A Digital Library Framework for Heterogeneous Music Collections—from Document Acquisition to Cross-Modal Interaction

Dissertation

zur

Erlangung des Doktorgrads (Dr. rer. nat.)

der

Mathematisch-Naturwissenschaftlichen Fakultät

der

Rheinischen Friedrich-Wilhelms-Universität Bonn

vorgelegt von

David Damm

aus

Köln

Bonn, März 2013

Angefertigt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät der Rheinischen Friedrich-Wilhelms-Universität Bonn

1. Gutachter: Prof. Dr. Frank Kurth
2. Gutachter: Prof. Dr. Michael Clausen

Tag der Promotion: 31.07.2013
Erscheinungsjahr: 2013

A Digital Library Framework for Heterogeneous Music Collections—from Document Acquisition to Cross-Modal Interaction

David Damm

Abstract

In the digital age, increasing amounts of digitized music materials result in the need for automated management processes. Especially for libraries and other content-providing organizations holding large and due to ongoing digitization efforts steadily increasing amounts of digital copies of music materials, there is a high demand on automatisms to cope with the vast number of documents and streamline their processing.

In this thesis, a digital library system for managing heterogeneous music collections is developed. The aforementioned heterogeneity refers to various document types and formats, as well as to different modalities, e.g., compact disc-audio recordings, scans of sheet music, and lyrics. The system offers a full-fledged, widely automated document processing chain: digitization, indexing, annotation, linking, access, and presentation. The system is implemented as a generic and modular music repository based on an extensible service-oriented architecture. As a particular benefit of the approach pursued in this thesis, the various documents, representing different aspects of a piece of music, are jointly considered in all stages of the document processing chain. Concerning retrieval functionalities, incorporated state-of-the-art music information retrieval techniques and adequately designed user interface components allow for integrated, synchronized, and multimodal presentation of documents (WYSIWYH: what you see is what you hear), cross-modal score- or lyrics-based navigation in audio and vice versa, as well as sophisticated cross- and multimodal retrieval. Hence, the type of repository proposed in this thesis might be called a truly *cross-modal* music digital library system.

This thesis describes a complete framework that the system is based on including business processes and system architecture, exposes applied music information retrieval techniques incorporated in the document processing chain, and illustrates implemented functionalities for user interaction. As a part of the German PROBADO digital library initiative, with a view on practical application and integration into existing business processes, it is described how the framework is put into practice at the Bavarian State Library Munich which houses a music data stock whose volume is of internationally significant rank.

Keywords: Music Digital Library Framework · Multimodality · Multimodal Fusion · Multimodal Search · Cross-modal Navigation · Retrieval Result Visualization · Content-based Retrieval · Music Synchronization · Music Information Retrieval

Acknowledgements

This thesis has been developed in the Multimedia Signal Processing working group headed by Michael Clausen at the Department of Computer Science III, University of Bonn, under the supervision of Frank Kurth (Efficient Signals Intelligence group at Fraunhofer Institute for Communication, Information Processing and Ergonomics FKIE). The work underlying this thesis has been funded in part by the German Research Foundation (DFG INST 11925/1-1), in the context of the PROBADO ("PROtotypischer Betrieb Allgemeiner DOkumente") Music project.

I am grateful to many people who supported me during the preparation of this thesis. Therefore, at this point I would like to thank the following people for their invaluable help in preparing this thesis. First of all, I would like to express my thanks to my supervisors Frank Kurth and Michael Clausen, who always supported, advised, and guided me in an exemplary manner. Through these people I had the opportunity to get into science, work on interesting projects, and participate in international conferences. In particular, I want to thank Frank Kurth, who initially brought me to the topic of digital libraries in the music domain and guided me in an excellent way. Moreover, I had the opportunity to finalize this thesis while being with Fraunhofer FKIE. Also, I would like to thank Michael Clausen for making possible my stay in Shanghai, China, which was initiated by Axel Mosig at the PICB in Shanghai, and funded by the Robert Bosch Stiftung. Thank you for the rewarding and exciting time.

I would also like to thank my former colleagues at the University of Bonn for many helpful discussions, assistance, and the pleasant working atmosphere, in particular Sebastian Ewert, Christian Fremerey, Harald Grohganz, Verena Thomas, and Meinard Müller. Also, I would like to thank Raoul Wessel for inspiring discussions. I also thank the student assistants Markus Schäfer and Joscha Rieber for good collaboration. Furthermore, I would like to thank the technical support group for the premier technical support, in particular Thomas Fuchs, Peter Lachart, and Walter Steiner, and the secretary team, in particular Martina Doelp and Heidi Georges-Hecking.

Finally, I would like to thank my family, Gabriele Damm, Corinna Voglis, Elmar Damm, and Eva Nellessen for their support, love, and patience.

Bonn, March 2013

David Damm

Table of Contents

Chapter 1

Introduction

This thesis deals with the automated acquisition and further processing of digitized music materials, where the term music materials in general encompasses diverse manifestation forms, of which no doubt the most prominent are acoustic and graphical renditions, i. e., performances and sheets of music. To this end, a flexible solution aiming at achieving a robust and open music digital library (DL) system for the semi-automatic creation, processing, and dissemination of digitized music materials has been developed.

The major topic of this thesis is the cross- and intermodal indexing, multimodal retrieval, and modality-fused presentation of digital copies of music documents by means of state-of-the-art music information retrieval (IR) techniques. This chapter motivates a framework for the construction of a music DL system, discusses goals, and outlines practical solutions and concepts to preprocess and disseminate large-scale musical data stocks typically held by libraries and other content-providing organizations, generally referred to as content providers.

This chapter is structured as follows. Section 1.1 gives some background on the motivation for this thesis. Section 1.2 defines the goals of this thesis. Section 1.3 discusses strategies and scientific challenges. Section 1.4 exposes the contributions and scientific impact of this thesis. Section 1.5 relates this thesis to a concrete application context. Section 1.6 gives an overview on the overall structure of this thesis.

1.1 Motivation

Recently, significant digitization efforts have been carried out for large collections of multimedia content, including books, newspapers, images, music documents, and videos. This inherently leads to the need of powerful tools that automatically process, analyze, and annotate the scanned documents, providing the basis for efficient and effective content-based searching, navigation, and browsing in the digitized data. The mere digital capture of documents, i. e., digitization and storage, does not or at least hardly poses a problem—scanned documents can mostly be created and stored automatically in a streamlined fashion by means of scan robots. However, especially for content providers holding a vast amount of digital music content that is steadily increasing due to ongoing digitization efforts, besides the mere capture there is a high demand for further processing techniques and automatisms to cope with the large number of documents. Two key challenges are how growing collections can be organized automatically and how users can be enabled to access documents in an adequate and intuitive way incorporating novel access mechanisms.

In the case of scanned text documents, various solutions for automated document processing have been proposed, which typically contain a component for optical character recognition (OCR) to extract the textual content from the images, as well as a component for fault-tolerant full-text indexing and retrieval. The general idea is to suitably combine the strengths of both types of data representations, i. e., scan and text, for a convenient navigation and search in the scanned documents. A well-known example for this is the Google Book Search project [53], where one can search and navigate in entire books.

In spite of these advances in the textual domain, there is still a significant lack of corresponding solutions for handling general digitized non-textual documents including musical data such as audio recordings and sheets of music, or graphical data such as images, videos, and 3D data. In particular, tools are needed to automatically extract semantically meaningful entities or regions of interest from the scanned documents and to create links between related entities.

1.2 Goals of this thesis

This thesis focuses on the domain of music. Increasing digitization of musical data of all kinds led to extensive and often unstructured music collections. In real-life application scenarios, those data stocks are in general heterogeneous and comprise various documents of diverse types and different formats, expressing musical content at different semantic levels and addressing different modalities. Explicitly addressing the visual and auditory modality, compact disc (CD)-audio recordings and scanned sheets of music play a major role in ongoing automated digitization efforts and constitute the main types of music representations considered in this thesis. In addition, there exist other music-related documents such as lyrics and libretti texts, album cover artworks, and score-like documents in digital formats such as MIDI [63] or MusicXML [52]. We refer to [116] for a survey of symbolic music formats.

As a main goal, this thesis aims at processing digitized music documents largely *automatically*. Particularly, the following tasks are addressed:

1. Digitization of music documents and creation of digital versions.

2. Indexing, annotation, and synchronization of digitized documents.

3. Design and development of a DL software and hardware system considering specific library requirements.

4. Design and development of administrative user interface (UI) components for data maintenance and quality assurance.

5. Design and development of UI components for document searching, presentation, navigation, and browsing.

1.3 Strategies and scientific challenges

Beyond the mere recording and digitization of musical data, the key challenges in a real-life library application scenario are the automated preprocessing and subsequent access to the musical data. On the one hand, methods for automatic annotation, linking, and indexing of different music documents are required. On the other hand, tools and UIs have to be devised for a unified and adequate presentation, navigation, and search in the documents that explicitly consider multiple available modalities.

