A Concept for Using Combined Multimodal Queries in Digital Music Libraries*

David Damm[1], Frank Kurth[2], Christian Fremerey[1], and Michael Clausen[1]

[1] University of Bonn, Department of Computer Science III
Römerstraße 164, 53117 Bonn, Germany
{damm,fremerey,clausen}@iai.uni-bonn.de
[2] Research Establishment for Applied Science (FGAN), FKIE-KOM
Neuenahrer Straße 20, 53343 Wachtberg, Germany
kurth@fgan.de

Abstract. In this paper, we propose a concept for using combined multimodal queries in the context of digital music libraries. Whereas usual mechanisms for content-based music retrieval only consider a single query mode, such as query-by-humming, full-text lyrics-search or query-by-example using short audio snippets, our proposed concept allows to combine those different modalities into one integrated query. Our particular contributions consist of concepts for query formulation, combined content-based retrieval and presentation of a suitably ranked result list. The proposed concepts have been realized within the context of the PROBADO Music Repository and allow for music retrieval based on combining full-text lyrics search and score-based query-by-example search.

1 Introduction

Increasing amounts of digitized musical content result in the need for managing them automatically. Especially for libraries holding a vast amount of musical content that is steadily increasing due to ongoing digitization there is a high demand on automatisms to cope with the large number of documents. At the same time, technology from the field of Music Information Retrieval (MIR) has been developed for content-based querying of music collections using different modalities, e.g. by entering text or score-fragments, by cropping audio snippets from audio recordings, by whistling, singing or humming melodies, or by tapping a rhythm. Several of those query types have been realized for existing music collections [1,2]. In this paper, we consider the music repository set up within the PROBADO Digital Library Initiative [3].

The retrieval functionalities commonly provided by existing query engines, consisting of the steps of query formulation, query processing, visualization of query results and user interaction are, in a sense, *unimodal.* That is, there is no

---

general concept on how to integrate and jointly process multiple query types in this processing chain, hence affording *multimodal* retrieval. Exceptions to this are some straight-forward approaches which process different query modalities separately and then use unions or intersections of the individual query results [4,5,6]. Especially in the case that a user has only sparse or fuzzy knowledge regarding the different modalities, the combination of different, multimodal queries might compensate this issue and eventually result in improved search results. Think for example of a user remembering a rough harmonic progression as well as some of the lyrics of a piece of music he listened to on the radio.

As currently available search engines are limited to unimodal queries, queries may be formulated very quickly, which is very convenient for the user. On the other hand, those unimodal queries frequently result in a lot of matches in case the query is not specific enough. Consequently, this kind of searching is likely to end up in the time-consuming process of reformulating the query in order to constrain the high number of matches. To overcome the limitations of unimodal retrieval, this paper presents a retrieval concept for realizing combined, multimodal querying in the context of music libraries. Particularly, we consider the task of integrating full-text (lyrics-) queries and audio retrieval using the query-by-example paradigm. We present concepts for the full processing chain of (a) formulating multimodal queries within a suitable user interface, (b) content-based retrieval including an adequate ranking strategy and subsequent generation of a result list, (c) suitably visualizing the list of query results providing a user-friendly and content-based overview representation, and (d) enabling possible user-feedback and interaction. The approach has been motivated in the real-life context of setting up the PROBADO Music Repository [3]. The special case of combining the textual and acoustic modalities serves as a starting point for integrating several different query modalities that will be of interest within the PROBADO Music Repository.

The rest of this paper is organized as follows. The subsequent Sect. 2 contains an overview on the current status of the Digital Music Repository within the PROBADO Digital Library Initiative. Sect. 3 contains our proposed concept of combining multimodal queries and presents a retrieval mechanism for the example of combined full-text- and audio retrieval. The paper concludes in Sect. 4 with prospects on future work.

