2. Technical Point of View

TECHNICAL CHALLENGES PROVIDING TOOLS FOR *THE ETHNOARC* PROJECT

CHRISTIAN FUHRHOP, RAJU VAIDYA

ABSTRACT

The authors of the paper implemented a set of tools to allow researchers to check out ethnomusicological databases and collate their search results. Providing these tools presented a number of technical challenges, some specific to the project, some being more generally applicable. The paper documents these challenges and the methods used to address them.

INTRODUCTION

During the course of the ethnoArc project (September 2006 – August 2008) we implemented a number of tools to allow researchers to query multiple ethnomusicological databases and collate their results, even though the underlying databases might be structured differently.

Information about the project can be found at www.ethnoArc.org and the software implemented is available at developer.berlios.de/projects/ethnoarc as OpenSource.

Fraunhofer FOKUS had been active in the area of content management systems (CMS) in the area of audio-visual content for a number of years before getting involved in the *ethnoArc* project. While this gave us the experience to deal with many of the technical aspects of the project, the audio and visual data we had previously encountered were all associated with commercial recordings (CD, MP3, DVD) and used a small number of standardised or common metadata formats. Compared to this, ethnomusicological content we had to deal in *ethnoArc* was much more complex. Moreover, *ethnoArc* should support multiple content providers with their own structure and language.

While developing the software, a couple of decisions had to be made that influenced technical aspects, based on the specific requirement of ethnomusicological archives.

REF/JEF, *1–2*, p. 11–20, Bucureşti, 2009

In the following sections, we will describe the various issues that were specific to the problem of providing access to multiple ethnomusicological archives in different countries, using different languages and structures. Following the description of the situation at the archives, we describe how these situations influenced the way the *ethnoArc* software was developed and implemented.

COMPLEXITY

While commercial recordings can be described with a limited set of fixed metadata, ethnomusicological content is generally rich in metadata. Recordings are not only accompanied by written annotations, but also by extensive field notes, photographs, other representations (MIDI or musical notations), their relation to travels and other recordings, and similar. For commercial recordings, metadata can usually be arranged in two "flat" sets of metadata, one for the recording itself and one for the data carrier (record or CD), while ethnomusicological data often consists of many small metadata groups with complex relations.

To be able to deal with this metadata, we decided to allow a rather freeform definition of the database structure to be used in *ethnoArc*. While most database structure builders are table oriented, since the resulting structures can easily be matched to SQL tables, this approach tends to encourage users to create a small number of hierarchical tables with a large number of metadata elements per table, regardless of their suitability for the archive and potentially losing metadata relation details.

Our approach was to use a simple element/relation model that could be used to model detailed structures within the metadata as closely as possible, without explicitly or implicitly prompting users to group metadata elements in "flat" table structures.

References with an archive are not always in a strict hierarchy, but may include of cross-references and circular references. Rather than implementing a tree structure and handling such references as special cases, we explicitly allowed our relations to be non-hierarchical and even circular, allowing archives to mirror their existing structures as close as possible.

ACCEPTANCE

An obstacle to acceptance of new database software and structures is the learning effort to get accustomed to a new metadata format and the effort to convert existing metadata into the new format. To reduce that effort, we avoided to force the archives to use a common metadata format, but provided them with the tools to build a metadata structure that mirrored their existing structure as closely a

possibly, thus allowing them to work with structures that they are already familiar with and keeping the conversion effort from existing data minimal.

DIVERSITY

The structures used in ethnological collections often still reflect the interests of the original collector or collectors. While for other types of audio collections, such as broadcasting archives, a common metadata standard can usually be defined (sometimes with a few private extensions to handle unusual metadata types), most ethnomusicological archives only share a few common metadata fields. While a large amount of the underlying information is similar between the archives, their specific representation and their relation to other metadata fields within the archive differs significantly between sites.

Encouraging different archives to a use a common metadata standard is a desirable goal, and recommended for newly acquired content and metadata, but is in many cases not practically attainable for legacy content and metadata. Information may not be available anymore, supplementary information is difficult to fit into pre-defined fields and other fields might require essentially trivial, but time consuming format changes. Additionally, if not used carefully, format changes can create data artefacts, which can be difficult to detect later.

Examples for this are date references. For some recordings only the year (or even an approximate year) is known, while other recordings have detailed information. Thus, many archives have a free-form date field, which is used to copy the handwritten comment on the original info sheet from a field trip. If forced to conform to a fixed date format, information might get lost or invalid information might be added. While not specifically archive related, the use of the MP3 ID3 tag provides a cautionary tale: In the first version of ID3, only recording years could be stored. In the second version, a recording day and month could be added as well. While these were not mandatory fields, a large number of MP3 tools added a day/month tag (TDAT) when creating a version 2 ID3 tag. Since no information about the day was available in the original ID3 tag, the majority of MP3 files now seem to have been recorded on the 1st of January.