We particularly target an application scenario, where sheets of music as well as audio recordings belonging to the same piece of music are available. Each of the documents represents the same musical content, however, using different modalities. The strategy pursued here is to exploit the availability of those multimodal data to implement the particular tasks of (a) automatic annotation via cross-modal alignment, i. e., the spatio-temporal alignment or *synchronization* of scanned sheet music images and audio recordings, (b) cross-modal navigation, i. e., using the spatio-temporal alignment to synchronously navigate and present specific parts in both representations, (c) cross-modal search and retrieval, where a query is formulated in one modality while results may be of another modality, as well as (d) multimodal search, where queries may be composed using information from multiple modalities, like, e. g., lyrics and score fragments.

Over the last decade, research efforts in the field of music IR have produced various methods for automatic content-based analysis, synchronization, indexing, as well as retrieval of and navigation in music documents. In contrast, to this date there is a lack of suitable frameworks that provide *integrated* music IR solutions for the usage in real-world music DLs. Such frameworks should exploit and systematically combine available music IR techniques with appropriate tools and UIs for multimodal music access, including cross-modal search, navigation, and multimodal playback. In this thesis, such an integrated approach to a music DL system is proposed, see Chapters 4, 5, and 6.

1.4 Contribution and scientific impact of this thesis

The framework proposed in this thesis incorporates methods for the automated organization of large collections of digital music documents by exploiting state-of-the-art music IR techniques. It further comprises a preprocessing workflow consisting of feature extraction, audio indexing, and music synchronization. An essential objective of practical relevance is to develop a generalized workflow that allows libraries, specifically its staff, to efficiently handle collections of digital music documents while minimizing required administration effort for managing the documents. Considering user interaction, novel and easily operated UIs are offered for the multimodal presentation of music, particularly audio-visual playback, cross-modal navigation, and cross-modal content-based search and retrieval. The underlying design concepts of the UIs are intended to ensure that the access to music documents is as natural as possible, i. e., as one is accustomed to from their respective physical equivalents. With this, we aim at bridging parts of the gap between the real and the digital world and to give users an intuitive way to handle and explore music documents. In the context of the PROBADO digital library initiative [73], both preprocessing tools and UIs had to be integrated into the library service of the Bavarian State Library ("Bayerische Staatsbibliothek", BSB)[1] Munich. The latter holds

[1] http://www.bsb-muenchen.de/

a large amount of music documents, particularly collections of scanned sheet music books and CD-audio recordings along with associated metadata.

The main contributions of this thesis are summarized as follows:

- A complete music DL framework is proposed comprising (a) a system architecture, (b) a data model for organizing both heterogeneous document collections and associated metadata, as well as (c) a full-fledged workflow for content-based document processing.
- For the design of the proposed framework, several real-world requirements are identified; needs-driven and practical solutions are developed, guided by a real-world application scenario. The latest version of the system is currently installed at three different sites, including the BSB Munich as part of the PROBADO digital library initiative, and available to the public.
- As a fundamental paradigm of the proposed system, cross-modal document processing is exploited in all stages of the document processing chain, using state-of-the-art music IR techniques.
- To facilitate cross- and multimodal retrieval, a sophisticated retrieval strategy is proposed that allows for *composite* queries which can be constructed by a weighted linear combination of a number of partial queries, each of which may independently address an individual modality.

1.5 The PROBADO digital library initiative

The framework presented in this thesis relates to the PROBADO digital library initiative. The PROBADO digital library initiative is part of a total of four projects belonging to the Centers of Excellence (“Leistungszentren”) funded by the German Research Foundation (DFG). The PROBADO digital library initiative is a multilateral joint project between research facilities and libraries with the major aim to create and develop frameworks for building prototypes of next-generation multimedia DL services for digital content dissemination, focusing primarily on non-textual documents with the approach of a user-centric view. The objective of building up digital libraries is to provide an intuitive operable user interaction environment for the convenient access and handling of digital multimedia content (documents), enriched with value-added functionalities that do not exist in the analog world. For this purpose, a complete framework for developing and integrating music DL services in content providers has been created, from ingesting documents up to an value-added cross-modal interaction with them, including catalog- as well as content-based searching, browsing, and accessing, utilizing modern state-of-the-art technologies and computational methods. In the context of this project, beside the (re-) usage of a wide range of useful state-of-the-art techniques from the field of music IR and evaluation of best practices, new problems raised and research has been conducted, whose contributions related to the author of this thesis have been published in [33, 76, 32, 31, 48, 35, 123, 10, 9, 34, 122]. A prototypical implementation of a music DL service based on the framework is set up at the BSB Munich.

1.6 Structure of this thesis

This thesis is structured as follows. In Chapter 2, related work, existing approaches, and systems in the context of music and multimedia DLs are discussed. Some of the relevant music IR techniques are briefly summarized. In Chapter 3, fundamentals on music representations with an emphasis on their cross-modal interrelations are described. The presented work-centric data model is capable of adequately mapping complex interrelations on multiple levels between musical entities. As the first major part of this thesis, Chapter 4 proposes a complete framework for constructing a real-life music DL, including business processes and a flexible service-oriented architecture (SOA) for the distribution of musical content over the Internet. As the second major part of this thesis, Chapter 5 presents fundamental music IR techniques that lay the foundation for cross-modal indexing and retrieval of music documents, and describes the steps of a document processing chain for cross-modal music processing or indexing. The third major part, Chapter 6, presents UI components providing functionalities for multimodal music access comprising retrieval, presentation respectively playback, navigation, and browsing. Concluding, Chapter 7 summarizes the application scenario of the system at the BSB Munich within the PROBADO digital library initiative and gives some prospects on future challenges and ongoing work.

Chapter 2

Related work

In this chapter, related work and systems in the context of music and multimedia DLs are discussed. We will see that there is great public interest on this topic and a number of initiatives have been established in recent years to build up DLs on a large scale.

The present chapter is structured as follows. Section 2.1 gives some background on music IR, techniques, and systems. In Section 2.2, some background on librarianship is given. Section 2.3 provides the reader with some history on the evolution of DLs, and exposes existing related music- and general-media DLs.

2.1 Background on music information retrieval

As subarea of IR, music IR is an interdisciplinary research area that focuses on the specific domain of music. Since 2000, its diverse disciplinary communities meet at the recently established International Society for Music Information Retrieval (ISMIR)[1] conferences.

2.1.1 Music IR

Music IR is a growing research area in a number of disciplines including audio signal processing, IR, as well as library and information studies. Within music IR, there are several subfields, all subjects of current research and cutting-edge technology. The music IR community so far produced various methodological principles to analyze, classify, categorize, and organize digital music collections by means of extracting high-level semantics, aiming at managing collections of digital music documents. Nowadays, music IR is a field of rapidly growing commercial interests like, e. g., the music identification service Shazam [132].

2.1.2 Music IR techniques

As mentioned before, up to now the music IR community suggested various approaches towards the automatic processing of musical data such as indexing methods and content-based retrieval.

[1] `http://www.ismir.net/`

In particular, the indexing methods include different synchronization methods for the automatic cross-linking of two data streams of different formats such as the linking of CD-audio recordings with symbolic score-like formats such as MIDI [2, 39, 62, 112, 120, 126], lyrics [133] or scanned sheet music [98, 37, 80]. Furthermore, different algorithms were developed to determine the structure and repetitions of representative parts of audio, see, e. g., [6, 55, 85, 91, 102]. In addition to traditional text-based methods for music search based on annotations and symbolic music data [25, 128], content-based search methods that work directly on the audio data have been proposed recently, see, e. g., [20, 77], as well as cross-modal search, see, e. g., [103, 121, 51, 35]. For an overview on the issues of the development of automated music data indexing, we refer to [71, 88, 100].

2.1.2.1 Music transcription

Manual music transcription is the process of listening to a piece of music and writing down music notation for that piece. In automatic music transcription, an algorithm that takes an audio signal as its input, attempts to recognize and capture pitches, onset times, and durations of the notes that are present in the sound signal, plus loudness, dynamics, and performance style (vibrato, portamento, glissando); see, e. g., [7]. The aim is to obtain a representation of the input signal as sequence of notes in a standard music notation (plus loudness, dynamics, and performance style such as vibrato, portamento, and glissando) being preferably very close to the original performance. That is, it attempts to find a—generally non-unique—symbolic representation of music that a human listener would write down for the same input signal. In this sense, automatic music transcription is similar to the task of automatic speech recognition that aims at recognizing textual content from a recorded speech signal. By the current state of scientific knowledge, automatic music transcription still remains an unsolved scientific problem, as this is a complicated cognitive task that can only be properly performed by highly skilled humans.