2   The PROBADO Music Repository

The PROBADO Digital Library Initiative is a research effort to develop next generation Digital Library support for non-textual documents. It aims to contribute to all parts of the digital library workflow from content acquisition to semi-automatic indexing, search, and presentation of results. Currently, two different repositories are set up, PROBADO-3D containing 3D objects from architecture, and the PROBADO Music Repository considered in this paper.

Sections 2.1 and 2.2 contain an overview on the PROBADO Music Repository which is currently set up at the Bavarian State Library (BSB), Munich. While 2.1
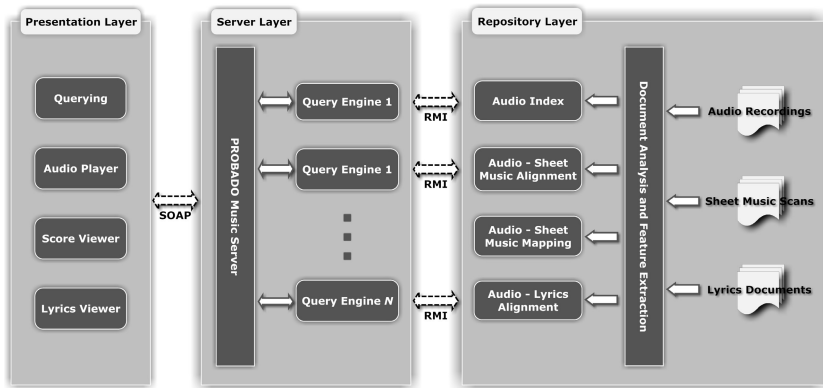
Fig. 1. Overview of the structure of the PROBADO Music Repository

contains a general system overview, 2.2 puts a focus on the generic functionalities for content-based retrieval.

2.1   System Overview

The PROBADO Music Repository incorporates the storage of and the access to digital music documents. For preservation purposes, digital copies of available musical content held by the BSB such as audio recordings, sheet music and other music-related material are made. For indexing purposes, these digital copies are analyzed and annotated by recent state-of-the-art MIR techniques.

One key task is to build up content-based search indexes in order to search for, e.g., lyrics phrases or score fragments. Another key task is to consolidate all available documents (particularly audio recordings and scanned sheet music) for the same piece of music and relate them among each other. Mapping- and synchronization-techniques are used to create alignments between meaningful entities within (a) sheet music pages and time segments within audio recordings (score-audio-synchronization), (b) words and time segments of audio recordings (text-audio-synchronization), and (c) time segments of different audio recordings of the same piece of music. A more detailed view on the topic of extracting meaningful entities from scanned sheet music and its mapping to audio recordings is given in [7,8,9]. The resulting benefits are content-based and cross-modal searching capabilities, synchronous multimodal playback and visualization of pieces of music, and advanced cross-modal browsing capabilities.

All these functionalities are realized and implemented within a modular system as depicted in Fig. 1. The system is organized as classical 3-tier architecture and consists of the *presentation layer*, the *server layer* and the *repository layer* (left to right). Search indexes, annotations and linking structures between different modalities are obtained in a preprocessing step which is carried out offline in the repository layer. The access to index structures and synchronization data as

well as the delivery of musical content to the user takes place in the server layer. The presentation layer consists of user interface components for accessing musical content, especially content-based searching for musical content, navigation and browsing within search results, as well as synchronized playback of audio and sheet music or lyrics. For each system interaction such as retrieving search results and accessing musical content, there is a dedicated module, referred to as *Query Engine*. The communication between the presentation and the server layer is provided by a service-oriented architecture (SOA) [10].

2.2   Content-Based Querying

This subsection gives some more detail on the query functionality provided within the PROBADO Music Repository. As mentioned before, a key task in the context of the PROBADO Music Repository is to enable content-based search using lyrics phrases or score fragments as queries. Due to the consolidation of all musical content belonging to the same piece of music, each content-based search may also be viewed as cross-modal. This is, one can use either of the visual or textual modalities as queries, while aiming to find matches in the other modality. Up to now, two distinct options for content-based querying are available, *lyrics-based retrieval* as proposed in [11] and *score-based retrieval* using audio matching as proposed in [12]. Both approaches use indexing techniques to achieve a high retrieval efficiency.

