

Ecrins: an audio-content description environment for sound samples

Yann Geslin, Pascal Mullon, Max Jacob
INA-GRM, IRCAM

Abstract

The Ecrins project aims at providing an online sound sample database, accessed through high-level browsing features based on audio content-related information. This information is founded on an automatic analysis of the sound compared to textual descriptions made by the user himself. This paper describes two main aspects of the project: the way the sounds can be described by the user, and the data-based model that makes the sound accessible through these descriptions.

1 Introduction

Mass storage devices have increased their capacities and performances and are now affordable to such a point that it is conceivable for a sound designer (sound engineer, composer, etc.) to store and keep accessible hundreds of hours of samples and sound files. The problem of manipulating such a huge amount of sound files is less a question of keeping track of the production conditions (place, date and use) which features are generally provided by conventional database systems, than finding a right sound for a specific purpose, or comparing sounds properties and differences.

Within the scope of the Ecrins project, research teams, developers and industrial partners gathered their knowledge in order to conceive, realize and disseminate a system providing an online sound database. The prospective aspects of the project mainly consist in the elaboration and implementation of descriptions of the audio content itself: whereas most of existing databases of samples provide classical catalog-like documentation, we focused on elaborated, machine-readable textual or quantified descriptions, some of them being automatically extracted from the signal. Research works ranged from study of textual descriptions to signal processing, and analysis of cognitive and perceptive aspects of sound hearing.

In a second phase, these different approaches had to be organized into a generic logical structure, on

which the development of a coherent, operational system could be based. A logical model, permitting the implementation of most of the proposed audio descriptors, and the integration of innovative features (such as automatic classification), was conceived. The whole set of both descriptors and relations between them then constitutes a metadata network allowing a multi-axial, dynamic description of the audio material.

This project has joined for 18 months the forces of two research institutions and a commercial firm, respectively the IRCAM (Institut de Recherche et Coordination Acoustique Musique) for the signal and perceptive-related sound analysis, features already experimented in the "Studio On Line" project (Wöhrmann 1999, Ballet, et al. 1999), the GRM (Groupe de Recherches Musicales from the National Audiovisual Institution) for the user sound description, classification and interface experience, and DIGIGRAM, in order to include the project in a finalized application. The project has been supported by the CNC foundation (Centre National du Cinéma).

2 Audio content descriptions

Basically, sound contents can be seen in two separate ways, that we will call high-level and low-level:

- From the user point of view, a sound has perceptive properties (loudness, pitch, timbre) as well as emotional and contextual use aspects (e.g. pleasant, disagreeable). But the definitions and qualities terms used to describe a sound tend to be infinite. So it is necessary to order the descriptions in a hierarchical way, and keep constant the system and variable types that are used.

- From the technician point of view, a sound is an audio signal which digital samples can be analyzed with different tools to extract some parameters or mathematical abstractions, like its spectral representation. One problem is to reduce the amount of computation data and a second to have a correlation with some perceptual characterization of

the sound properties. Some of these operations can now be almost automatically done.

Finally, the main problem is not only to find and improve tools to provide a response to the two domains, but also to find a relationship between both sets of characterizations. In Ecrins, this will be done by specific properties of the data base organization.

2.1 A general approach to sound description: taxonomy and categories

To order the elements describing some characteristics of the world, one have at the same time to describe each element that belongs to categories, and to built a representation system that explains the relationship between the categories and elements; this is called a taxonomy.

We decided to split us in two teams. The one starting with the description of sounds, to identify descriptive viewpoints and then organize the relation between them; the second starting to built a preliminary taxonomy establishing every possible relation and then linking the categories and elements. We had then to make a fusion of the two approaches.

2.2 Categories: the Spheres

How do we describe a sound perception? We mainly divide it in three domains and focus on one or the other, depending of the situation and necessity. A sound can be heard as a cause, for example, “this is the sound of a train, a violin, a voice”. It can then be heard for its semantic properties: the train is arriving in the station, the violist plays not in tune), the voice is crying (fear). And finally, in some cases, we do have a “technician” way of hearing, that is being listening to some pure sonic qualities: the train sound has a doppler effect, the violin note is a “A3”, the voice is reverberated.

We have called these three perception modes *Spheres*, because their domains are more complex that just dimensions and because there is some overlap between these descriptive viewpoints. We have then the *Causal Sphere*, the *Morphological Sphere* and the *Semantic Sphere*, to which we have added a *Genetic Sphere* to allow a free classification upon production conditions (e.g. production name and date, format, CD number, etc.).

