- a temporal location that determines the lower and upper bounds of the excerpt and which equals the temporal location of the current line segment. Figure 8 focuses on the first two lines of the play (Philinte addresses Alceste who then answers).

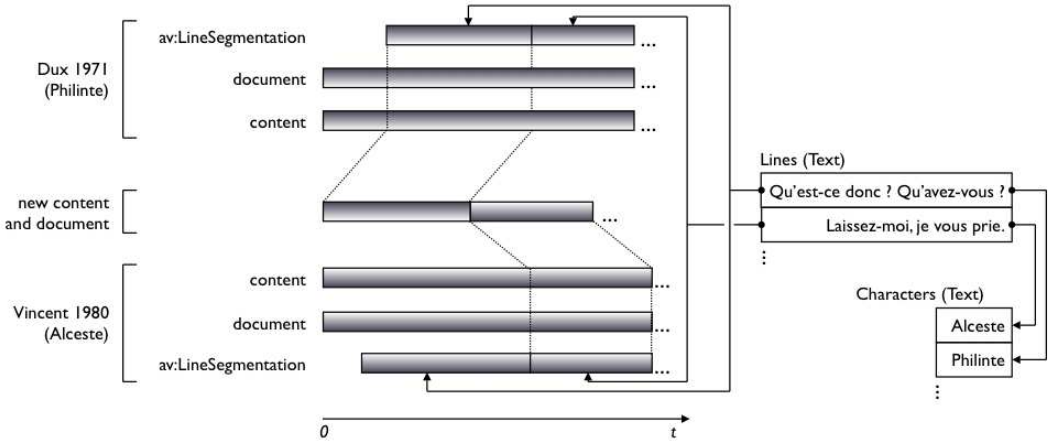

Figure 8: Dynamic creation of a new audiovisual document by J*AM! synthesizer*

While creating the new audiovisual document, *JAM! Synthesizer* also creates a multi-structured description which is composed of: a documentary descriptive note, a speaker segmentation, and act and scene segmentations. This requires the correct classification of the descriptors classes and has been obtained with the help of the FDL framework.

2.7.Usability

No formal evaluation of *JAM!* applications has been conducted but all their functionalities have been extensively tested by several persons at INA. In any circumstances, time responses are always more than satisfactory (no perceptible waiting time) and their behaviour is always fluid.

As only one theatre play (namely *The Misanthrope*) has been processed, the question remains whether the applications would support a more important amount of data, especially as time responses are concerned. Indeed, for some operations, temporal information is computed. For instance, in order to map the segmentation into phonemes to the segmentation into words, a temporal comparison is performed, as boundaries of words are not contained in the segmentation into phonemes.

At the moment, these segmentations are stored in simple lists, and a sequential and exhaustive search is performed, leading to a $O(N^2)$ complexity of the algorithm. If response time proved to be unsatisfactory as size of data grows up, simple solutions exist to reduce complexity to O(N.Log(N)). As involved segmentations contain no overlapping segments, lists sorted on beginning time points would suit, associated with a classical binary search procedure. For the general case of any arrangement of time segments, it is always possible to implement temporal interval binary trees [9], also associated with a binary search procedure. Nevertheless, note that this latter solution is more costly in terms of storage.

3. Related Work

3.1. Multimedia Applications

JAM! applications are very similar to the hypervideos from the Advene system [4]. In a recent improvement [2], this model makes use of OWL to define composition relations and to include ontology-based annotations of audiovisual contents. The main difference with our approach is related to the expressiveness for the definition of composition relations and to the way to provide references between descriptors. Moreover, one may reuse audiovisual documents and related descriptions from one application to another; the second category of applications we delineated above also re-purposes both audiovisual documents and descriptions. From this perspective, our work also bears similarities to [23] that defines an algebra to handle, transform, and reuse operations on descriptions.