The lyrics-based retrieval allows for formulating a query in the textual modality by entering a few words in order to find the positions within audio recordings where the words are sung. The mapping of positions within the lyrics text document to time segments within an audio recording is performed using lyrics-enriched MIDI files as described in [11]. Here, onset times of individual words or syllables are explicitly given within a musical context. This information, in turn, is then used to synchronize the lyrics to the audio recording. The subsequently used indexing technique is based on inverted files which are well known from classical full-text retrieval [13] and enhanced for the special case of a lyrics search. The search is fault-tolerant w.r.t. misspelled or omitted words in both the query as well as the lyrics, see [11].

The score-based retrieval follows the query-by-example paradigm. A query is formulated in the visual modality by selecting a portion of a sheet music page, particularly a few consecutive measures. The system retrieves all occurrences of the selected music excerpt within the indexed audio recordings. Note that the sheets of music are images obtained from scanned analogue pages and thus the actual musical content or semantic is expressed in the visual modality. For the purpose of synchronization with audio recordings, the images have to be transformed to a symbolical representation. This is done by optical music recognition (OMR) techniques [14,15]. Although the output of the OMR processing is quite error-prone it is sufficient for synchronization with the audio recordings. Exploiting this synchronization, instead of querying the selected score excerpt, the according snippet of an associated (synchronized) audio recording is used for the search process. Here, a sequence of audio features is extracted from the

snippet and subsequently a feature-based search on an audio features index is performed. Due to the extraction of consecutive features that reflect the chromatic harmonic progression of the underlying audio snippet at a coarse level, the audio retrieval system is robust against changes in timbre, instrumentation, loudness and transposition and therefore musically similar snippets can be found regardless of a particular performance [7,8]. For a more detailed view, we refer to [16] and the references therein.

3   Combining Multimodal Queries

In this section, we propose how to incorporate a concept for combined multimodal queries into the typical stages of a query-retrieval chain, particularly query formulation (Sect. 3.1), content-based retrieval and ranking (Sect. 3.2), presentation of query results (Subsect. 3.3) and mechanisms for user feedback and navigation (Sect. 3.4).

One of our key contributions is that our system enables the user to formulate a query that may consist of different modalities, including the textual, visual and auditory modality. In particular, he can query a combination of metadata, lyrics and audio fragments. For this, he formulates single, unimodal queries and adds them successively to his search. Single queries are gathered in a special structure

Fig. 2. User interface for query formulation tabs (top left), Query Bag (top right), display of the results (bottom left) and the document viewer (bottom right)

for representing sets of queries, referred to as *Query Bag*. After the Query Bag is submitted to the retrieval system, the user is presented a list of pieces of music that match in at least one modality regarding his query. To organize the result list we introduce a new ranking approach based on a combination of multiple result lists ensuring that pieces of music containing more matching modalities are given a higher rank, see 3.2. The results, i.e. pieces of music, may be viewed in detail and played back in the document viewer module. Additionally, both the result list and the player can be used for querying, query refinement and document navigation, see 3.4.

To give a user-friendly and intuitively operable interaction environment, our approach was to incorporate the look-and-feel of well-tried Internet search platforms that a wide range of users is familiar with. The user interface for the retrieval, browsing, playback and navigation in musical content is completely web-based and runs in every state-of-the-art Internet browser. Fig. 2 shows a snapshot of a typical system configuration. Similar to popular existing query engines, the top part of the GUI contains the query formulation area while the result view area is located below. Both areas are further subdivided to facilitate the subsequently described functionalities. The query formulation area is split into both a tab cards region and the Query Bag region. The result area is divided into the result list pane and the document viewer.