Of course, we generally name a sound by its cause more than its acoustics properties and it is not realist to compare the sounds with their causes; but in any case, naming the causes is the first step of our perception, and remains always useful. That is why sound databases are first organized through a causal description and classification.

It is then difficult to ask a user to describe every qualities of a sound, and furthermore, some perceptive aspects do change in the time, depending on the context, the conditions of the listening session,

etc. And even if people agree on some characterization of a sound, basically the pitch, loudness, and primary timbre, they have no reasons to give the same emotional attribute to a sound. So we have to avoid a fixed classification, and find a universal but highly individualized and dynamical system.

The causal Sphere

It is obvious that it is impossible to build an exhaustive list of every natural and artificial cause that can produce a sound. There is also some doubt about the effective cause of a sound, e.g.: is this violin a real violin (a perfect one or a very bad)? or is it a synthetic sound, a “sound-like” kind of perception? So, we had to explore different way of ordering the description possibilities of the world. This research was conducted by Romain Leblanc who chose a syntactic structuration approach to this problem (Leblanc 2000)

The fundamental idea is to built a kind of description set “imitating” the grammatical construction of sentences. Three members were established as: *actor*, *action*, and *complement*. Then, describing the cause or event that produces a sound consist in defining an actor, an action, an object, and the way the action has been realized.

-Actor: is the subject, what or who makes the action.

- Action(s): are described by infinitive verbs.

- Complements: the main complements are associated to the prelude verb; they are generally object complements, that is the being or thing on which the action is performed.

We can also add some circumstantial complements (adverbial-), the one which are not essential to the sentence construction, but will give some precisions to the action: place, time, cause, way of, middle, goal.

The sound description is then organized in an horizontal structuration of members or complements:

High level							
Actor	Action	Main complement	Circumstantial complements		Semantic		Environment
			Middle	Way	Cause	Goal	Place Time

This organization allows the user to make some distinction between causal and middle conditions; for example, if we hear a horn, we have to tell if it happens during a wedding fest or in a traffic jam.

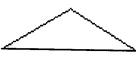
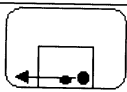
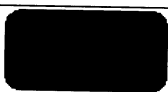

Nom et catégorie								
Haut niveau								
Acteur	Action	Complément essentiel	Compléments circonstanciels		Sémantique		Environnement	
			Moyen	Manière	Cause	But	Lieu	Temps
Avion		Passer		Rapidement			Extérieur	
Bas niveau								
Concernant les différentes actions		Critères morphologiques				Critères techniques		
		Concernant la totalité du son						
		5-10s					Qualité : Bonne Emplacement actuel : CD « Exemples d'applications », page 1 Provenance : Hollywood edge	

Figure 1 — syntactic description: The actor is the plane, the action is flying, the way is very fast, the environment outside, with a Doppler effect

The figure 1 shows an example of this syntactic construction for a jet-plane sound, from a prototype software designed to test (in french) a few sound descriptions.

Beside this horizontal description, there is a vertical decomposition for each sentence element, in a conventional tree arborescence. For example, the access tree for a cock:

Actors						
Humans ▶						
Animals ▶	Wilds ▶					
Objects ▶	Domestics ▶	Birds ▶				
		Reptiles ▶				
		Fishes ▶				
		Poultry-yard ▶	Hen			
		Insects ▶	Cock			
		Rodents ▶	Duck			
			Goose			

The causal description of a sound is then either a simple word naming the main cause that produced the sound, or a whole “sentence” describing most of the factors that has contributed to generate the sound.

The morphological Sphere

Up to now, a lot of work has been done to describe and classify the sounds produced by the occidental music instruments. And recently in the SOL project, a description and classification of some timbre qualities has been successfully experimented. However, most of the sounds we hear are not instrumental sounds, and have to be described with other definitions than the note pitch, instrumental timbre, loudness and “playing mode”. During the sixties, Pierre Schaeffer and his team at the GRM has worked on the perception of sounds considered for their general perceptive properties. He called this capacity “écoute réduite” (reduced hearing) and elaborated a set of definitions to give qualities to sounds properties and classify them (Schaeffer 1966,

Chion 1983). A few years later, a collection of typical sound examples were realized to illustrate this theory (Schaeffer 1967).

Emmanuel Deruty conducted the main research on a rationalization of these definitions and the possibility of a dynamical structuration of the description viewpoints.