While this is often quite irrelevant for commercial audio recordings, the specific date of a recording is important ethnomusicological research topics, especially if they are related to the study of festivals and seasons of the year. Knowing that information is incomplete or even faulty (such as having a comment “I can”t read whether this is a 2 or a 5.”) is more useful than information required to fit a specific format, which may be inappropriate in some cases to represent the original information.

To help archives to cope with such metadata, we avoided strong typing of the metadata fields and allowed text based metadata in all element fields. While this leads to slightly reduced search efficiency, the capability of entering data from paper sources "as is" instead of "as should be" helps preserving the original information and allows commenting.

Additionally, to allow the preservation of the diversity between archives, no fixed pre-defined common or standardized structures have been provided (although archives are encouraged to harmonize their metadata structures and exchange wordlists and thesauri where appropriate).

LANGUAGE

Almost all information is only available in the local language of the archive site. While it would certainly be desirable to have the information available either in either the native language of the researcher or some widely used language (which typically defaults to English for most European applications), this is in most cases impractical. Automatic translation of metadata, especially if they concern lyrics or descriptive texts) is not of sufficient quality to be useful for researchers, so an automatic cross-translation between the archive language and the researcher language is not feasible and while a professional (human) translation of all metadata to English would clearly be beneficial, the cost for this is high and, except for special cases, the effort is not made by archives. This is compounded by the fact that such translation effort provides little additional value for the archives itself. For local users at the archive, the original language is sufficient, so spending money on foreign translations provides no advantages at the archives themselves or for local users.

Approaches have been made, most recently in the MultiMatch project, to use automatic translation tools at least for search phrases. This allows researchers at least to determine whether archives have any metadata related to a specific search phrase, without the need to have a full text translation of either the query or the archive metadata.

In *ethnoArc*, we decided not to use automatic translation tools, partly because this would have used up a significant amount of resources in an area not central to the project and partly because of the work performed in MultiMatch.

To improve access and usability for researchers that are not fluent in the language used by the archive site we required archive to provide a description of the metadata fields in English (not the metadata itself, just the description of the field, such as "This field contains the birth date of the performer of a musical piece."). This helps the researcher in understanding the structure used in an archive and determining whether metadata fields, which relate to the research theme, exist

at the archive at all. Providing such information does not require a large amount of effort at the archive site, since the description only needs to be provided once on definition of the database structure and there is no need to provide translations for individual metadata items.

We also provided specific relations ("AlternativeLanguage") in our database to allow archives to specify metadata in other languages. While this is typically not used for metadata elements that allow free text entry, due to the cost and effort mentioned earlier, it allows archives to provide translations for elements that provide only a limited number of choices, such as type of data carrier, gender of persons or lists of instruments. Similar to the use of English descriptions, the effort of providing English (or other language) alternatives for such lists has to be made only once when setting up the database, so the accessibility for foreign researchers comes with comparatively little cost to the archives.

ACCESSIBILITY

Understanding the metadata structure of an archive can be a daunting task. This is especially true for researchers that are external to the archive. Metadata fields have often cryptic names, sometimes based on a language foreign to the researcher, relations between different tables are only implicit (for example, "PersID" in table "Objects" refers to "ID" in table "Persons") and written documentation of the structure is unavailable.

Since providing access to external researchers was a main goal in the *ethnoArc* project, we addressed this problem in a couple of ways.

The names of elements are not restricted in length or character set, so archives are encouraged to provide meaningful names ("Last name, first name") instead of formats determined by the restrictions on database fields names, which may make the meaning of the field harder to determine (such as naming the field "LaNaFiNa" instead).

Archives are required to provide descriptions of all metadata fields in English, so that researchers not only have the field name as an indication of the meaning of a field, but also a textual description, which, hopefully, includes information about the purpose of the field as well as annotations and remarks about its use.

Relations between elements are explicitly defined in the database description and not just implicitly given by similar names in SQL table elements. This not only allows a researcher to follow the relations between metadata elements, but also allows software tools to determine, use and display the relations. Based on that, the ethnoArc search engine can provide an interactive graphical representation of the metadata structure used by an archive, allowing the researcher to navigate the structure, zoom into areas of special interest, observe the relation between metadata fields and read the additional metadata description for additional help.