[16] addresses user interface issues in multimedia applications and proposes Savanta, an advanced general-purpose visualization and navigation tool for video databases based on modeling sets of temporal intervals coupled with the OntoLog model. Temporal visualization issues are also addressed in *JAM!* applications. However we do not only address direct video content access but also the production of new content. So our description language uses a more fine-grained model that can build relations between temporal and non temporal descriptors to cope with specific properties of the content, namely recorded performances of theater plays. In Savanta, navigation and search are driven by the terms of the annotation system and it is only needed to handle a term hierarchy and term-to-interval relationships. On the contrary in *JAM!*, our purpose is to give access to audiovisual content through temporal and/or semantic relationships between several descriptions, so we need to handle multi-level temporal and non temporal structures and therefore more complex relationships between descriptions.

One way to propose enhanced video-based applications is to exploit additional resources [22] when they exist, as it is the case in *JAM!* that highly benefits from *The Misanthrope* text. Experiences with films and their scripts have also been performed in [26], where the focus is on the authoring of the alignment which does not require reasoning capabilities as provided with FDL.

3.2. Description of Audiovisual Heritage Resources

An audiovisual resource is intrinsically temporal. Furthermore it is composed of hierarchically organized temporal units. For instance, a TV broadcast is defined as a sequence of time slots containing programs; a theater play is a composed of acts that follow one another, an act contains a sequence of scenes; and a similar decomposition exists for movies. Some of these units may overlap others, some other may be separated by gaps.

Therefore, modeling hierarchical temporal structures that cope with the intrinsic structure of audiovisual content like theater plays has been the focus of the definition of our description language FDL, the description language used in *JAM!*. It is a complementary work to OntoMedia [19] which is an ontology framework that extends Hunter's ABC ontology and CIDOC CRM for cultural heritage data modeling. OntoMedia is a model for entities and events for describing fiction, where an event is an interaction between entities which is temporally located by occurrences that carry time information. This general model fits well with queries over characters for instance but not for exploring the synchronization of multiple levels structures as we have done in *JAM!*.

FDL is an object-oriented knowledge representation language (following our previous works [6] and [28]). It originates from MPEG-7 lack of formal semantics [10]. This issue has been addressed by many authors that combine OWL and MPEG-7 to overcome the latter's lack of formal semantics. OWL [24] is a widely used ontological language in the context of the semantic web for intelligent access to information. It has been designed on the basis of both description logics and the RDF language, so inference calculus can be done thanks to numerous existing reasoning tools. [27] makes use of ontologies and rules to formalize a subset of the semantic constraints of MPEG-7 DAVP profile [7]. [18], [12] and [29] have proposed ontologies that organize MPEG-7 description tools.

As stated in [2], these approaches do not cope with the needs for extensibility and semantic interoperability with existing web ontologies. That is why, instead of providing a one to one

translation of MPEG-7 descriptions at a syntactical level, they have chosen to express MPEG-7 semantics with some basic patterns of the foundational ontology DOLCE. Similarly, extensibility is provided in FDL thanks to a basic set of descriptor classes and the object-oriented inheritance paradigm with a more advanced handling of structured temporal descriptions. This extensibility has been used in *JAM!* applications for the definition of descriptor classes targeted to theater performances. As shown in the previous section, these descriptors are inserted in the class hierarchy and therefore inherit from FDL basic descriptive and computing features, such as provided by the event manager for the dynamic production of references, or by the interface controls for hierarchical navigation.

To provide knowledge representation features, [21] proposes an extension of one of the LISP dialects for OWL reasoning within an object oriented environment; it is still an ongoing work. Our approach quite differs because we had the opportunity to take advantage of the INA audiovisual application framework. This framework is built upon C# technology and covers the whole chain of services required for the cultural heritage domain: managing huge multimedia archives, handling multiple formats, offering efficient access to different kinds of users such as documentation librarians, specialists or general users. Then, instead of trying to bridge OWL and C#, which would be a long term work, and because we especially focus on the use of temporal descriptors and aggregations, we defined the FDL description language as an extension of the C# metamodel. This approach combined with tools for engineering FDL descriptor classes and instances resulted in FERIA, a complete multimedia production chain that fits our needs.