There was first the necessity to discuss once more these terms and verify their quality and pertinence. In particular, some of them, although very innovative (e.g. “grain”, “allure” (rate), “profil” (profile)) are very often subject to interrogations or confusions and have to be better circumscribed. And another aspect is that these description tools tend to promote some averaged values, in what Schaeffer calls an “equilibrated sound”. In sound production, one may use every kind of sounds. It would then have been a pity not to be able to classify half of the sound material, because of its length or its complexity. We had also to verify the possibility of relationship between the sound descriptions and the automatic analysis: generally, the automatic analysis has been efficiently used only for very short and simple sounds. And we finally tried to reduce the number of categories and elements to get a synthetic but efficient map of every sound qualities.

The morphological sphere is divided in two descriptors sets, main and complementary. The main descriptors are: the duration, the dynamic profile (amplitude evolution), the melodic profile (pitch evolution), the attack (in amplitude and spectrum) the pitch (either note pitch or area), and the spectral distribution (brilliance). The complementary descriptors are the space (position and movement) and the texture (vibrato, tremolo, grain).

For most of the descriptors there is a choice between full or restricted sets of values. For example, in the dynamic profile, the restricted values are: flat, increasing or decreasing slope; to which the full set adds the two and three slopes combinations (e.g. attack-sustain-release), plus a drawing representation capacity.

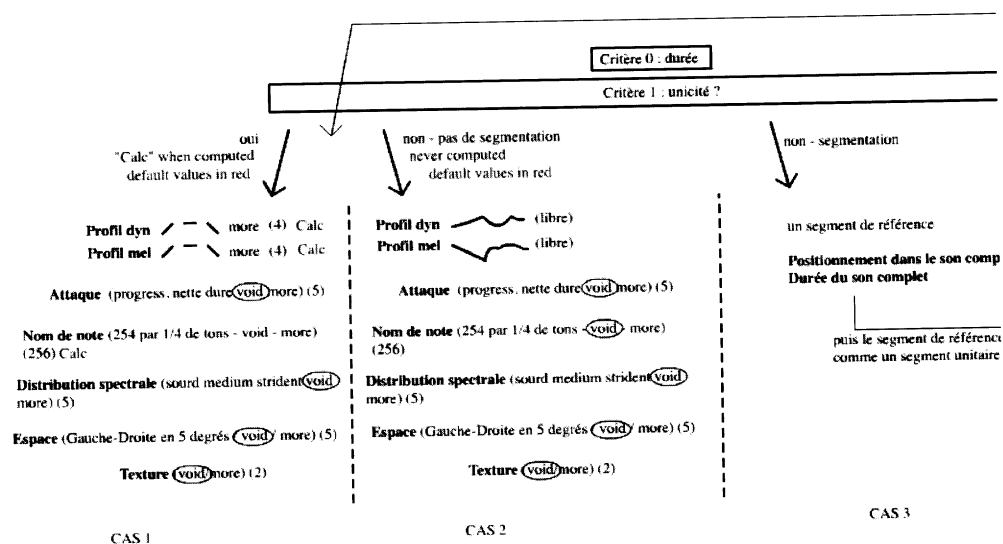


Figure 2 — three description cases: automatic, manual, segmented

Finally, as we wanted also to avoid the user to have to describe every quality of the sound, which is tedious, we decided that the user could only have to verify the composition of the sound to make a distinction between unitary sounds and complex sounds. A unitary sound is defined as a sound that cannot be divided, either in time (succession) or spectrum (polyphony). The unitary sounds has the possibility to be automatically analyzed by the low-level part software; the complex sounds will have to be described by the user in order to verify that there could be a true relation between the description and the automatic analysis. The idea is that the more simple the sound is, the more automatic it has to be described. And if the user chooses complex sounds, he will have to spend more time to verify the pertinence of the analysis. For example, in the case of a very long and varied sound, the user has to make a segmentation or select a representative part of the sound.

Here are the different stages of the morphological sound description: - The first step is the verification of the duration of the sound. Although this is automatically done, the user has the possibility to give a personal estimation (some sounds seems to last more than the reality).

When the users morphological description is achieved, the descriptors can be compared to the low-level information to point matching properties of the sound description.

- The second step is the verification of the uniqueness of the sound. This is absolutely made by the user, and will decide how the sound is described.

