Jerome Barthelemy

IRCAM Artistic Tesbed

slide 2 This talk presents the work that has been done by IRCAM in the CASPAR project.

It is structured into three parts.

- the context, i.e. musical production with digital components since the 70s. An example explains what the structure of the objects handled and preserved from IRCAM is and what the risks and good practices for their preservation are.
- the preservation issue, paying particular attention to the definition of Representation Information (RepInfo) and how IRCAM extracts it from existing objects, to Preservation Description Information (PDI) and to the methodology to assess authenticity.
- the work already done and the future work. The use of the MustiCASPAR server, developed by IRCAM, and its implementation is presented in a testbed scenario and then the methodology for RepInfo validation is explained. The last parts concern the current status of the repository and the work that IRCAM proposes to do in the next few years.

slide 3 The IRCAM context is musical production with digital machines.

IRCAM started with the development of audio musical digital processor hardware based in the 70s and until the early years of the 80s, and from the middle of the 80s onwards, it has developed software, principally Max/MPS.

IRCAM makes musical creation using audio digital processing: it has created 4,450 watts since ’77.

Since the middle of the 80s, the problem of preservation has been organized with no formal approach; IRCAM started producing documentation on paper from the 80s until 2000. Since 2002 it has attended to digital storage of documentation and digital objects on the Mustica server.

slide 4 The goal of preservation in this context is not simply to record audio files, but to be able to re-perform the work.

As IRCAM wants to make the interactions between human performers and digital processes possible, not only the results, but also the processes themselves must be preserved.

These processes are a kind of ‘digital musical instrument’: audio effects, reverberation, harmonizers and so on, and they are encoded in the form of ‘software’.

slide 5 For example: *Diademes* was created by M. A. Dalbavie in 1986. It was written for traditional and live electronic instruments.
The live electronic part was performed using Yamaha hardware.
At the moment of its creation the performers preferred to use instruments from the industry since they imagined that it was more convenient for preservation and to avoid risks relating to obsolescence.
Facts have proved that it was not a good choice: since 1995 no new performances of *Diadem*es had been done because of issues related to the FM synthesis component.
A new performance was done in December 2008.

slide 6 [Without entering into details of the live electronic part, the set-up (the effects are on the right part of the image) and the three pieces of hardware are shown on the slide.]

slide 7 The FM Synthesis was the original implementation; it was by Yamaha under their patent. It results in some complex sounds that are not producible by other means.

slide 8 Starting from the above-mentioned example, the general case can be explained.
The main data are: data files, mainly audio; data coming from processes (e.g. in a process, a recorded input is processed and then sent to speakers); information files like sketch plans, instruction files and so on.
The whole set defines a ‘work’ (in addition to the musical score).

slide 9 The general case is much more complicated but it can be simplified as in the example above.

slide 10 As we explained, the process is a kind of ‘musical digital instrument’ that has to be preserved, like for acoustic instruments.
The obsolescence for the process is very fast. Audio files from the middle of the 80s cannot be played at the moment but, for example, the first PCM data, now well-known, can be migrated without any problem.
New performances generally imply a migration, a porting or an emulation of the main process.
The most important question is the level of authenticity obtained after the migration.

slide 11 Several good practices on authenticity were already developed before CASPAR.

One of the most important is to work on some samples in input and to the record audio output after processing it in order to keep track of the intended use of the process.

slide 12 The preservation in IRCAM involves the whole process, not only the recording; the data files, such as audio files; all documents that are related to the work and the logic, i.e. the relationship between all the elements that compose the work.

slide 13 The most important preservation issue concerns the process that is named the ‘patch’.
The patch is really the core of the work; it encodes the logic of the relationship between the different elements (hardware, like microphones or speakers, and data file).
If the patch is preserved, it is also possible to preserve the logic of the work, but it is also possible, in some way, to extract the logic of the work from the patch.
As the process is a kind of software, its preservation could be very difficult. Fortunately, it is a very specific kind of software, most of the time encoded with specific software like Max/MSP, PureData or Reaktor, that are all based on graphical programming and are similar to each other. The work of IRCAM is generally based on Max/MPS.
The major risk is due to the fact that, when there is the need to re-perform, the most important part of the work is done on the patch; this is the reason why IRCAM efforts are focused on the preservation of the patch.

slide 14 The Representation Information (RepInfo) of the patch is extracted as a structure of a block-diagram flow and the semantic of each of its individual elements is already defined in existing documentation that generally has a very normal form.
The methodology consists of reducing the block-diagram flow to algebra. IRCAM has chosen to use an existing language named FAUST (developed by Gram, the center for music in Lyon, France) and that is very concise and sufficiently expressive. IRCAM also stores the semantics of each element, extracting these semantics from the existing documentation.