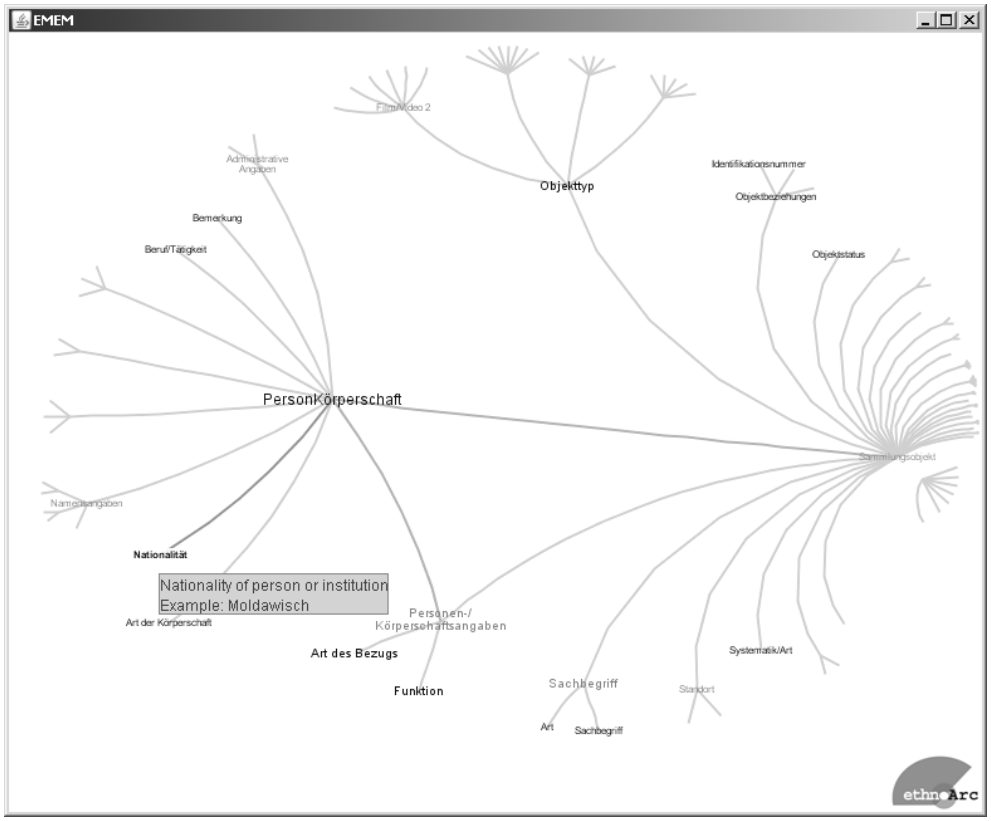

Fig. 1 – Example of metadata structure display.

In addition to displaying the description of a metadata element, an example value is taken from the actual database and presented to the researcher to provide additional information about the purpose of the database field.

IT KNOWLEDGE

Since smaller archives usually do not have a specialised IT department and the computers are often maintained by one of the archivists as a secondary task, we tried to keep the required IT knowledge at a minimum.

To achieve that, we provided as much information and documentation as possible, kept the number of components to be installed small, relied on few standard pre-requisites (mainly Java and mySQL) and designed the system architecture to encourage secure handling of the software.

As an example, for the last point we provided a query server that handles queries from outside the local network of the archive, thereby avoiding the need to

expose outside access to the SQL server. While technically the SQL server can be sufficiently secured to allow remote access directly to the server without compromising site security, this requires a fair amount of IT knowledge, which may not be available at an archive site. Just firewalling the SQL port and installing the ethnoArc query server is easier to achieve at such sites.

While basic IT knowledge for installing and starting software can be assumed for the archives, knowledge of structuring databases cannot be expected. Additionally, the archivists knowing most about the metadata elements and their relations are not necessarily the same people that maintain the computers, software and network.

As a result of this, care has been to keep the metadata structure description as flexible as possible, avoiding restrictions based on technical concerns (such as tree structures), and by having no elements solely needed due to the internal handling of the database (such as index elements or the specification of specific key or index elements) as part of the database specification, allowing the designer of the metadata structure to concentrate of mapping the existing structure as closely as possible, without having to pay attention to additional requirements.

COST

A common problem of all the archives involved was a lack of funding for installing and operating the ethnoArc tools, especially for software coming from a research project and not a commercial provider. While software installations, even test or trial installations, in industrial environments can cost tens of thousands of Euros without causing budgetary problems, to be considered in archives, software must be available at low acquirement and operation cost.

Providing the software resulting from the *ethnoArc* project as OpenSource (and thus free to use) was already part of the original *ethnoArc* project contract and thus given. Put costs are not only created by the software itself, but also from resulting software and hardware requirements.