- If the sound is unitary, the user will have to verify the restricted values proposed by the automatic analysis for the following descriptors:

- dynamical profile: flat, increasing or decreasing.
- melodic profile: idem

- attack (long, medium, sharp)
- note pitch
- spectral distribution (dark, medium, strident)
- space (fixed stereo localization)
- texture (no texture)

If the value the user has to jump to a manual description; for example, if the dynamic profile is not a simple slope, he will have either to select a more complex profile or to draw an approximated slope.

The figure 2 shows the diagram of the morphological categories, and the different steps of the sound description:

When the users morphological description is achieved, the descriptors can be compared to the low-level information to point matching properties of the sound description.

When the users morphological description is achieved, the descriptors can be compared to the low-level information to point matching properties of the sound description.

The semantic Sphere

This part is at the moment not definitely established, because it is more subject to individual appreciation and has to be verified in the applicative situation. Roughly, we will have main descriptive viewpoints, in which the user will bring his own words and appreciations:

- perceptive categories
sight, ear, taste, touch, smell.
- animal or human production
speak (cry, etc), animal, corporal sounds

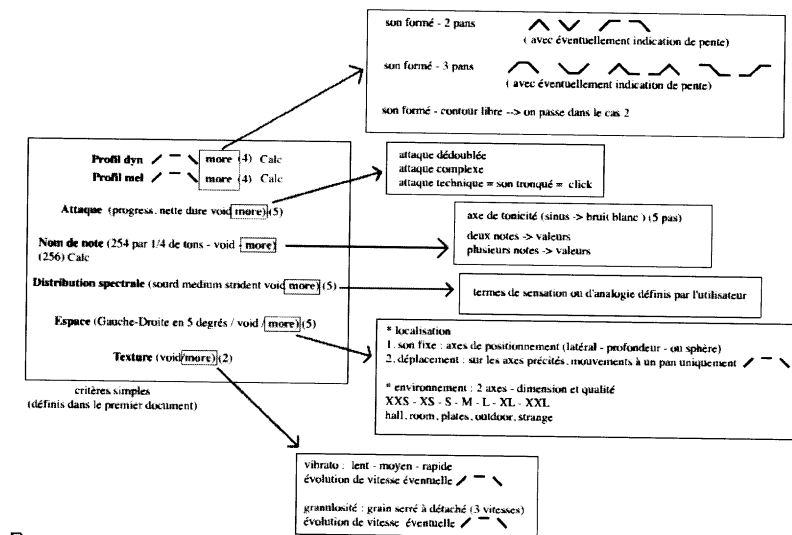


Figure 3 —descriptors full set - automatic and manual description diagram

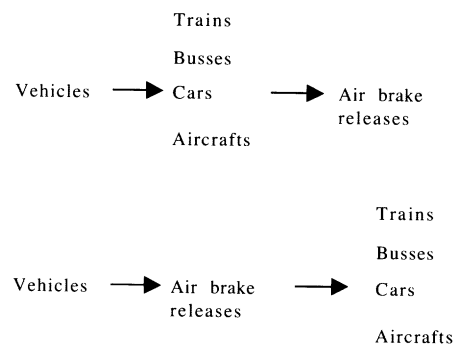
- substance characteristics
- appearance, origin (air, fluid), behaviour (elastic,bouncing), qualities (heavy, hollow)
- psychological behaviour
human behaviour (mad, delirious)
feeling (alarming, charming)
aesthetic (beautiful, hideous).

One should not forget that the user might choose arbitrary appreciations, as he is not in a scientific situation but a personal viewpoint attitude. For example, one user can feel that a sound is alarming, and the other not, or may choose a fanciful description, for fun, where everybody would suppose to agree on the perception.

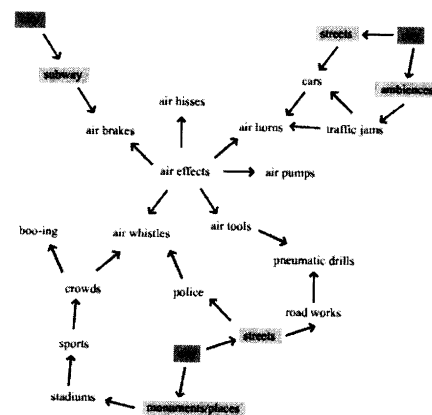
2.3 Taxonomy: a dynamical structured organization

One aspect of the classification of the causal categories into a tree and hierarchical structure is that it cannot solve every relation we want to describe. For example, if we have a source like a car, we will place it in the “car ” class which belongs to the “vehicle ” class, a more general and important category. But, if we speak of a city, with traffic jam, the car belongs to the street which is part of the city. So we have different point of view for the same sound. One answer is to combine the tree structure with transversal bonds, but we won’t avoid ambiguous responses.