slide 15 The RepInfo for real-time process is extracted by means of several tools.
The DOC tool extracts the semantics of each individual element that has been stored, from the existing documentation.
The FUNC tool parses the code of each existing process in order to identify the elements that are present in this process, verifies if they are already documented in the RepInfo and provides PDI for the process according to the defined patch ontology. If the RepInfo is missing, a warning is generated, the

elements must be documented and the DOC tool is applied on the undefined elements.
The FILE tool analyzes the global structure of all the provided files and encodes the logic and the relationship between the elements, i.e. the PDI of the entire work, according to the provided ontology.
Finally, the LANG tool re-encodes the original process in an XML complaint data file.
The whole set of objects are stored inside the MustiCASPAR server in the form of XFDU.

slide 16 Ontologies are necessary in order to define the PDI which is required to accomplish the actions explained above.
The PDI is based on the CIDOC-CRM ISO standard.
Ontologies templates, expressed in RDF, are provided for each element which is preserved: work, real-time process (patch, library, function) and documents (booklet, hall program, biography, interview, audio sample, video sample, score and recording).

slide 17 The ontology template for work is an example of these ontologies. It is a template consisting of what is ingested in the repository and its relationship, encoded in the existing relationship of the template.

slide 18 The whole relationship, the semantics and the demands in the ontology template for the patch are defined according to CIDOC-CRM.

slide 19 Authenticity is regarded in CASPAR as a process and Authenticity Protocols (AP) have been defined.
Each AP is composed of several Authenticity Steps (AS) and must be executed at different phases according to the life cycle of the object. Each execution is composed of different Authenticity Step Executions (ASE) and the overall result should lead to an evaluation of the authenticity of the object. The APs are defined accordingly to a Designated Community.

slide 20 Different APs must be attached to different steps in the life cycle of the object, for example: in the case of a migration there is the need to execute a different protocol than in the case of the verification of the existence of an object.
There should be different APs depending on the context: the AP depends on the work in which it is used (for example, for FM Syntesis, several works have been identified where APs should be different according to different composers); APs are also dependent from the point of view of different communities: for the developers of FM Syntesis the AP is a piece of code; for the musical assistant

the FM Syntesis component is something that produces a certain result that has certain qualities.

slide 21 An example of AP is a simple case of the maintenance of an audio file where two actions are to be accomplished: to compute fixity and to verify provenance.

slide 22 A much more complex example is the case of the migration of an audio file from one format to a different one.
The first step is to verify the semantic (often what was supposed to be an audio file, was not really one but a storage file; in these cases where a storage file was used in order to store parameters of files, if you apply migration onto this kind of file you could lose all the parameters entirely).
The second step is to verify that the file is not compressed, and so on.

slide 23 A third example of AP is the case of the migration of an audio effect from AFX1 to AFX2.
According to good practices, an ingest phase has to be defined; this phase takes place through the definition of a list of audio samples in input, the application of the audio effect AFX1, the storage of the audio samples in output, and finally, the definition of the comparison features.
Next, in the migration phase of the file, the migrated audio effect AFX2 has to be applied to the same input audio file and the resulting audio file has to be stored and finally the two audio samples have to be compared.

slide 24 Having defined how IRCAM extracts the RepInfo, defines the PDI and preserves authenticity, it is now possible to enter into the IRCAM testbed scenario.
Its starting point is the occurrence that Max/MSP is no longer available.

slide 25 All the steps of the scenario have been defined in detail.
The first is the ingest of the new work and its components.

slide 26 The second part of this scenario is the notification of the loss of availability for a component.

slide 27 The third part is the search for equivalent components or something else to migrate the components.

slide 28 The fourth sub-scenario is the ingest of a new version of a component and the notification of the Preservation Orchestration Manager, after having verified the authenticity.

slide 29 At this point in the speech, a demo movie shows the use of the MustiCASPAR server for ingestion, the use of the Preservation Orchestration Manager and the association of Authenticity Protocol.

demo This scenario sets the case where a piece of software becomes unavailable and due to this event some components become obsolete.

The starting point is the ingestion of the component and its associated RepInfo.

Considering a component ready to be ingested, a warning may show that its RepInfo is missing, we therefore associate to the component the RepInfo extracted from existing documentation. Different documentation templates and different kinds of source can be managed. (The video shows some examples of source files in PDF for RepInfo and, how the reformed RepInfo is stored in the OAIS package). As a result, the warning regarding the missing RepInfo disappears.