In the following, a further in-detail look at each particular stage of the query-retrieval chain is given. Throughout the whole section, the piece of music "Gefrorne Tränen" belonging to the song cycle "Winterreise" by Franz Schubert will serve as our running example.

3.1 Query Formulation and Interface

In this subsection, we give a detailed look at the query formulation and its interface. The query formulation area, shown in Fig. 2 (top), consists of various query formulation forms per modality which are organized as tab cards (top left). It further contains the Query Bag (top right), where single queries can be added to, viewed, revised or removed. Currently, the user is enabled to formulate metadata, lyrics and audio fragment queries within the appropriate, designated tab card. Additional query formulation types such as entering a melody by a virtual piano, humming a melody or tapping a rhythm are planned to be integrated in the future.

From within any tab card the user has the choice to either perform an immediate, unimodal search using the just formulated query (classical query) or to add the latter to the Query Bag and continue with the formulation of another query in order to gather a couple of unimodal queries. The Query Bag stores all queries and offers an overview representation of all gathered queries. So, the user at any time is informed about which queries he has collected so far. Each single query inside the Query Bag can be examined more precisely by clicking the plus-sign icon left to the query. To the right of each query there are icons for reformulating the query and for removing it from the Query Bag as well, by clicking either the pencil- or the "x" icon, respectively. By clicking

the pencil icon, the corresponding query formulation tab opens for editing. Once the user has finished assembling the individual queries, the search button at the bottom of the Query Bag can be clicked in order to submit them to the search engine as one integrated, multimodal query. Subsequently, a multimodal search is performed.

3.2   Content-Based Retrieval and Ranking

Once the Query Bag is submitted, the system disassembles it and delegates each contained single query to an appropriate query engine which is capable of handling the particular type of query. The query engines act independently from each other and for each modality a homogeneous list of matches is returned. In this, each match consists of a document ID, the position of the matching segment, and a ranking value $r \in [0, 1]$. In case of content-based queries, the latter segments are generally short parts of the document. If, however, a document matches due to its metadata description, the document is said to match at every position; i.e., a matching segment ranges from the beginning to the end of the document.

Due to the synchronization of different document types such as audio recordings, sheets of music and lyrics documents, all matching segment boundaries can be expressed in the time domain, i.e. translated to a start- and an end-timestamp. Thus, all segments are directly comparable, which will be exploited in the subsequent combined ranking and merging. For merging and ranking of multiple result lists returned by the different query engines into a single, integrated result list we use a straight-forward bottom-up approach explained in the following.

Each result list returned by a query engine consists of document IDs and for each document ID there exists a list of matching segments. These segment lists are inserted into a hashtable, where a single data entry stores a piece of music's ID in together with related segment lists. For each inserted segment list, the respective modality is stored as well. With this, all inserted segment lists associated to the same piece of music are clustered and stored within a single hashtable data entry. Subsequently, for each entry of the hashtable a merging of the contained segment lists is performed. This step is now described in detail. Let M be the global number of queried modalities and m the local number of non-empty segment lists stored in a currently considered hashtable entry. We now consider the merging step of two segment lists, $L^1 := \{(b_1^1, e_1^1, r_1^1), \ldots, (b_{|L^1|}^1, e_{|L^1|}^1, r_{|L^1|}^1)\}$ and $L^2 := \{(b_1^2, e_1^2, r_1^2), \ldots, (b_{|L^2|}^2, e_{|L^2|}^2, r_{|L^2|}^2)\}$, where the i-th entry of list k is a segment $s_i^k = (b_i^k, e_i^k, r_i^k)$ consisting of a beginning- and an ending timestamp as well as a ranking value. In this, we assume that each segment list corresponding to a single modality does only contain non-overlapping segments. Let L be the merged, integrated segment list. For merging two lists L^1 and L^2 into L, we consider two cases. If a segment s_i^k does not overlap in time with any segment s_j^l of the other list, s_i^k is simply copied to L. If there is a temporal overlapping of a segment $s_i^k := (b_i^k, e_i^k, r_i^k) \in L^k$ and a segment $s_j^l := (b_j^l, e_j^l, r_j^l) \in L^l$, s_i^k and s_j^l are merged into a new segment $s := (\min(b_i^k, b_j^l), \max(e_i^k, e_j^l), r)$ which is inserted
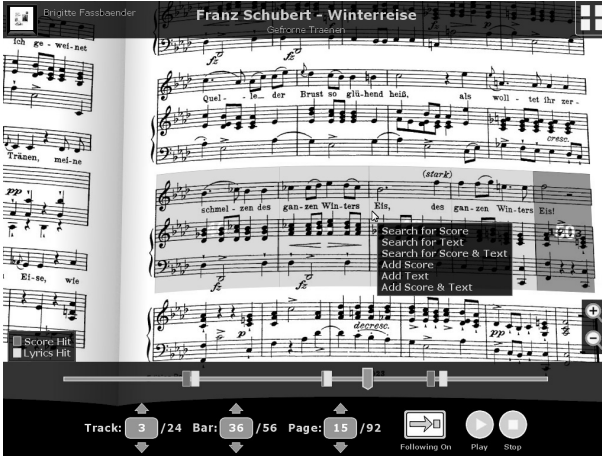
Fig. 3. Document viewer in the score visualization mode. Multimodal content of selected measures can be queried.