Our object-oriented approach for the use of descriptions somehow contributes to the field dealing with semantic web formalisms within an object-oriented environment ([20] and [21]). The interesting result obtained with FDL is its efficiency in computing subsumption relations thanks to its limited expressive power, that is nevertheless sufficient in our context. Basically the rule that defines the subsumption between two temporal aggregation types *t1* and *t2* states that all instances of *t1* are also instances of *t2* [3]. Similar rules are applied to handle cardinality. *JAM!* applications have indeed taken advantages from this classification service to share descriptors and their associated tools (for instance temporal segments visualization tools).

4. Discussion and Future Work

This paper has described the experimental *JAM!* multimedia applications that explore multimedia functionalities engineered on the basis of multiple levels of synchronization between the narrative structure of the text of the play and the narrative structure of several recorded theater performances. It shows the relevance of our approach that considers: (i) temporal aggregation typing that constrains descriptor classes with Allen relations and cardinalities; (ii) multi-structured descriptions of audiovisual documents with linking operations: hierarchical, temporal and semantic cross-references.

While the applications presented here contribute to the area of digitized theater plays which has yet rarely been exploited, the underlying FERIA system is a contribution to the production of multimedia applications 'in the large' thanks to reusability provided at content (audiovisual archives), description (annotations) and code levels. Thanks to the classification system, new descriptor classes can be easily inserted into the hierarchy in order to inherit from existing tools. As a consequence, the second application has been very quickly realized. This framework does not only help in the development of multimedia access and navigation applications but also enables the production of new multimedia documents in which the initial content is reorganized and combined with other content. developers have

The two applications described in this paper have been tested by two categories of users : INA librarians and students learning multimedia archiving. Surveys have been organized to get their feedback in order to obtain a qualitative evaluation of the tools. This initial evaluation has given the following positive results:

- Efficiency of the FERIA framework: access time for the content is good, even when the *JAM! Dual Players* has to update the synchronization between the two performances (cf. section 2.7).
- Usefulness of the multiple level structures of navigation in the performances: the utility of multilevel access has been proved for a long time [6], specially for complex audiovisual content. Recordings of theater plays are a good use case for providing such navigation features because the structures can be mined from several media such as text, audio and video.

- New perception of the play thanks to the integration of a great deal of related information in a single multimedia document: data from descriptive notes, source text and audiovisual content are synchronously presented to the users together with several active controls. Most users have noticed that *JAM!* applications provide a richer experience than just browsing an existing content. This provides new ways for conducting deeper analyses of the play.
- Time saving thanks to *JAM! Dual Players* that provides a straightforward comparison between two performances while it is usually considered as a tedious activity. An useful application of this tool has been pointed by our students: the tool helps them to learn how to pronounce and play a particular (sequence of) verse(s) because they can directly access the different interpretations of that verse. Accessing all performances of a same play is also a way to work on an historical perspective of the actor's play and actor's pronunciation (our corpus contains recordings that spread over a period of 20 years, from 1959 to 1980).
- Exciting and creating experiences thanks to the amusing possibilities given by *JAM!* in producing new theater performances by combining existing content.

However, several drawbacks have been noticed while testing the applications. The main one is related to the small amount of information extracted from the video content: only act and scene segmentations have been (manually) done. Fine synchronization is therefore not possible, for instance to better generate new performances with *JAM! Synthesizer*. The set of descriptors should be completed with a shot segmentation. While the interface of *JAM! Dual Players* has been positively evaluated, the interface of *JAM! Synthesizer* has been judged as being too much basic: for instance, the performers to be selected should be proposed through a set of thumbnails.

As it has been stated in the related work, we have made the choice to use our own description language and implement it in an object-oriented environment. The main advantage of this approach was its efficiency and extensibility while providing an expressive power comparable to standard languages. The set of technological tools we have developed (from low level signal analysis to metadata description and resource management) are at the level of the state of the art and our main contribution is to allow the emergence of innovative access to recorded theater collections that, to our knowledge, have not yet received this level of support.