The second step is the generation of the notification when the software able to run the component becomes unavailable.

A user logs into the system and generates the notification about the obsolete component, by searching for the component and then generating the notification. The following phase is the reception of notification by a third user.

The third step is the migration of the component based on the provided RepInfo. This migration is outside the scope of this demonstration.

The fourth step is the ingestion of the new component replacing the old one and identified as the new version of the old one, through different phases: first the location of the obsolete component, then the addition of the new version of it, followed by the generation and association of the RepInfo, its ingestion, and finally the association of an Authenticity Protocol.

The final step is that the new component is available for all eventual work making use of the old component.

slide 30 The RepInfo validation is carried out in 3 steps.

First the completeness of information is validated and the original process from the extracted RepInfo is reconstructed.

The second step is to give proof of its usefulness by reconstructing an equivalent process from the extracted RepInfo, but executed from PureData, that is more or less the equivalent to a migration.

The third consists of assessing the authenticity by comparing the audio outputs in accordance with the defined AP.

slide 31 In other words, the first step is the verification of the completeness of the information from the original process.

The RepInfo can be extracted totally in an automatic manner and the original process can be reconstructed exactly the same as the first one. This is needed since the second validation does not give any proof of completeness.

slide 32 After, there is the proof of usefulness from the extracted RepInfo. It is possible to generate a different process but based on PureData instead of Max/MSP.
There may be some possible losses of information and inconsistencies which require manual adjustments, particularly for certain details, e.g. a comment should no longer be valid with the new version since it refers to elements that have different semantics.

slide 33 Then the AP is applied; it is previously defined, at the ingestion phase, and is based on an input audio file and on an output audio file.

slide 34 Finally, after a migration, the new version can be compared with the old file, according to what you want to hear, for instance by making an audio engineer hear the two audio files, or by any other method of comparison, for example comparing spectrograms.

slide 35 If the simple migration based on RepInfo has not been completely satisfactory, adjustment have to be made, particularly on the sliders, since the semantics of this element is sometimes not well defined.

slide 36 In order to evaluate the current state of repository towards OAIS, IRCAM has taken two kinds of action.
Firstly, a preliminary analysis of the whole content of the repository has been done. Several common file formats including ZIP, DMG, RAR and ISO CD have been found. Moreover, it has been verified that the reconstruction of PDI is possible for the work, but not for every part of each work; this reconstruction can be made partially in an automatic manner.
Secondly, IRCAM has done a test on the migration of a single work. The purpose of the test was to evaluate the amount of work there was of a complete migration for the whole content of the repository.
The most important issue that was identified is that there is information missing regarding provenance and context, so there is the need to identify some people who are able to provide this information in order to complete the migration of the repository.
Some unexpected files have been found for which the RepInfo and the PDI are completely missing, but one can also work on these files applying automatic tools to them.

slide 37 The achievements made by IRCAM consist of the development of the

MustiCASPAR server that is OAIS compliant and that gives the user a great deal of information about the current state of the repository by comparing the state of what he ingested to the ontology templates provided; as a result, warning and more information about what is missing can be generated.
IRCAM has also defined a methodology and tools that can be applied in order to extract RepInfo; has developed CIDOC-CRM based ontologies for the expression of PDI; made scenario definitions and implementations and has evaluated the current state of repository.
From the IRCAM point of view, the validation is triple and consists of: firstly, the ability to build APs and RepInfo in order to do good migration; secondly, the fact that CASPAR and the OAIS model have proved that they are compliant with the existing best practices that can be shared in the community; finally, the efficiency, which is demonstrated by the capacity to provide to the user and the archivist with information about the current state of the repository, e.g. information concerning the RepInfo or PDI missing.

slide 38 IRCAM has planned future developments. It is involved in two research projects that are funded by National Research Agency in France (until the end of 2012).
The first one is focused on RepInfo, as IRCAM is not completely satisfied with the RepInfo which is currently extracted from the Max/MSP, since the migration is not completely automated and a great deal of manual adjustments have to be done; moreover not all the RepInfo proper data has been defined and IRCAM is not always able to provide all the RepInfo for all the details in the patch.
The second project is focused on PDI, by tracking provenance, and it will be pursued with INA and UTC that are already CASPAR partners, and also with EMI music.
Furthermore, IRCAM will continue to develop the MustiCASPAR server in the two research projects: ASTREE and GAMELAN. The first one pays attention to the community of Max/MSP developers and GAMELAN is focused on the audio production.
Finally, IRCAM have to integrate MustiCASPAR with the current repository and produce adequate RepInfo and the adequate missing PDI.