into L. Note that overlaps do reflect simultaneously arising hits and for this reason, we want them to get higher ranked in general. To additionally promote small individual ranking values r_i^k, r_j^l in the latter case of segmental overlap, the assigned ranking value is defined as $r := (r_i^k + r_j^l) \cdot f_{boost}$, where $1 \leq f_{boost} \leq M$ is a constant global boosting factor. The merging of the m segment lists is done iteratively until no residual segment list remains. Note that the factor f_{boost} is applied only once during the processing of the segment lists. When all m segment lists are merged into a single, integrated segment list, all of the segments' ranking values are normalized by applying the factor $1/(M \cdot f_{boost})$ resulting in a final ranking value in the interval $[0, 1]$. This algorithm can be implemented in a straight-forward manner with a time complexity linear in list lengths, as long as each list L^k is sorted ascending w.r.t. the beginning timestamps b_i^k of its matching segments $s_i^k := (b_i^k, e_i^k, r_i^k)$.

In the end, for every piece of music there results an individual, integrated list of multimodal matching segments along with assigned ranking values. The overall ranking value for a piece of music is determined by the maximum ranking value of its integrated segment list. Finally, the pieces of music are put into a new result list and sorted in descending order of their respective ranking values. This means that the final result list is organized such that the more modalities within pieces of music do match, the higher their assigned ranking values are. Therefore they occur at earlier positions in the list. In turn, pieces of music matching in less modalities occur at later positions in the list.

3.3 Integrated Presentation of Query Results

Typically, available search engines provide the user with a flatly organized result list only where the list entries commonly consist of single documents. However,

in the case of the music domain, there are multiple document types (in our case audio recordings, sheet music and lyrics documents) representing a piece of music using different modalities. As in our applications we have multiple documents of the different types available for a piece of music, we believe that it is of special interest to present all those documents in a collective manner, even if some of them do not match a user's query. Therefore, we took this consideration into account concerning the presentation of query results.

The bottom area of Fig. 2 shows the result list (left) and the document viewer (right). While the result list shows the matching pieces of music regarding the query, the document viewer offers access to all content belonging to the currently selected piece of music. It furthermore gives a more detailed view on matching regions within its multimodal content and is also responsible for playing back the latter. As mentioned before, the resulting matches are presented to the user not at document level. Instead the user is offered every piece of music where at least one document representing that piece contains one or more matches to the current query. All documents belonging to the same piece of music that match the user's query are summarized within a single list entry. The entry shows the artist's name and the title of the piece of music, a lyrics excerpt as well as the matching documents along with their number (in brackets). Additionally, at the bottom there are links to show more titles of the artist and to save the result (see also Sect. 3.4). A more detailed view of the single matching documents as well as the exact matching positions therein, is given in the document viewer.