2.1.2.2 Optical music recognition

Computer-aided optical music recognition (OMR) is the procedure of an automatic extraction of a symbolic representation from music notation in form of image scans of printed sheets of music. In OMR, an algorithm takes a set of sheet music scans as input and attempts to recognize and capture musical notes, stems, beams, measure bar information, etc. The aim of OMR is to obtain a symbolic representation of the input as a sequence of notes in a standard music notation which is preferably very close to the original abstract musical information content. That is, OMR attempts to find a—generally non-unique—symbolic representation of visual music that a human would write down for the same set of sheet music pages. In this sense, OMR is similar to the task of OCR that aims at recognizing textual content from image scans of printed text documents. By current state of scientific knowledge, OMR still poses many scientific problems, as this is a complicated cognitive task that is performed by trained human musicians capable of reading music.

2.1.3 Music IR systems

A music IR system provides several means for music retrieval, which can be the identification of a hummed audio signal (query-by-humming scenario), but also music genre classification or retrieval of text information about the artist or title. A survey of existing music IR systems is presented in [128]. A prominent example for a non-commercial music IR system supporting query-by-humming is Musipedia, see Subsection 2.3.5.4.

2.2 Background on librarianship

Classically, libraries are concerned and deal with physical objects, representations of knowledge, culture, information, facts, and beliefs [8]. The stocks libraries are hosting range from books to complex multimedia documents that are physically available in the library and require a large number of shelves for storage.

2.2.1 Content organization in libraries

Libraries organize their holdings and resources in library catalogs. In the Anglo-American cataloging rules, a library catalog is defined as a “list of library materials contained in a collection, a library, or a group of libraries, arranged according to some definite plan.” [54]. Library catalogs comprise bibliographic records for resources making up a collection. The purpose of creating and maintaining library catalogs is that library customers are enabled to efficiently find what they search for. A bibliographic record includes a description of a corresponding resource, containing standardized information such as contributing authors and title, publication details, and other kinds of information under which that record can be found. In order to provide a standardized way of describing resources being cataloged, the International Standard Bibliographic Description (ISBD) is a set of rules produced by the International Federation of Library Associations and Institutions (IFLA)[2]. A general framework, referred to as ISBD(G), was originally published in 1977 [140] and revised most recently in 2004 [66]. Libraries also share their records in cataloging networks, from which the world’s largest cataloging network is the Online Computer Library Center (OCLC)[3] which covers over 80 million cataloged resources.

2.2.1.1 Cataloging

In the context of traditional libraries, the term *indexing* is concerned with the cataloging, i. e., the preparation of bibliographic information for catalog records, of the holdings of one or more libraries. The result of the indexing process are the library catalogs. Therefore, the task of indexing is also called *cataloging*, and the two terms are sometimes used interchangeably. For consistency reasons, catalogers in most cases use a set of indexing tools that are based on widely accepted international rules and standards.

The task of indexing can be further subdivided into *descriptive cataloging* and *subject indexing*, which are briefly explained in the following.

1. Descriptive cataloging is the cataloging of a resource (e. g., a book) for the purpose of bibliographic evidence in a library’s *alphabetical catalog*. The alphabetical catalog includes the bibliographic description of the resource as well as the determining of formal

[2] http://www.ifla.org/

[3] http://www.oclc.org/

features and search terms, i. e., the names and the property titles under which to file the entries in the alphabetical catalog, and under which records can be searched in order to locate and find the resource. A property title is the factual naming of a work.

2. Subject indexing is the content description and indexing of a resource. Here, a distinction is made between

   (a) *verbal indexing* that uses mainly natural language terms (keywords and tags), and

   (b) *classificatory indexing* which is primarily based on hierarchical classification systems with *identifiers* determining the membership of a particular group.

2.2.1.2 Access points

A catalog record is found by an author's name, title, description, or subject. Since those entries give access to the record, they are called access points. Catalogers determine access points using cataloging rules, with particular attention to what users are likely to search for.

2.2.1.3 Formats of catalogs

Traditionally, library catalogs, more precisely the records, have been stored on cards, books, and microfiches. Increasingly, library catalogs are available in electronic form, i. e., the records are stored on computer systems, and customers find their information using a computer. Today, the most commonly accepted and widely used type of catalog is the online public access catalog (OPAC). While card-based catalogs still provide a flexible, user-friendly method of storing and retrieving library records for small-sized libraries, these are no longer eligible for medium- and large-sized libraries in the course of streamlining indexing processes. By means of sophisticated electronic catalogs, customers are enabled to look up almost any piece of information, particularly combinations of details, in order to find certain records they look for.

2.2.2 Manual cataloging and classification versus machine-made indexing

The verbal indexing is subjective and varies not only over time but also from individual to individual. Moreover, it depends on daily condition. For these reasons, it is not a "unique method" approach. The classificatory indexing is more objective because determination of classes is made by many people in consent and matures over time. Certainly, the correct indexing of items into appropriate classes depends on individuals, again.

These two indexing approaches stand against computer-aided classification techniques. The latter produce determinate classification results and are hence objective. They rely neither on time period nor on human beings. In particular, they work uniformly in the sense that they are not affected by individual and varying personal properties such as background of knowledge, taste, and mood. However, by the current state of scientific knowledge, classification made by machines is not nearly as accurate and reliable as by humans.

Another purpose of content-based indexing methods is that, given a resource, library customers can be enabled to find further, similar resources. Here, similarity refers to the automatically extracted actual content (more precisely, local features) instead of manually created high-level descriptions of that resource. Resources that are similar—in some sense—w. r. t. certain features can then be retrieved by means of feature-based retrieval techniques. This approach follows the *query-by-example* paradigm, where the customer gives the search system an excerpt of a resource as example of what he is searching for and, in turn, obtains a list of similar resources from the system as result. Note that this kind of search service has only been recently made possible by latest developments in the field of IR.

2.2.3 Digital services of libraries

For decades now, libraries offer digital services. Most of them became indispensable such as the electronic library catalog [97], but also the ever-growing range of media such as books and articles are being offered in digital form. With the upcoming of new technologies, limitations of traditional libraries are more and more removed. Large-scale mass digitization projects result in a significant change in dealing with physical objects, which, in turn, may have fundamental impact for library services.

However, technical conditions have significantly changed in recent years—and with them, the demands of library customers have also grown. Meanwhile, it is a matter of course to have everywhere and at any time access to the Internet, whether by desktop computers or by mobile devices. Many libraries are therefore enforced to look for new digital ways to support their customers in their needs. This especially includes tools and mechanisms for the dissemination of digital content over the Internet.

2.2.3.1 Digital repositories

To provide this kind of service, contents are stored and managed in *digital repositories*. Those consist of the actual contents and access mechanisms provided by a service interface, through which contents are remotely accessible using a network infrastructure. With them, data can be provided very quickly and reliably over the Internet. Repositories allow the access to digital documents without the restrictions associated with the classical way of gaining access to their physical equivalents. Moreover, repositories can also ensure that objects can be found globally, not only by human beings, but also by machines. This, in turn, allows for the creation of new and larger systems that are built on a distributed network comprising and utilizing single, small services.

2.2.3.2 State-of-the-art

Currently, many applications are still developed as detached, local services that do not fit in the context of a global communication network. There is still a lack of standards that must be taken into account, or, that have to be developed first. A key challenge towards this direction is how libraries can provide their digital services permanently and in a coordinated way. To tackle this challenge, new ways of collaboration between libraries and developers are to be established. It should, however, also be mentioned that even if standards regarding key technologies that these services rely on are enforced, the fast moving character of information and communication technologies entails that DLs will be constantly evolving.

2.2.4 Digitization of materials

Before analog materials such as books or other types of information-carrying objects can be integrated into the holdings of a DL, they—more precisely, the information content thereof—must be available in a digital form. For this purpose, the material has to be *digitized*. *Digitization* is the process of converting an analog object, image, sound, document or a signal into a digital form, i. e., the analog information is captured and then sampled. The result of this process is referred to as *digital copy* or *digital representation* which embodies the information content of the original object by a discrete set of points or samples. The making of a digital copy of an analog object consists of two steps in general: (a) analog acquisition (capturing), followed by (b) analog-digital conversion with subsequent quantization, performed by a technology readily available in the form of analog-digital converters. Nowadays, these two steps are integrated and assembled in digital recording devices such as sound recorders, cameras, or other capturing devices such as scanners that allow to convert analog sounds or images into digital audio recordings or images, respectively.