For example, these two relationships are equally acceptable:



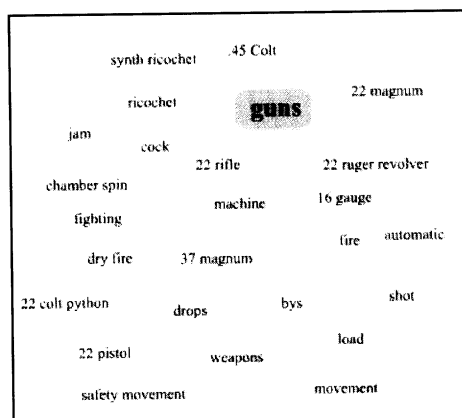
And if we look at the word “city ”, we will find a lot of relationships between very different kinds of categories:



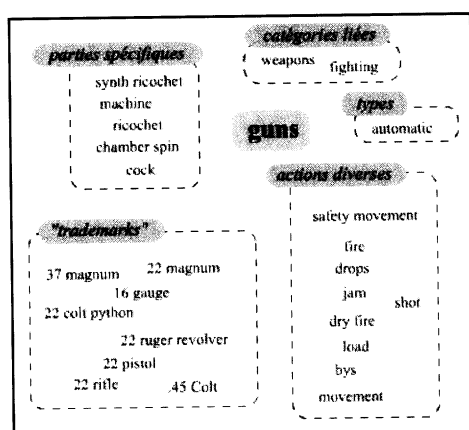
The solution is to allow the user to build as many bonds as necessary between the words that represent the sound sources, and not to preliminary define the types of relations. The word used to characterize first a sound is placed in a center, and then the other complementary words are placed as satellites. It is

then possible to group some of the terms, if necessary or useful, even arbitrary.

For example, the term gun can be associated to many complementary terms:



And these terms may also be organized if necessary, but of course, this will be a user choice:



This organization is a kind of open structuration, and has to be solved by the database structure. The biggest obstacle is that each time the user wants to change one of the relationship between the terms describing one particular sound, he would have to examine and compute again the consequences on the whole database; which seems to be unrealistic and inaccessible.

2.4 Automatic analysis

The low-level analysis is in itself the subject of a whole technical paper concerned by digital signal representation, analysis, parameters extraction and information reduction.

Mainly, the signal will be sampled through a FFT window (40ms size, every 10ms), which main extracted parameters will be the spectral centroid, the spectral spread, the spectral skewness, the spectral slope, the loudness, the sharpness, the timbral width, the harmonic parity, the harmonic tristimulus, the harmonic deviation, the harmonicity, the attack time, and the temporal centroid.

3 Logical model and implementation

The entire set of audio content-related metadata to be managed by the final Ecrins system aims at integrating the various approaches described in the preceding section, including numerical information extracted from the signal, like estimation of fundamental frequency, attack time, etc. There obviously exist logical links between these descriptors of various types. Finally, the set of all Ecrins descriptors constitutes a metadata network for which we have to give a logical generic structure taking into account the relationships between descriptors.

Then, we have at least three levels of data:

- *audio samples*;
- *descriptors values*, which are values taken by a descriptor for a given sample (fundamental frequency = 440 Hz, source=car, etc.);

descriptors considered for themselves, as functions of samples (fundamental frequency, dynamic profile, source, etc.).

These three levels of data can be summed-up in the equation: $descriptor\ value = descriptor(sample)$. In the space of descriptors (considered as a set of functions), relations from descriptor to descriptor can be managed regardless of the particular values taken by these descriptors on each sample. In the following, we describe the logical modeling which we used to implement this space of descriptors in the Ecrins system. This leads us to define two other kinds of data (*descriptive viewpoints* and *terms*), corresponding to increasing degrees of abstraction.

3.1 Classes

Basically, descriptors are either structured textual descriptions¹ (classification of samples into labeled "directories") or numerical metadata. Textual descriptions are in fact nothing but qualitative, Boolean information: a sample *is* or *is not* a car sound or, in other words, does or does not belong to the set of samples featuring a car sound. The notion of *set of samples* then naturally appears to be a central concept for any classification of audio data.

However, since we are mostly interested in audio content description, the notion of set of sample only makes sense if this set has been constituted on content related criteria. This is why we introduce the notion of class: a *class* is a set of samples gathered on the base of an audio content criterion.