One key feature of the document viewer is the integrated display of matching segments along the timeline bar at the bottom. Besides the adjustment of the current playback position by using the slider knob, it is used to show all matching positions within the currently selected multimodal contents used for playback. The matching positions are indicated by colored boxes along the timeline bar, where the color and brightness of the boxes encode modality and ranking value, respectively. Additionally, matching segments within "inactive" documents, i.e. others than those ones used for playback, are displayed as gray boxes.

Below the timeline bar there are further buttons to control the playback state (start/pause, stop) as well as the playback position. While the control buttons retain their positions, the labels are exchanged depending on the currently selected visualization type, i.e. scores or lyrics, in the center.

Another key feature of the document viewer is to simultaneously playback multimodal content associated to the currently selected piece of music. Here, the content might be audio recordings, sheet music and lyrics as well, where the last two ones are displayed in the center area. From within a pop-up menu that is reachable by clicking the album cover in the top left corner, the user can choose which contents are used for playback. Here, all available content associated to the piece of music are shown.

While an audio recording is played back, available sheet music or lyrics are also displayed synchronously; i.e., the user can visually track the currently played measure or the currently sung words, respectively (cf. Fig. 3 and 4). Due to this style of enjoying music in a multimodal way, the document viewer may be

Fig. 4. Document viewer in the lyrics visualization mode. Selected text can be queried.

thought of as being a video player, but in addition it provides sophisticated user interaction options such as navigation and query refinement, which are examined in Sect. 3.4.

3.4   Query Refinement and Navigation

From within the result list, for each retrieved piece of music, the user is enabled to request more titles of the same artist by choosing the appropriate "get more titles from artist..."-link which is available from the context menu. Once the user selects this option, the Query Bag is flushed, rebuilt with only a simple metadata query consisting of the artist's name and a subsequent new search is performed, what finally results in an updated list that displays all pieces of music by this artist contained in the database.

From within the document viewer, the user can select which audio-visual content is used for playback in case that more than one document per modality is available. For example, if different audio recordings of a piece of music are available, the user has the choice to decide which specific performance he wants to listen to. With this functionality, one is also allowed to switch between different performances while retaining the actual musical playback position. Thus, the user can additionally draw local comparisons between different interpretations of a piece of music. Sheet music books may also be exchanged, if more than one is available.

Furthermore, the user may thumb easily through the sheet music book or the lyrics text, by clicking on a measure or a word, respectively, whereupon the playback position is changed accordingly.

Moreover, the user can utilize content-based searching capabilities from within visual content following the query-by-example paradigm. When the user selects a portion of either a sheet music page or the lyrics text, he can use this excerpt for

a new query, see Fig. 3 and 4. He has the option to start either a completely new search based only on the selected portion, or to add the query as an additional partial query to the Query Bag. In the case of sheet music, a portion may consist of two modalities, score and text (cf. Fig. 3). Here, the user can choose, whether he queries both modalities together or separate from each other.

As matching segments within multimodal contents are displayed as boxes along the timeline bar at the bottom of the document viewer, they can be simultaneously utilized for navigation purposes. By clicking on a box, the playback is started or resumed at the corresponding time position. This functionality enables the user to jump directly to the found segments matching the user's query.

4   Conclusions and Future Work

In this paper, we presented a concept for integrated multimodal access to digital music libraries. Especially, we considered the integration of multimodality into all stages of the query-retrieval chain, aiming at facilitating web-based user access to pieces of music by means of the various available kinds of music documents.

The concept of combining multiple queries has not to be restricted to using different modalities as used in this paper. Naturally, multiple queries of the same modality (like multiple score-based queries in one Query Bag) might be used. Using the Query Bag approach, even intentionally different queries might be combined into a single query resulting in an integrated result list, readily available for browsing.