2.3 Music- and general-media digital libraries

Over the last years, DLs have taken over a central role in our society. DLs provide fast access to digital information collections of vast amounts encompassing the full range of multimedia data of any kind including textual, numerical and graphical data, scanned images and graphics, audio and video recordings, 3D architectural data, just to name a few. The process of acquiring, creating, processing, retrieving, and disseminating of knowledge, information, data, and metadata has undergone and still continues to undergo significant changes. This includes an ever-increasing public access to online resources, an evolution and proper explosion in the amount and diversity of resources that are available over the Internet, and a social shift in the paradigm of how to experience and utilize information. In the digital age, the access to knowledge being promoted in this way also contributes to the preservation of cultural heritage (CH).

2.3.1 Historical background on the development of digital libraries

Libraries as institutions for CH preservation are facing the problem of the imminent decay of documents over time. Some of those documents are unique exemplars and accordingly of high importance and priceless value. To make sure that future generations have access to these stocks, they must be conserved if possible. A major technology available today for preserving documents—more precisely, the information content carried on the underlying materials—is digitization. On the one hand, in this way not all properties of the originals can be captured and preserved, e. g., information about the chemical composition of the underlying matter for purposes of later investigations. But on the other hand, besides the fact that this procedure by the current state of scientific knowledge is the only choice for saving the stocks, at least the manifested information content carried on the originals can be saved and preserved, and that in a durable and reliable way. Once digitized, a document's digital copy can be reproduced as often as desired without loss of quality using redundant storage and error-correction technologies. Therefore, libraries have massively begun to digitize their stocks in the course of ensuring the preservation of CH materials.

Furthermore, the availability of both powerful computers at affordable costs and the spreading of communication networks with an ever-growing pervasion into all areas of life also led to a significant interest in creating and distributing digital content. This, in turn, led libraries from their traditional role as static storehouse of books to dynamic institutions of accelerated information content generation and its dissemination. Through the integration of indexing and retrieval techniques, nowadays' DLs can offer unimagined possibilities of user experience and interaction w. r. t. the handling of multimedia content that have never been possible before in the domain of analog documents.

These forces have driven the development and acceptance of DLs, and that in a global context—information is now accessible at a high pace from everywhere and at any time.

2.3.2 Definition of a digital library

A DL consists of information in digitized form and provides mechanisms for the storage, manipulation, and dissemination of the information. The information stored in a DL is accessed through electronic gateways based on communication infrastructures such as intranet or Internet and a predefined protocol. A DL covers the creation and distribution of all types of electronic documents in diverse digital formats ranging from converted materials to kinds of information that have been generated in the physical world along with textual annotations or high-level information about the documents. The access to a DL or its content is usually done remotely by network-connected client computers. Therefore, DLs help to meet the information needs of customers with greater speed, accuracy and reliability, and without boundaries or particular restrictions in access w. r. t. location or time.

DLs are currently built up using interdisciplinary efforts. They apply technologies from various research areas, including non-exhaustively library sciences, data management and information systems, multimedia information retrieval, audio, image, and video signal processing, web technologies, human–computer interaction, and digital curation. Because of the multidisciplinary character it is inherently not possible to subsume the various aspects and concepts under a universal and generally valid definition of what a DL is. Rather, the term DL has a multitude of definitions and meanings, mainly depending on individual application use cases. The conceptual understanding of a DL has evolved substantially since the early idea of it being a system for providing electronic access to digitized text-based documents such as books [44, 16, 119, 67, 19]. For example, the DELOS Network of Excellence on Digital Libraries[4] envisions a DL as "a tool at the center of intellectual activity having no logical, conceptual, physical, temporal, or personal borders or barriers to information" [11].

A definition of DLs from the librarians' point of view was proposed by the Digital Library Federation (DLF)[5] as reported by Waters [134]: "Digital libraries are organizations that provide the resources, including the specialized staff, to select, structure, offer intellectual access to, interpret, distribute, preserve the integrity of, and ensure the persistence over time of collections of digital works so that they are readily and economically available for use by a defined community or set of communities." Three main characteristics that distinguishes DLs are (i) the storage of information in digital form, (ii) the usage of communication networks to access and obtain information, and (iii) the copying of information by either downloading or printing. In [27], Cleveland characterizes DLs as "the digital face of traditional libraries

[4] `http://delos.com/`

[5] `http://www.diglib.org/`

that include both digital collections and traditional, fixed media collections. [. . .] (DLs) will include all the processes and services that are the backbone and nervous system of libraries [. . .] (and) will require both the skills of librarians and well as those of computer scientists to be viable. [. . .] (DLs) will not be is a single, completely digital system that provides instant access to all information, for all sectors of society, from anywhere in the world. [. . .] Instead, they will most likely be a collection of disparate resources and disparate systems, catering to specific communities and user groups, created for specific purposes. They also will include, perhaps indefinitely, paper-based collections." Sharma and Vishwanathan state in [117] that "The growth of digital libraries involves: digitisation of existing library materials; connectivity to the users in the world online and offline; integration with networking; and availability on the World Wide Web".

2.3.3 Evolution of digital libraries

Driven by the developments of information and communication technologies in recent years, the explosion of digital content, and high user demands, a paradigm shift in the conceptual understanding of a DL has taken place from *content-centric* systems that support the mere storage, organization, and access to digital copies of documents towards *user-centric* systems that deliver innovative, evolving, feature-rich, and personalized services to users [19]. With this, the term DL has emerged to refer to systems that are heterogeneous in scope and provide diverse types of functionality. These systems consist of repositories of digital objects and metadata, reference-linking systems, long-term preservation archives, administrative document management systems, as well as intuitively operable UIs that allow for a previously impossible content-interaction. On the other hand, the proliferation of short-term solutions based on transient technical standards and changing trends in methods and tools result in DL services and systems that do not deliver interoperability and reuse of content and technologies.

Therefore, there is a great interest in creating systems that comply with long-lasting interface and protocol specifications guaranteeing long-term interoperability. Since the middle of the last decade, there have been world-wide initiatives of collaborations—some of which were substantially funded by up to half a billion U.S. dollars—between librarians, scientists, and system engineers, to envision and establish generic frameworks, standards, and protocols in order to create long-lasting, complex interoperable systems that integrate and deliver advanced DL services. The most prominent of such systems and frameworks towards this vision are discussed in Subsections 2.3.5, 2.3.6, and 2.3.7.

2.3.4 Functional requirements on multimedia digital libraries

With the increasing pervasion of information and communication technologies, traditional libraries storing information within a constrained physical space (e. g., books and other print materials on shelves) have given way to modern multimedia DLs that store electronic documents of various kinds such as texts, audio recordings, graphics, and videos on electronic storage systems of virtually unlimited space and efficiently disseminate them to customers using global communication infrastructures. As libraries gradually digitize their large-scale data stocks, one can speak of a veritable explosion of digital data that is created. This, in turn, has a direct impact on the librarians' daily tasks and the requirements that are imposed on a digital library system. A manual handling of created data volumes is virtually impossible with reasonable effort due to the resulting masses of data. Processes must be devised and streamlined to perform tasks better and faster even with increasing data volumes. This can only be achieved by incorporating intelligent, high-graded automatisms and computer-aided workflows.

2.3.5 Music-related digital libraries

In recent years, several DL systems for music documents were developed. Some of these systems include printed music (e. g., sheet music and musicological books) and various systems are currently available [114, 129, 130, 131, 137]. In [60], Hankinson et al. evaluated several of these music DL systems w. r. t. their UIs and identified three main drawbacks that may be observed in most of the systems. First, the systems do not keep document integrity and present sheet music books as a series of separate images. Second, simultaneous presentation of related music documents is often not possible. As third drawback, the metadata of a currently selected music document can not be accessed at a glance, omitting further valuable information.

Besides those shortcomings, these systems restrict the user in the possibilities of experiencing a musical work. As music can be expressed through various representations, a music DL system should offer the access to as many different representations as possible. It should be noticed that the notion of multimodality incorporated in the way some initiatives (like, e. g., EASIER or Musipedia) use it, does not refer to the simultaneous, amalgamated presentation of various modalities as it is used in this thesis. These initiatives use the term multimodality only in the meaning of different media formats that are available to the user.