¹ Free textual descriptions (commentaries on samples) will also be provided in the Ecrins project; they will be used to retrieve samples using basic full-text searches. Automatic analysis of such data would imply a linguistic approach, which goes beyond the scope of the project. Thus, we consider commentaries as unstructured information; for this reason, they cannot be integrated into our theoretical model. They do not appear in the present study.

“A3”, “Pianissimo to fortissimo”, “Brilliant” are examples of classes, because they provide information about the audio content (the samples belonging to these classes share the same note, dynamic profile, etc.).

Samples can also be gathered on the basis of non-content-related considerations; this can be very useful for a user (for example: “All my samples recorded on 12/12/2001”, “Samples shared with Kylie”), the resulting set of samples is not a class, and is called a *folder*.

Classes and folders being sets, inclusion can exist between two of them. These inclusions are very meaningful for the user (“car sounds” are included into “machine sounds”), and thus very common in existing classifications. Using these relations of inclusion, classes and folders can be presented in a tree-structure allowing hierarchical browsing:

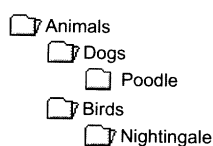


Figure 4 — Tree structure of classes

This navigation process, based on two elementary actions (opening/closing a tree node and viewing the content of one particular tree-node) is very similar to what graphical file system managers currently offer. It is thus familiar to most users.

Nevertheless, classes are not directories: especially, one sample can belong two more than one class. Intersection between classes is thus a meaningful operation, which the Ecrins system also provides.

3.2 Descriptive viewpoints

Classes reveal a great plurality: some of them can be organized in open lists where no bound nor order can be defined (for example: list of all possible sources of sounds, including “all” machines, animals, etc.). On the contrary, other qualitative descriptions are obviously ordered (musical notes for example), and thus constitute a scale. It is obvious that the classes corresponding to the different values of such a scale have to be gathered in some way, and that the system has to know the order relation existing between them. Classes corresponding to sources of sounds (“Machines”, “Animals”, etc.) should also be gathered, and separated from classes corresponding to morphological information (“Pianissimo to fortissimo”, “Fortissimo to pianissimo”).

So it finally appears that descriptors themselves have to be classified into high-level structures corresponding to the various cognitive dimensions along which the audio material can be described. Each of these structures can be seen, both as the

modeling of one general characteristic of audio contents (“Dynamic profile”, “Source”, etc.) and as the set of *all* descriptors which values are descriptions of this audio characteristic. These structures are what we call *descriptive viewpoints* (or simply “*viewpoints*”).

Descriptive viewpoint notion is partly based on the notion of spheres exposed in the first section of the document, a sphere being also a cognitive context in which a description is elaborated. One difference between spheres and viewpoints is that each viewpoint is supposed to be perfectly separate from all others. Indeed, the descriptors described in the first section of this document have one important property: each of them only gives information about *one* audio characteristic (for example, we giving information both on duration and spectrum, or both on audio source and temporal envelope, etc.). For this reason, a descriptor can only belong to one descriptive viewpoint, and various descriptive viewpoints can be seen as independent: there is no logical relation between values taken by a sample for two descriptors belonging to two different viewpoints. In other words, the description of each sample (e.g. the values taken by all descriptors for this sample) can be organized in a set of cognitively independent sub-descriptions corresponding to the various descriptive viewpoints.

It appeared to us that viewpoints quite easily match with a hierarchical organization based on an is a kind of meaning (“Dynamic profile” and “Duration” viewpoints are kinds of the “Morphological description” viewpoint). Since description viewpoints are not very numerous, such a tree structure seems sufficient to allow simple, user-friendly presentations of viewpoints. Besides, using viewpoints, descriptors (especially classes) can be accessed by the user via a preliminary cognitive filtering (“Show me only morphological descriptors”, “Show me the list of sound sources.”).

Thus, the hierarchical structure of viewpoints, which leaves contain descriptors, can be combined with the tree-structure of classes, leading to trees which root nodes are viewpoints (in bold and identified by the icon in Figure 5), and which final leaves are descriptors. In such a presentation, the user first chooses the viewpoints that are relevant for the kind of search he is performing, and then obtains the list of descriptors which are meaningful for the samples he might be looking for:

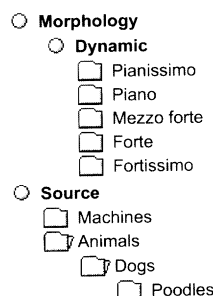


Figure 5 — Inclusion of class trees into viewpoint trees

At this level, intersection between classes belonging to different viewpoints is a typical use-case, leading to samples lists defined by multiple criteria such as “pitch is B3, instrument is violin and dynamic is pianissimo”.