The future work on the topic of this paper as well as in the context of the PROBADO Music Repository, on the one hand consists of improving the retrieval performance of the query engines described in Sect. 2.2. On the other hand, we want to investigate the idea of a *direct* alignment of lyrics from within sheet music, recognized by OMR software, and "clear" text documents, containing the lyrics as well. This allows for a direct mapping between word positions within the text documents and image regions within the scanned sheet music, which, in turn, allows for a mapping to time segments due to the synchronization of sheet music and audio recordings. Thus, the need for an intermediate representation like MIDI as proposed in [17] is not applicable. In the context of the PROBADO Music library project, further studies will be conducted to evaluate and improve the system design and usability.

References

1. Musipedia: `http://www.musipedia.org/`
2. Typke, R.: Melodyhound: Search within the music (2006), `http://melodyhound.com/`
3. Kurth, F., Damm, D., Fremerey, C., Müller, M., Clausen, M.: A framework for managing multimodal digitized music collections. In: Christensen-Dalsgaard, B., Castelli, D., Ammitzbøll Jurik, B., Lippincott, J. (eds.) ECDL 2008. LNCS, vol. 5173, pp. 334–345. Springer, Heidelberg (2008)

4. de Kretser, O., Moffat, A., Shimmin, T., Zobel, J.: Methodologies for distributed information retrieval. In: Proceedings of the 18th International Conference on Distributed Computing Systems (ICDCS 1998), pp. 66–73 (1998)
5. Henrich, A., Robbert, G.: Combining multimedia retrieval and text retrieval to search structured documents in digital libraries. In: DELOS Workshop: Information Seeking, Searching and Querying in Digital Libraries (2000)
6. Kailing, K., peter Kriegel, H., Schönauer, S.: Content-based image retrieval using multiple representations (2008)
7. Hu, N., Dannenberg, R., Tzanetakis, G.: Polyphonic audio matching and alignment for music retrieval. In: Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (October 2003)
8. Bartsch, M.A., Wakefield, G.H.: Audio thumbnailing of popular music using chroma-based representations. IEEE Transactions on Multimedia 7(1), 96–104 (2005)
9. Müller, M., Fremerey, C., Kurth, F., Damm, D.: Mapping sheet music to audio recordings. In: Proceedings of the 9th International Conference on Music Information Retrieval, ISMIR 2008 (2008)
10. W3C: Web services, `http://www.w3.org/2002/ws/`
11. Müller, M., Kurth, F., Damm, D., Fremerey, C., Clausen, M.: Lyrics-based audio retrieval and multimodal navigation in music collections. In: Kovács, L., Fuhr, N., Meghini, C. (eds.) ECDL 2007. LNCS, vol. 4675, pp. 112–123. Springer, Heidelberg (2007)
12. Kurth, F., Müller, M.: Efficient Index-based Audio Matching. IEEE Transactions on Audio, Speech and Language Processing 16(2), 382–395 (2008)
13. Witten, I.H., Moffat, A., Bell, T.C.: Managing Gigabytes, 2nd edn. Van Nostrand Reinhold (1999)
14. Byrd, D., Schindele, M.: Prospects for improving OMR with multiple recognizers. In: Proceedings of the 7th International Conference on Music Information Retrieval (ISMIR 2006), pp. 41–46 (2006)
15. Jones, G.: Sharpeye music reader (2008), `http://www.visiv.co.uk/`
16. Kurth, F., Müller, M., Fremerey, C., Chang, Y., Clausen, M.: Automated synchronization of scanned sheet music with audio recordings. In: Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR 2007), September 2007, pp. 261–266 (2007)
17. Krottmaier, H., Kurth, F., Steenweg, T., Appelrath, H.J., Fellner, D.: Probado — a generic repository integration framework. In: Kovács, L., Fuhr, N., Meghini, C. (eds.) ECDL 2007. LNCS, vol. 4675, pp. 518–521. Springer, Heidelberg (2007)