A serious deficit of most music DL systems mentioned here is that nearly none of them allows for comparable content-based search functionalities as offered by DL systems for textual documents. Those systems are mostly restricted to metadata search functionalities. However, there are various music IR techniques available which would enhance the functionalities of a digital music library system by, e. g., allowing for a content-based and multimodal search, cf. Subsection 2.1.2. Two systems, already fulfilling most of the latter requirements for a music DL system are Variations2 and EASAIER, see Subsections 2.3.5.1 and 2.3.5.2, respectively.

Figure 2.1 depicts a chronological classification of related music- and general-media DL systems and frameworks. The DL framework considered in this thesis was developed in the context of the PROBADO[6] project, see Chapter 7 for more information.

2.3.5.1 Variations2

Variations2[7] [40] is a prominent music DL system for educational institutions to share music collections throughout the class room. It is developed and hosted at the Indiana University[8] in Bloomington, USA. The data stock offered is provided by the local Cook Music Library[9], which, with over 700 000 items, is recognized as one of the largest academic music libraries in the world. It holds documents of Western classical music, primarily sheet music and audio recordings. The system offers access to more than 10 000 audio recordings and hundreds of digitized scores

[6] http://www.probado.de/

[7] http://variations.indiana.edu/

[8] http://www.indiana.edu/

[9] http://www.libraries.iub.edu/

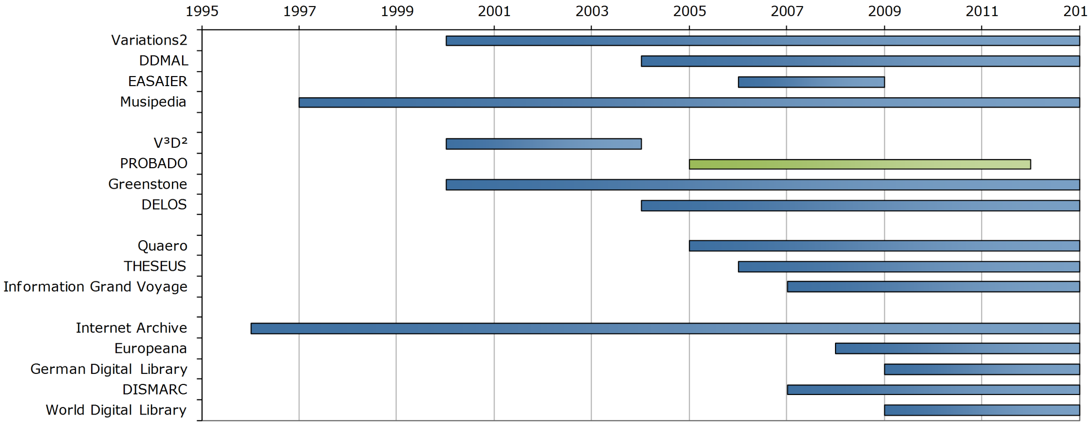


Figure 2.1 Chronological classification of related music- and general-media DL systems and frameworks.

in a *work-centric*[10] fashion, utilizing the Functional Requirements for Bibliographic Records (FRBR) [64] model which also serves as the basis for the music DL data model presented in this thesis. Besides UIs for the simultaneous presentation of metadata information, audio tracks, and sheet music, the system offers tools for manual music analysis, particularly musical structure and musical beat.

However, no music IR techniques for automatically generating structure information or cross-modal synchronization are incorporated. At least, the system supports a synchronized playback of sheet music and audio recordings, but the synchronization data has to be created manually. Efforts in this direction have been undertaken, but not been integrated into the system so far [113]. To offer enhanced search functionalities, a query-by-humming system is proposed [12, 13]. Among other features, the follow-up project Variations3 [65] concerns the setup and operation at other locations like the general-media DL frameworks Greenstone and DELOS, see Subsections 2.3.7.1 and 2.3.7.2, respectively.

2.3.5.2 EASAIER

EASAIER[11] [36, 82] is a European research project based at the Centre for Digital Music (C4DM) at the Queen Mary University of London[12], UK, that enables access with simultaneous presentation of various music representations such as audio recordings, score documents, and images. In addition to content-based retrieval mechanisms, several different audio analysis and processing tools are available, like, e. g., time stretching and source separation. In the current state, the project does not yet use music IR techniques for automatically calculating structure information or cross-modal synchronization between different representations of a piece of music. Also, there is no possibility of manual synchronization between different representations of a piece of music, as it it also the case for Variations2 or DDMAL. The project supports the visual presentation of sheets of music and acoustic playback of audio recordings, but not in an interlocked fashion, e. g., for the purpose of a visual score following alongside audio playback.

[10] See also Subsection 3.5.6.

[11] `http://www.elec.qmul.ac.uk/easaier/`

[12] `http://www.qmul.ac.uk/`

2.3.5.3 DDMAL

The Distributed Digital Music Archives and Libraries Lab (DDMAL)[13] is a joint project between the Swiss RISM[14] organization and the McGill University[15] in Montréal, CA, that in a collaborative effort develops and evaluates best practices, frameworks, and tools for the design and construction of worldwide distributed digital music archives and libraries. [60] reports on efforts in the direction of a synchronized playback of sheet music and audio recordings of the same piece of music that have been undertaken, but the synchronization structure has to be created manually, cf. the Variations2 project, see Subsection 2.3.5.1.

2.3.5.4 Musipedia

Musipedia[16] is a collaborative music encyclopedia providing a melody search engine for identifying pieces of music, which is based on the query-by-humming system Melodyhound [110]. A search can be performed fourfold by either whistling a theme (query-by-humming scenario), playing it on a virtual piano keyboard, tapping the rhythm on the computer keyboard, or by entering a rough description of the melodic contour as Parsons code [109]. The community-supporting platform enables individuals to modify the collection and enter MIDI files, sheet music images, lyrics, textual information such as work and composer names, and melodic contours using Parsons code. The search engine provides a SOAP interface through which a search can be conducted mechanically and thus offers the possibility that the search engine can be integrated into other systems and utilized from within other contexts.

2.3.5.5 Sheet music sources

Besides entire music IR systems that consist of multimodal data stocks, there also exists a number of small- and mid-scale community-driven, public-domain web portals that offer access to electronic sheet music of primarily Western classical pieces of music, including Digital Image Archive of Medieval Music[17], Munich Digitisation Centre (DIAMM)[18], Chopin Early Editions[19], Neue Mozart Ausgabe (NMA)[20], Sibley Music Library[21], International Music Score Library Project (IMSLP)[22], Mutopia Project[23], Werner Icking Music Archive (WIMA)[24], Choral Public Domain Library[25], and Wikifonia[26]. These web portals provide content that can freely be downloaded and used.

[13] http://ddmal.music.mcgill.ca/

[14] http://www.rism-ch.org/

[15] http://www.mcgill.ca/

[16] http://de.musipedia.org/

[17] http://www.diamm.ac.uk/

[18] http://www.digital-collections.de/

[19] http://chopin.lib.uchicago.edu/

[20] http://www.nma.at/

[21] http://www.esm.rochester.edu/sibley/

[22] http://imslp.org/

[23] http://www.mutopiaproject.org/

[24] http://icking-music-archive.org/

[25] http://www.cpdl.org/

[26] http://wikifonia.org/

2.3.5.6 RISM and RISM-OPAC

The Répertoire international des sources musicales (RISM)[27] is an organization with participating working groups of 34 countries around the world that gathers information about music resources in each country and makes it available both nationally and internationally. This includes books, manuscripts, music scores, maps, photographs, videos, analog tapes, and phonograph records. The working groups are usually based at the country's respective national library, at a musicological institute, or operate as an autonomous enterprise. One recent outcome of a fruitful collaborative work together with the BSB is the provision of the online catalog RISM-OPAC that contains incipits of over 700 000 musical works comprising mainly historical handwritten notes (the majority originated before 1800). In addition to a traditional metadata-based search, the incipits have recently become searchable by means of symbolic score-based queries.

2.3.6 Large-scale general-media digital libraries

2.3.6.1 Europeana

As the flagship of an European cultural heritage web portal and example for a large-scale multimedia DL, the Europeana [41] has been launched in 2008. At time of launch, Europeana covered two million digital objects from data stocks of cultural institutions of the EU including libraries, archives, and museums. It offers public online-access to a large data stock including texts, paintings and other types of images, music, and video material. The data stock also contains large amounts of music-related documents such as score documents and audio recordings, primarily from the Western classical genre. Over 1000 cultural organizations from all 27 EU countries across Europe have provided material to Europeana. With the stated goal of providing access to at least 15 million digitized objects by the year 2015, the creation of Europeana is a key step in preserving and disseminating Europe's cultural heritage. It is worth to note that at time of launch an estimated 10 million hits per hour caused the provider to take the platform down in order to double the computing power capacity before re-launching it—what indicates the great public interest in such large-scale multimedia digital libraries. The user interface is available in all official EU languages and for the next years, the portal will make use of semantic technologies to enable cross-language searches. That is, a searching for, e. g., "painting" will also lead to results as with "Gemälde", "peinture", etc.