Viewpoints also offer a useful structure for guiding the user during manual classification process: the list of viewpoints (with their descriptors) can be displayed to the user, who is invited to provide one description of the new sample on each of these viewpoints.

According to relations existing between descriptors within a viewpoint, specific kinds of viewpoints can be isolated, with additional interesting properties.

Many viewpoints are *exclusive*, which means that each sample can only have one value among all the descriptors of such a viewpoint. In this case, automatic classification, including preliminary learning features, can be attached to the viewpoint. For example, the viewpoint “Dynamic”, containing classes “Pianissimo”, “Piano”, “Mezzo forte”, is exclusive (no sound can be both “Piano” and “Mezzo forte”), and automatic classification modules could place new samples in one of these classes. During manual classification, exclusivity of viewpoints can be automatically taken into account to avoid classification errors (such as, for example, associating one sample with two dynamic profile classes).

Furthermore, an exclusive viewpoint can be *ordered*, which means that an order can be defined in the list of classes it contains. In this case, the audio characteristic represented by the viewpoint is often of continuous nature. This continuum might be estimated by numerical descriptors belonging to the viewpoint. On the other hand, this continuum might be split in segments, each of which being associated to a class of the viewpoint. A typical example is pitch, which is basically continuous, and can be estimated by fundamental frequency estimators, but which is often expressed through notes, which are a scale of ordered classes. “Pitch” viewpoint thus contains both classes and numerical descriptors, and gathers the various forms that metadata on pitch can take. This gathering permits the integration of conversion features from descriptor to descriptor (for

example, automatic or semi-automatic correspondence between musical notes and fundamental frequency), and leads to a uniform presentation to the user.

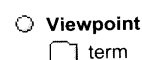
3.3 Terms

Classes, descriptors and viewpoints are identified by a label, which consists in a small textual expression (“A3”, “Mosquito flights”, etc.) used in all user interfaces. It appears that the labeling of descriptors, viewpoints and numerical descriptors does not lead to particular issues, because these elements are not numerous and, being of technical nature, often benefit from a well-known, unique denomination.

On the contrary, the denomination of classes, especially when they are referring to elements of the real world used in audio scene descriptions, leads to theoretical problems of two kinds. First of all, we face issues of linguistic nature, like polysemy and synonymy. Within the scope of our work, where natural languages problems were not to be tackled, we only considered simple solutions mainly consisting in precisions included in the labels themselves (for example, “Piano (dynamic)” versus “Piano (instrument)”).

But, regardless of linguistic issues, labels reveal insufficient to identify uniquely each class: does a class labeled “Bus” contain samples where buses are passing in the street, or does it contain samples featuring the sound ambiance inside a bus, with talks and background engine sounds? In other words, is “Bus” considered as the source of the sound, or as some environmental information? As we have seen it, this ambiguity is solved by the notion of viewpoint: both “Environment” and “Source” are viewpoints, that could contain a “Bus” class. Each of these “Bus” classes would contain different samples (“Bus” as the main source versus “Bus” as a background audio element). Thus, though they share the same label, those two classes are different. Nevertheless, they obviously share one common information (the object “Bus”), which is a textual representation of an object of the real world. This is what we call a *term*.

According to this system, a class is identified, not by a term, which can possibly play several different descriptive functions (cause, environment, etc.), but by a couple (viewpoint, term), where the viewpoint indicates the particular descriptive function of the term. As we have seen, this couple is presented to the user in the form:



One of the first applications of terms is multilanguage management: considered as the representation of an item of the real world, a term is not associated to a particular language: “Bicycle brake” and “Frein de bicyclette” are one single term.

The set of all terms potentially constitutes a multi-language semantic thesaurus, giving some kind of “representation of the world” on which one can rely in order to describe concrete sound samples, in a manner that can approach the proposals of natural-language-like descriptions proposed in the first section of this document. In order to be useful, especially for browsing, this thesaurus thus has to be completed with links between terms. It then constitutes a kind of semantic network.

First of all, hierarchical links can be defined between terms. As opposite to what has been seen with viewpoints, it appears to be impossible to reduce those links to one single type. For example, “is a trademark of”, “is a component of” are different term-to-term links, both of hierarchical nature. Moreover, the types of the links themselves can be hierarchically organized: “is a trademark of” link is a kind of “is a subset of” link.