The next step of our work consists in improving *JAM! Dual Players* by allowing synchronized navigation from one segment to another at a finer granularity. *JAM! Synthesizer* will be improved with an export feature towards a multimedia authoring language to allow manual authoring adjustments such as inserting opening and closing credits, various types of cuts between acts and between scenes, additional music, talk over, and so on. Descriptor instances must be introduced in the FDL classification structure in order to answer complex queries, such as a query for segments that contain Célimène's lines that are immediately followed by Alceste's.

Other functionalities may be added to better fit the needs of specific audiences such as actors students: bringing out verses that differ from one performance to another in the way they are pronounced, bringing omitted words and omitted verses to light, highlighting poor and rich rhymes as well as verse internal rhymes, and so on could be used in actor classes. In a similar fashion, assigning emotion labels to verses or lines could provide useful and lively acting examples. Moreover, full text search of lines or fragments of lines together with playing the results from each of the different recordings of a play may help for the remembering work.

The *JAM!* corpus and associated descriptors constitute a very valuable resource from the linguistic perspective, such as prosodic studies, both at a macro-rhythmic level (turns, location of silences and pauses, acceleration and deceleration of the speech rate) and at a micro-rhythmic level (location of group accents, variation of the melody, and so on).

In a broader perspective, as [30], we think that "digital archive enables the user/contributor to interact with the recorded performance and to establish new narrative lines". A wide variety of tools like the ones presented in this paper are necessary to help such new practices to emerge.

5. Acknowledgments

Part of this work has been supported by the European Commission under contract FP-020726, Knowledge Space of semantic inference for automatic annotation and retrieval of multimedia content (K-Space).