2.3.6.2 German Digital Library

The German Digital Library (DDB)[28] is a digital library that will be integrated at European level in the Europeana, initiated in 2009 by the German Federal Government. As a newly networked institution to be established, it covers library materials of 30 000 German cultural and scientific facilities to be made available to the public on its common platform. The library comprises digital copies of books, works of visual art, music, and movies. The construction and test operations were planned for 2011.

[27] `http://www.rism.info/`

[28] `http://www.deutsche-digitale-bibliothek.de/`

2.3.6.3 DISMARC

DISMARC[29] (DIScovering Music ARChives) is a web portal that provides access to discographic catalog information and audio content located in music archives and in private and public collections. The portal reveals large amounts of under-exposed, invaluable, European-owned, culturally-significant, scientific and scholarly original audio as well as other music-related material such as videos, photographs, books, documents, and manuscripts that are all part of Europe's cultural heritage from the early 20th century until today. Content providers such as archives, broadcasters, museums, universities, research institutes, and private collectors are enabled to provide their collections to the public. In the context of EuropeanaConnect, a gateway for the integration of other platforms and portals in Europeana, DISMARC serves as the Audio Aggregation Platform (AAP) for Europeana, through which content is made available to Europeana.

2.3.6.4 The Internet Archive

Another example of a general multimedia DL is the Internet Archive (IA)[30] [69]. It offers public access to collections of websites, music and video documents, as well as public domain books. The IA platform allows individuals to both download and upload digital material and provides unrestricted online-access to that material free of charge. As a member of the Open Book Alliance (OBA)[31], an organization concerned with the mass digitization of books like the Gutenberg project[32], the IA also supervises one of the world's largest book digitization projects. As a member of the American Library Association (ALA)[33], it is officially recognized as a genuine library by the State of California, USA, though it is not a library in the traditional meaning.

2.3.6.5 World Digital Library

As a further example of a general multimedia DL, the World Digital Library (WDL)[34] provides—as a collaborative effort between the National Library of Congress (LoC)[35] and the UNESCO[36]—free-of-charge public access to outstanding cultural documents from all over the world, aiming at preserving the collective memory of humanity and promoting international and intercultural understanding. The currently acting LoC's director Billington stated in his talk to the plenary of the UNESCO that fostering these global collaborative efforts poses an opportunity that cultural heritage materials that "[...] institutions, libraries, and museums have preserved could be given back to the world free of charge and in a new form far more universally accessible than any forms that have preceded it".

[29] http://www.dismarc.org/

[30] http://archive.org/

[31] http://www.openbookalliance.org/

[32] http://www.gutenberg.org/

[33] http://www.ala.org/

[34] http://www.wdl.org/

[35] http://www.loc.gov/

[36] http://www.unesco.org/

2.3.6.6 THESEUS and CONTENTUS

The THESEUS research program[37] was established in 2006 by the German Federal Ministry of Economics and Technology (BMWi) as a flagship project in the field of semantic search engines. It emerged from the Franco-German Quaero project (see Subsection 2.3.6.7) after the German Federal Government announced its withdrawal. The focus of the research program is set on semantic technologies, whose goal is to better understand the context in which information is stored, to identify the meaning of information, and to classify it. With these technologies, computer programs should be enabled to draw logical conclusions about content, and consequently the research program is making important contributions to the two recent developments, Internet of Services [21] and Web 3.0[38], where the finding of semantic interrelations between distributed information fragments on various granularity levels play a key role in scientific questions.

One of the basic technologies developed by the research partners include functions for automatic creation of metadata for audio and video documents, 2D and 3D images, and their combination, as well as mechanisms for the semantic processing of multimedia documents and their associated services. The focus of research includes the development of tools for the management of ontology-supported knowledge representation. In addition, new methods of machine learning (ML) and situation-sensitive dialog processing are developed. At the same time, also innovative UIs are developed. One application scenario where the basic technologies are applied on is concentrated in the sub-project CONTENTUS[39] whose aim is the development of new technologies for the creation of “media libraries of the future”—large-scale, next-generation multimedia DLs whose data stocks are made available online, such as, e. g., the German Digital Library and Europeana, see Subsects. 2.3.6.2 and 2.3.6.1, respectively.

2.3.6.7 The Quaero project

Initially started in 2005 as a Franco-German cooperation with similar goals as THESEUS, the Quaero project[40] is a large-scale research effort for digitization, content preparation, and management of digital media assets, focusing on personalized video services, image search engines and mobile portals. Since May 2008, 24 partners from companies and institutions are involved, including the Institut national de recherche en informatique et en automatique (INRIA)[41] and the Institut de recherche et coordination acoustique/musique (IRCAM)[42] on the French side, as well as the RWTH Aachen University[43] and the Karlsruhe Institute of Technology (KIT)[44] on the German side. Quaero and THESEUS complement each other and are closely linked through regular meetings of participating working groups.

[37] http://theseus-programm.de/

[38] http://www.w3.org/standards/semanticweb/

[39] http://www.contentus-projekt.de/

[40] http://quaero.org/

[41] http://www.inria.fr/

[42] http://www.ircam.fr/

[43] http://www.rwth-aachen.de/

[44] http://www.im.uni-karlsruhe.de/

2.3.6.8 The Information Grand Voyage project

The Information Grand Voyage project[45] started in 2007 as a Japanese analog of THESEUS addressing the threefold of services, technology, and legal developments to face the "Info-plosion" problem[46]. The technology development includes among others the development of audio-visual data navigation and browsing tools, as well as location-dependent referral services through mobile phones. Since 2009, the project is supported by THESEUS in continuation of its international activities for the employment of research and development programs [136].

2.3.7 General digital library frameworks

2.3.7.1 Greenstone Digital Library

The Greenstone Digital Library (GDL)[47] [138] is another interesting project in the context of multimedia DLs that has evolved from the New Zealand Digital Library (NZDL) project[48] at the University of Waikato[49]. It is developed and distributed in cooperation with the UNESCO and the Human Info NGO[50]. In contrast to the other projects introduced above, the GDL aims at offering software tools for the creation, management, distribution, and presentation of digital document collections. The GDL software provides tools and mechanisms for organizing information and publishing it on the Internet or on CD-ROM and enables users, particularly in universities, libraries, and other public service institutions, to build their own DLs. Some of the main features are the support of multimedia document collections, content-based retrieval, a basic, extensible UI, and a plug-in mechanism to extend functionality. Furthermore, an assembling tool for the creation of a DL from a given digital document collection was proposed [5].

2.3.7.2 DELOS

DELOS[51] is a "Network of Excellence on Digital Libraries" (NEDL), partially funded by the European Commission in the context of the Information Society Technologies (IST) program. The main objectives of DELOS are research, where research results are in the public domain, technology transfer through cooperation agreements with interested parties, as well as the development of next-generation DLs based on comprehensive theories and frameworks for the life-cycle of DL information. DELOS is conducting a joint program of activities aiming at integrating and coordinating ongoing research efforts of major European teams working in DL-related areas. DELOS is currently working on the development of a DL reference model that is designed to meet the requirements on next-generation DL systems, and on a globally integrated prototype implementation of a DL management system (DelosDLMS), where the latter serves as a concrete partial implementation of the reference model encompassing many software components developed by the DELOS partners.

[45] http://www.igvpj.jp/index_en/

[46] http://www.infoplosion.nii.ac.jp/info-plosion/

[47] http://www.greenstone.org/

[48] http://www.nzdl.org/

[49] http://www.cs.waikato.ac.nz/

[50] http://www.humaninfo.org/

[51] http://delos.info/

Chapter 3

Music documents and cross-modal interrelations

This chapter gives a general overview on fundamentals of music representations, notations or recordings, and documents. It covers digital music data formats found in practice in the context of music DLs and discusses how music documents are interrelated. Furthermore, it is discussed in which way music documents can be organized properly. For this sake, a modified, music-specific version of a work-centric data model, which has been established in the last years, is used. The adapted data model is powerful enough to adequately map all interrelations and mutual dependencies between document parts, both within and across different modalities. By means of this data model, a complex, hierarchically organized, and tightly linked navigation graph can be formed for a given collection of music documents.