Non hierarchical links also exist between terms. Types can also be defined on those links: “is a behavior of”, “is associated to” are examples of non-hierarchical link types.

This typology of links can be used by the system to perform automatic inferences. For example, considering the fact that “Ferrari” term is linked to “Car” term with a “is a trademark of” link, the system, knowing that “is a trademark of” link type inherits from “is a subset of” link type, could automatically infer that Ferrari is a subset of car.

Applications of such a system, quite heavy to implement, have not been set up in the Ecrins system. Nevertheless, the current database model already takes these concepts into account, so that related features can be seen as possible evolutions of the system. Such applications mainly consist in rich browsing features among terms, like those proposed in the first sections of this document for audio scenes description. Starting from the term “Car”, one user could ask the system for the list of trademarks of cars, while another user would ask for the list of components of cars, both of these results being lists of terms. For each term, the classes using this term as label can be presented or, more efficiently, we could display the list of corresponding viewpoints: assuming that our second user chose the term “Horn” among the list of car components, he would obtain a choice list like: “Horn as environment”, “Horn as source”, etc

Choosing one of these term-viewpoint combinations determines one unique class, and gives access to the corresponding sample list.

4 Technical aspects

The Ecrins system is implemented as a series of Internet/intranet services provided by a server platform located at Ircam. The user interface mainly relies on the use of html frames, javascript procedures

being used to manage typical client features like selection of items or synchronization between frames. These technologies allow compatibility with various kinds of clients (Linux, Windows, Mac, etc.).

At server side, an application server embeds a series of java servlets, each of which being dedicated to one particular high-end service, directly callable by the client (login, provide all classes of a given viewpoint, provide all samples of a given class, provide one sample as an audio stream, etc.). Servlets also take in charge the construction of the html frames (including class and viewpoint trees). Servlets thus take in charge two functions:

- constitution of high-level complex data provided to client via high-end services;
- dynamic construction of the user interface.

In the internal structure of servlets, we distinguished these two functions, in order to be able to use the first one independently from the second one or, in other words, to be able to use the system without using the html user interface. This could lead to an API, possibly relying on an XML formulation of data, and featuring the same services than those offered to the final user. This programming interface could allow automatic, massive enrichment of the sample database, or integration of the Ecrins features into an independent client application.

These servlets mainly rely on a relational database, managing all data but audio material itself. In particular, the database contains all metadata and descriptors related data; it thus implements a logical data model deriving from what has been presented in the preceding section. Classes, viewpoints, terms, are stored in the relational database. This database is wrapped by a set of java modules hiding the complexity of data relational modeling, and providing packaged services usable by servlets. These services especially consist in samples and descriptors browse and search features.

Audio samples are stored on an external storage device, accessed by a dedicated server running C++ modules providing all audio processing, which means: all audio format conversions (file format, sampling rate, etc.), numerical descriptors calculation, and automatic classification processing. Finally, high-end services provided by servlets are based on calls to both database modules and audio processing modules.

For CPU load issues, we also isolated the streaming process, which is based on an on-the-fly mp3 encoding, from other audio material processes. We then have four logical servers, respectively dedicated to audio material management, database management, streaming and web services. Dialog between these four logical entities is done using the Corba technology. This allows a certain flexibility in the way logical servers can be distributed on

hardware devices: some users, dealing with small volumes of data, or not interested in streaming feature, could reduce the hardware investment, and run the same system on a two machines platform and with lighter storage system, for example.

References

- Wöhrmann, R., G. Ballet 1999. "Design and architecture of distributed sound processing and database systems for web-based computer music applications", *Computer Music Journal*, vol 23, Number 3, p.73-84.
- Ballet, G., Borghesi, R., Hoffmann, P., Levy F. 1999 "Studio Online 3.0: An Internet "Killer Application" for Remote Access to IRCAM Sounds and Processing tools" in JIM99 - Journées musicale proceedings.
http://www.ai.univ-paris8.fr/~jim99/actes_html/BalletJIM99.htm
- Leblanc, R. 2000. "Elaboration d'un système de classification pour sons non instrumentaux" INA, Paris
- Schaeffer, P. 1966. "Traité des objets musicaux, essai interdisciplines". Seuil, Paris.
- Schaeffer, P. 1967. "Solfège de 174pp. + 3CD (english, français, Paris 1967, Ina, Paris 1998.
- Chion, M. 1983. "Guide des objets sonores, Pierre Schaeffer et la recherche musicale". INA-GRM/Buchet-Chastel Paris