6. References

[1] Allen JF (1983) Maintaining knowledge about temporal intervals. Commun ACM 26–11:832–843.
[2] Arndt R, Staab S, Troncy R, Hardman L, Vacura M (2007) COMM: Designing a Well-Founded Multimedia Ontology for the Web, K. Aberer et al (Eds.): ISWC/ASWC 2007, LNCS 4825, pp 30–43
[3] Artale A, Franconi E, et al (1996) Part-Whole Relations in Object-Centered Systems: An Overview. Data & Knowledge Engineering 20: 347–383
[4] Aubert O, Champin PA, Prié Y (2006) Integration of semantic web technology in an annotation- based hypervideo system. In: Proceedings of Workshop on Semantic Web Annotations for Multimedia (SWAMM'06), Edinburgh
[5] Aubert O, Prié Y (2005) Advene: active reading through hypervideo. In: Proceedings of ACM Conference on Hypertext and Hypermedia, pp 235–244, Salzburg, Austria
[6] Auffret G, Carrive J, Chevet O, Dechilly T, Ronfard R (1999) Audiovisual-based hypermedia authoring: using structured representations for efficient access to av documents. In: Proceedings of ACM Hypertext '99, Darmstadt, Germany[7] Bailer W, Schallauer P (2006) The detailed audiovisual profile: Enabling interoperability between MPEG-7 based systems. In: Proceedings of 12th Multimedia Modeling Conference, Beijing, China, pp 217–224
[8] Beaudoin V, Yvon F (1996) The Metrometer: a tool for analysing French verse. Literary and Linguistic Computing, 11–1
[9] de Berg M, van Kreveld M, OvermarsM, Schwarzkopf O (2000) Computational Geometry, Second Revised Edition. Springer-Verlag. Section 10.1: Interval Trees, pp. 212–217.
[10] Bloehdorn S et. al. (2005) Semantic Annotation of Images and Videos for Multimedia Analysis. In:Proceedings of ESWC, Heraklion, Greece, pp 592-607
[11] Caillet M, Carrive J, Roisin C, Yvon F (2007) Engineering Multimedia Applications on the basis of Multi-Structured Descriptions of Audiovisual Contents. In: Proceedings of the int. workshop on Semantically Aware Document Processing and Indexing, ACM, pp31-40
[12] Garcia R, Celma O (2005) Semantic Integration and Retrieval of Multimedia Metadata, In: Proceedings of the 2nd European Workshop on the Integration of Knowledge, Semantic and Digital Media
[13] Glass, J, Hazen T, Cyphers S, Malioutov I, Barzilay R (2006) Progress in spoken lecture processing. In: Proceedings of the Iint. Conference on Spoken Language Processing, Pittsburgh
[14] Gravier G, Yvon F, Jacob B, Bimbot F (2002) Sirocco, un système ouvert de reconnaissance de la parole. In: Proccedings of JEP'02, Nancy
[15] Hardman L, Obrenovi Ž, Nack F, Kerhervé B, Piersol K (2008) Canonical processes of semantically annotated media production. Multimedia Systems, 14:327–340.
[16] Hauglid JO, Heggland J (2008) Savanta - search, analysis, visualisation and navigation of temporal annotations. Multimedia Tools Appl. 40–2:183–210
[17] Hobbs JR, Pan F (2006) Time Ontology in OWL, W3C Working Draft 27 September 2006, http://www.w3.org/TR/2006/WD-owl-time-20060927
[18] Hunter. J (2001) Adding multimedia to the semantic web - building an MPEG-7 ontology. In: Proceedings of the 1st int. Semantic Web Working Symposium SWWS'01, Stanford
[19] Jewell MO, et al (2005) OntoMedia: An Ontology for the Representation of Heterogeneous Media, In Proceedings of the Multimedia Information Retrieval workshop, SIGIR, Brazil
[20] Knublauch H. et al (2006), A Semantic Web Primer for Object-Oriented Software Developers, W3C Working Group Note 9, http://www.w3.org/TR/sw-oosd-primer/
[21] Koide S, Takeda H (2006) OWL-Full Reasoning from an Object Oriented Perspective. The Semantic Web – ASWC 2006, LNCS 4185, pp 263–277
[22] Liu KY, Chen HY (2005) Exploring media correlation and synchronization for navigated hypermedia documents. In: Proceedings of the 13th annual ACM conference on Multimedia, Singapore
[23] Madhwacharyula CL, Davis M, Mulhem P, Kankanhalli MS (2006) Metadata handling: A video perspective. ACM Transactions on Multimedia Computing, Communications and Applications, 2–4
[24] McGuinness DL, van HarmelenF (2004) OWL web ontology language overview, W3C Recommendation 10 February 2004, http://www.w3.org/tr/owl-features/
[25] Pan F, Hobbs JR (2005) Temporal Aggregates in OWL-Time. In: Proceedings of the 18th int. Florida Artificial Intelligence Research Society conference, FLAIRS-2005, Clearwater Beach, Florida, AAAI Press, pp 560–565
[26] Ronfard R, Thuong TT (2003) A framework for aligning and indexing movies with their

script. In: Proceedings of IEEE International Conference on Multimedia and Expo, Baltimore
[27] Troncy R, Bailer W, Hausenblas M, Schlatte R (2006) Enabling multimedia metadata interoperability by defining formal semantics of MPEG-7 profiles. In: Proceedings of the int. Conference on Semantic and Digital Media Technologies, Athens, pp 41–55
[28] Troncy R, Carrive J, Lalande S, Poli JP (2004) A motivating scenario for designing an extensible audio-visual description language. In: Proceedings of CORIMEDIA'04, Sherbrooke
[29] Tsinaraki C, Polydoros P, Christodoulakis S (2004) Integration of OWL Ontologies in MPEG-7 and TV-Anytime Compliant Semantic Indexing. In: Proceedings of the 16th int. conference on Advanced Information Systems Eng. (CAiSE), pp.398–413
(30) Vanhaesebrouck K (2007) Digital heritage and performance. In: Image [&] Narrative - ISSN 1780-678X, Open Humanities Press,
http://www.imageandnarrative.be/inarchive/digital_archive/vanhaesebrouck.htm