3.1 Classical music representations and notations

A piece of music can be described or expressed by manifold forms and representations, various formats, and document types. Throughout this thesis, referring to a piece of music, we therefore formally distinguish between a *musical work* as intellectual creation in an abstract sense and its concrete manifestations. Examples of the latter are a *sheet music notation* or a particular *audio recording* containing a performance of a concrete interpretation, but also libretti or lyrics (text), a video recording of an orchestral performance, and other kinds of music-related materials such as album arts (images). In the real-life library scenario, probably the most widely used and recognized representations for pieces of music are printed sheets of music and associated CD-audio recordings. Consequently, the primary available music document types that we mainly concentrate on in the context of this thesis are sheet music and acoustic performances. Note that these document types describe pieces of music in different ways and fundamentally differ from each other. The involved concrete music documents and formats are presented in Section 3.2.

3.1.1 Sheet music

Sheet music is handwritten or printed form of music notation that uses musical symbols to describe music. More specifically, sheet music embodies a graphically arranged symbolic notation of a piece of music. Rather than an acoustic rendition of that piece shaping sound, it contains the *symbolic score* of that piece, which describes a composition on a high level of abstraction, regardless of a concrete interpretation and instrumentation. Its semantic meaning can be grasped by musicians who are able to read the graphical arrangement of individual musical symbols. This kind of representation allows musicians to rehearse and play particular compositions by means of musical instruments. In real life, sheet music is usually printed on pages or sheets of music, that are bundled together to sheet music books. Digitally available sheets of music are either digitally born documents or scans of printed sheets of music, which are available in specific image formats such as TIF, JPG, or PDF. In the context of this thesis, we focus on high-quality scans of printed sheet music books using Western notation style.

3.1.2 Performance

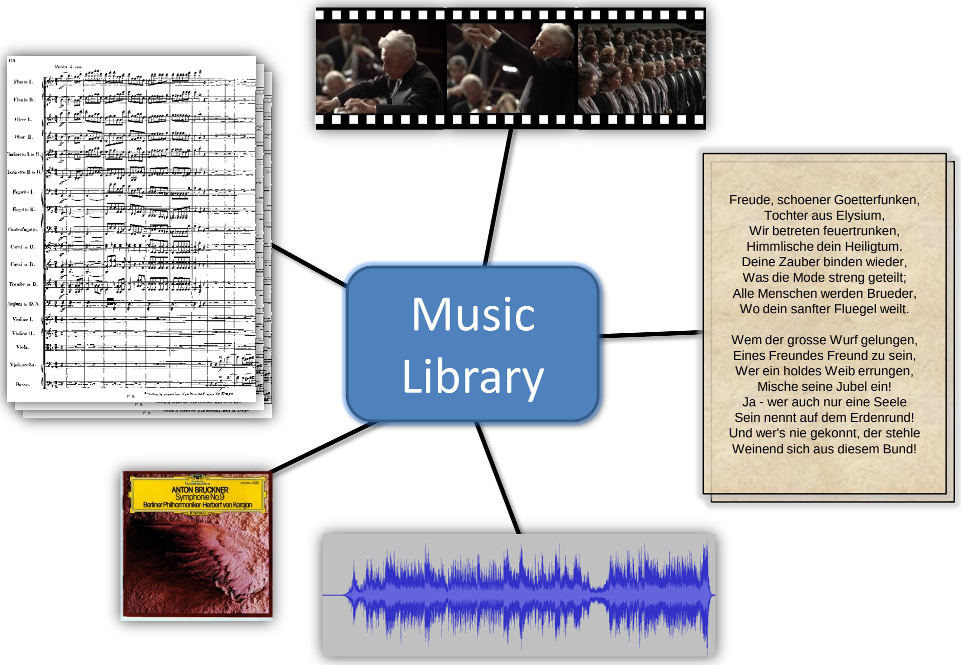
A performance embodies an acoustic rendition of a piece of music by means of sound-shaping musical instruments. Note that each performance, being unique, is an individual interpretation of a piece of music by performing musicians. The produced sound of a particular performance of a piece of music can be captured by an audio recording. A particular audio recording is stored on a medium such as a gramophone record or a CD, or is electronically available as a digital audio file encoded in a specific audio format such as WAV or MP3. Performances are recorded and distributed using a wide variety of media types and formats, some of which can store additional information such as textual metadata about the actual content, like, e. g., in the case of MP3-formatted files.

3.2 Digital music documents

In modern music DLs, heterogeneous document collections are stored, comprising large numbers of documents and a variety of document types. Various aspects of a piece of music can be described by these different types of music documents, such as sheet music and acoustic performances. Figure 3.1 depicts document types that are commonly found in the context of music DLs. In particular, music DLs contain a wide range of music data in form of acoustic data (e. g., audio recordings of performances), visual data (e. g., scans of sheet music, CD cover arts), textual data (e.g., lyrics, libretti, music analysis), symbolic data (e. g., MIDI, MusicXML), and audio-visual data (e. g., video recordings taken from orchestral performances). Considering those various types of information, music data pose many challenges, since musical information is represented in diverse data formats. These formats, depending upon particular applications, differ fundamentally in their respective structure and content.

3.2.1 Term definitions

In the context of this thesis, the following definitions regarding digitized versions of music documents are used. A digital music document encodes a digital copy of either one or more sheets of music or an acoustic performance of a piece of music. A digital sheet music document



Freude, schoener Goetterfunken,
Tochter aus Elysium,
Wir betreten feuertrunken,
Himmlische dein Heiligtum.
Deine Zauber binden wieder,
Was die Mode streng geteilt;
Alle Menschen werden Brueder,
Wo dein sanfter Fluegel weilt.

Wem der grosse Wurf gelungen,
Eines Freundes Freund zu sein,
Wer ein holdes Weib errungen,
Mische seine Jubel ein!
Ja - wer auch nur eine Seele
Sein nennt auf dem Erdenrund!
Und wer's nie gekonnt, der stehle
Weinend sich aus diesem Bund!

Figure 3.1 Music document types typically available in music DLs. Adapted from [34].

consists of one or more digital images of scanned pages of a score book that is composed of text (e. g., table of contents, capitals), graphics (e. g., artwork of cover pages), and symbolic score notation (i. e., the actual musical content). A digital sheet music book consists of an ordered set of digital sheet music documents. A digital audio document consists of one or more CD-audio recordings of acoustic performances that embody the sound[1] made by concrete realizations of pieces of music. An audio CD contains an ordered set of digital audio documents. An audio CD box is an ordered set of audio CDs.

3.3 Music data and formats

In the context of this thesis, we concentrate on two widely used formats for representing musical data, as they explicitly address both the visual and the auditory modality. The sheet music format contains information on the notes such as musical onset time, pitch, duration, and further hints concerning dynamics and agogics. The purely physical audio format encodes the waveform of an audio signal as used for CD-audio recordings of acoustic performances. Being something in between those formats, the symbolic score format explicitly represents content-based information such as note onsets and pitches, but may also encode agogic and dynamic subtleties of some specific interpretation. While sheet music and audio data are available immediately in a digital format, symbolic score information is in most cases only indirectly available by priorly extracting the symbolic score information from sheet music. This can be performed automatically by using OMR software which is able to obtain the actual symbolic score content from sheet music scans. As the most frequently encountered

[1] More precisely, the sound pressure level in the air.

digitally available types of music data are scanned sheet music, symbolic score data, and audio recordings, most of the presented techniques and frameworks mainly focus on one or several of these data types.

3.3.1 Sheet music scan formats

A sheet music scan is an image scan of one or more pages or sheets of music. Each resulting scan of a particular page is stored as a digital image using an image file format. An image file format is a standardized mean of organizing and storing digital image data. The digital images are composed of digital data that represent rasterized versions of the respective scanned pages. Once rasterized, an image becomes a grid of pixels, each of which has a number of bits to represent a color value. An image file format may store data in uncompressed, compressed, or vector formats. There are a number of different types of image files. The most common are TIF files and JPG files, which are used by the music DL system. The way image data is compressed and stored is referred to as *image codec* which determines how the content is encoded. For example, JPG-formatted image data are encoded with the lossy JPEG codec. JPEG is the most common image codec or format used by digital cameras and other photographic image capture devices. In the context of the music DL system, the following formats are used. The uncompressed, lossless TIF format is used for storage and indexing purposes. The compressed, lossy JPG format is used for the downstream of image data to clients in order to save transmission bandwidth.