To keep costs at a minimum, we used only freely available software as a base for our developments. This included mySQL as the database and free software libraries for graph display (hyperapplet), XML parsing (xerces) and the handling of Excel data (jxl). Since all our implementations were based on Java and the database engine is available for all important platforms (Unix, Mac, Windows), *ethnoArc* is essentially platform-agnostic, allowing archives to use their available IT environment. The database is also sufficiently lightweight and efficient, so it can run on any reasonably modern computer as a background process, without impacting resources on that computer significantly, thus allowing the use of any archive PC as the database server and not requiring the purchase of additional hardware.

Another result of keeping the cost low was the implementation of some database tools from scratch. This applies mainly to the data entry tool. While there are numerous libraries for creating entry forms and masks for databases on the market, no suitable tool was available as OpenSource or as a free library. Rather than requiring archives to purchase a runtime licence for such a library, the *ethnoArc* data entry was written without using such a library.

INITIAL SET-UP EFFORT

While operating costs were one of the concerns at the archive sites, set-up costs were another. Even with essentially free software, installing the software, filling the database and evaluating the usability of the software takes up resources and manpower.

While the installation effort can be kept low by using easily installed tools as well as tutorials and documentations and should not take more than a working day, archives need to specify their metadata structure, which can take a fair amount of effort. Evaluating the tools usually also required the conversion of existing data to the *ethnoArc* structure, which requires programming effort, which can often not be provided by the archive itself, usually requiring the hiring of a programmer for about a month.

There is little that can be done about this, since these tasks are archive specific and need to be solved individually for each archive. The only support that can be given is by example. Using the *ethnoArc* tools, a metadata structure description can be from any of the already available archives, which can serve as an example for other archives. Software for importing data from existing database into the *ethnoArc* database is available in source form for the archives involved in the project. While this software cannot be used directly by other new archives, it gives programmers an example on which software for the new archive can be based, avoiding the need to start from scratch and only with the API documentation.

CONTROL

A significant requirement of all archives was the need of keeping control over their own metadata. Any solution that required submitting metadata to a central server was likely to cause acceptance problems. By providing a concept that allowed archives to maintain their databases locally, without a central server, while still allowing researchers to perform queries on multiple archives, we attempted to reduce concerns at the archive sites, while improving researcher”s access to archive metadata.

LONGEVITY

A common problem in projects with a limited time frame (such as the two-year *ethnoArc* project) is the use of the project results after the end of the project. While effort can be spent during the course of the project to create interest in the tools and results, finding funding to promote and maintain the results after the end of the project can be difficult.

The main obstacles to continuous use of project results are running costs, maintenance and single points of failure. After the initial set-up, which is the largest cost for an archive, but occurs only once, the cost of running the server is quite low (see "Costs") above. By releasing the source code on an OpenSource platform that is not connected to the project (BerliOS), the code can be distributed and maintained, even if the original developers should not be available.

Since there is no central server required by the project, archives and researches can continue using the *ethnoArc* tools without concerns about funding a coordinating agency or service provider and without having to rely on specific set of other archives. If, for example, five new archives would adopt the *ethnoArc* tools and two of the original archives would stop using the tools, the usefulness of the tools would not be significantly diminished (except for the incapability to access those two archives). There is no dependency on the continued availability at the original archives (although, of course, the continued use there would be desirable).

It is also relevant that there is no critical mass of archives required to make the tools useful. If software only provides benefits if a large percentage of the targeted institutions make use of the software, that percentage needs to be reached during the course of the project, since the tools will otherwise fall into disuse if they are no longer actively promoted. The *ethnoArc* tools, however, already provide benefits if only one or two archives use them, with increasing benefits due to any additional archive, so the decision of an archive to adopt the software is not depending on the decision of other archives.

COMMUNITY

An issue that has been insufficiently handled in the *ethnoArc* project was the support of user and archive communities. While the issue was discussed early in the project, there was no significant effort spent to support communities of tools users. A related concern was that the support of a user community might introduce a single "contact point" and thus a special node in the distributed system architecture, which was conceptually problematic and might have lead to a "single point of failure" impacting longevity (see section above).

While none of these have been implemented, in hindsight the following should have been provided by the project: a server to store and exchange queries,

allowing researchers to make queries available to other researchers, to document, comment and rate such queries and improve upon existing queries (for example by extending an existing query to include an additional archive). Additionally, some method allowing researchers to annotate the metadata available at the archives would be a useful extension. While providing annotations to a single archive is a reasonably simple extension, a more useful tool would allow researchers to highlight relations between elements from multiple archives and comment on them. However, this function would take significant effort to implement and be difficult to achieve without some sort of central server.

SUMMARY

While the solutions chosen in the *ethnoArc* project are probably not applicable to other projects, the underlying problems that led to these specific approaches and solutions are likely to be similar for other software project in the area of ethnomusicological research and aimed at ethnomusicological archives and should be considered as such when developing solutions for that environment