3.3.2 Symbolic music formats

A symbolic music format describes the contents of a piece of music symbolically by means of explicit musical information such as musical notes, instrument names, playing or conducting instructions, etc. Traditionally, it does not store sound—unlike the audio format. A sonification or acoustic rendition can be produced from this format by utilizing musical instruments. Here, musical notes and other information are used to (re-) create an acoustic performance by means of controlling electronic devices that synthesize sound. There is a wide range of symbolic music formats, see, e. g., [116] and [52], from which probably the most prevalent ones are MIDI [87], NIFF [57], and MusicXML [52]. Besides a countless number of existing symbolic music formats, some well-known examples of other symbolic music formats are Humdrum, LilyPond, and MRO. Basically, all of these symbolic music formats consist of explicit musical information. For example, the MIDI (Musical Instrument Digital Interface) format contains standardized control commands that can be read by MIDI-enabled instruments in order to shape sound.

The described symbolic score formats can be divided into two classes. We distinguish between *interpreted* score in a time-based (1-dimensional) representation and *uninterpreted* score in a spatial-based (2-dimensional) representation. These two format types fundamentally differ from each other, especially concerning the nature of dimensionality of score content in both number and type, and hence are not directly comparable. For a comparison, these must first be reconciled by reducing an uninterpreted, graphical score representation to a concrete, interpreted score representation or performance rendition, based on timely arranged musical events. In general, the various involved document types have to be reducible to a common representation. While the MIDI, Humdrum, and LilyPond formats belong to the first class, the MRO and MusicXML formats belong to the second class. Below, particular formats are discussed in more detail.

3.3.2.1 Event-based formats

The MIDI, Humdrum, and LilyPond formats organize musical notes as a set of events arranged in time. Hence, note timings are directly available. These formats can be understood as being (modifyable) concrete performance renditions that result from particular score interpretations.

The MIDI format MIDI [87] is an industry standard for controlling and communicating with a wide range of electronic musical instruments and other devices. The MIDI format encodes music as timed events, in which the most basic events are notes. The MIDI format is made based on a set of available instruments and specifies the notes to be played on those virtual instruments. The MIDI format stores instructions on how to recreate a performance based on synthesis with a MIDI synthesizer rather than an exact waveform to be reproduced, as it the case of audio recordings of performances. A single note determines several parameters, from which most essential ones are onset time, pitch, and offset time. Onset- and offset-times are specified in fractions of a measure. A pitch is specified by a standard representation of the Western musical scale. The MIDI format supports different channels, each of which can be assigned a different instrument. The channels may be played back sequentially or in parallel. Furthermore, the MIDI format supports polyphonic, i. e., simultaneous, notes to produce chords. Encoded music can be monophonic or polyphonic, dependent on the MIDI type, where MIDI type-0 is monophonic and MIDI type-1 is polyphonic.

3.3.2.2 The MRO format

The proprietary MRO [68] format is used by the commercially available OMR software package SharpEye Music Reader 2.68 to describe symbolic score extracted from underlying scans of sheet music.[2] While previously described formats explicitly encode note timings of scores, the MRO format does not. In contrast, this format encodes extracted score as it is and organizes musical notes and other objects as spatially arranged symbols. As it concerns the visual domain, it is in the nature of sheet music notation that time information is hidden in musical symbols that are to be interpreted. Here, score content is not interpreted, raising it to a higher abstraction level providing an additional degree of freedom w. r. t. performance realizations. Note timings are parameterized by note values, key signs, metrums, but also jump directives and other influencing information. That is, note timings are implicitly available through contextual interpretation of such parameters. For the cross-linking of MRO-formatted symbolic score to performance renditions such as MIDI files or audio recordings, the former needs to be transformed from a spatial- into a time-oriented representation, supported by an adequate format. The transformation of spatial MRO-formatted symbolic score data relies on semantic analysis and interpretation techniques. In the context of this thesis, techniques developed in [46] are utilized for the conversion process of MRO-formatted symbolic scores into MIDI-formatted symbolic scores.

3.3.3 Audio formats

An audio format is a medium for storing sound and music, and refers to both the physical recording media and the recording formats of the audio data. Generally referring to the physical

[2] In addition to MRO-formatted output, SharpEye also produces MIDI-, NIFF-, and MusicXML-formatted ouputs.

method used to store the data, a digitally stored audio recording is often equated with the audio file. An audio file format is a standardized means of organizing and storing digital audio recordings. Audio files are composed of sampled audio data that can be played on a computer. Once sampled, an audio recording becomes a series of samples, each of which has a number of bits to represent an amplitude value of the underlying audio signal. An audio file format may store data in uncompressed or compressed formats. There are a number of different types of audio files. The most common are WAV files and MP3 files, which are used by the music DL system. The way audio data is compressed and stored is referred to as *audio codec* which determines how the content is encoded. For example, MP3-formatted audio data are always encoded with the MPEG Layer-3 codec, while WAV-formatted files support selectable codecs, like, e. g., PCM, MPEG Layer-3, and many other codecs. Additionally, some audio file formats do not only contain the audio data, but also contain additional header information which can contain other information about the data. For example, MP3-formatted audio data can contain information about performing artists, title, and other metadata. The WAV and MP3 formats are further discussed below.

3.3.3.1 The WAV format

The WAV format is a standard audio format mostly found on Microsoft Windows platforms. It is commonly used for storing uncompressed PCM-encoded audio data. Hence, WAV files are large in size. For a CD-quality WAV file (44.1 kHz, 16 bits, 2 channels), it takes 1 411 200 bits per second or approximately 10 MB per minute. Note that WAV files can generally be encoded with a variety of codecs in order to reduce the file size (e. g., the GSM or MPEG Layer-3 codecs). The music DL system uses the WAV format for storing and indexing PCM-encoded audio data ripped from audio CDs.

3.3.3.2 The MP3 format

The MP3 format uses the MPEG Layer-3 codec and is the most popular format for compressing and storing audio data. By eliminating portions of audio data that are essentially inaudible, MP3 files are compressed to roughly a tenth of the size of an equivalent PCM-encoded file while maintaining good audio quality, assuming a constant bit rate of 128 kbps or approximately 1 MB per minute. The music DL system uses the MP3 format for the downstream of audio data to clients in order to save transmission bandwidth.

3.3.4 Hybrid formats

3.3.4.1 The MusicXML format

The MusicXML [52] format is an exception and takes a special position among the symbolic music formats. In comparison to other symbolic music formats, MusicXML can additionally store symbolic performances, audio recordings, and images. Moreover, also scores can be generated from this format.

3.3.4.2 IEEE 1599 music encoding and processing standard

Recently, a new standard, referred to as IEEE 1599, was published to encode and process music on various levels of representation [84]. Based on accepted common XML, it defines a standard language for a symbolic music representation. This format offers the possibility to combine entire information related to a musical work (audio performances, scores, MIDI, lyrics, images, annotations) in a single XML file. The standard also provides the possibility of adding structural information such as synchronization information [29] as well as music IR models [104] to the XML file. Using this standard, UIs for a holistic presentation of musical works using all information available were proposed [4].

At time of public announcement of this standard in the music IR community, the implementation of the music DL data model was completed. In favor of interoperability, it would certainly make sense to incorporate standardized music data models in the music DL framework in the future. Nevertheless, the impact and acceptance of IEEE 1599 as industry standard remains to be seen, as the follow-up revision of MusicXML, that has achieved wide acceptance in the past, is also targeting such features, and many applications incorporate that data format already.

3.4 Music modalities and cross-modal interrelations

The various representations mentioned above describe the same underlying actual musical content of a piece of music on various semantic levels. Thereby, they address different modalities, each of which has its own specific strengths and weaknesses. Sheet music contains musical symbols describing the musical content *visually* and abstracts from a concrete realization, i. e., interpretation and instrumentation as well. In contrast, an audio recording contains a once performed concrete realization of a piece of music, and describes the musical content *acoustically*. Note that both sheet music and audio recordings may be considered as two natural forms of music representation as they explicitly address the visual and auditory modalities, respectively. Accordingly, both acoustic and visual representations are most widely employed by users for accessing music. As those representation types address multiple modalities in an explicit manner, we call the document collections considered in the context of this thesis *multimodal*.

The availability of such multimodal document collections naturally leads to the necessity of providing tools to automatically process, analyze, and prepare this multimedia data for an efficient and user-friendly access. Equally, corresponding multimodal user interfaces for an adequate presentation and interaction with music documents are of high importance in order to provide an intuitive and convenient handling. A holistic, modality-fused presentation and interaction with music documents using as many different media sources and types as possible can significantly support the experience of music, as well as help to analyze music w. r. t. different aspects. For example, prospective conductors might be interested in listening to specific acoustic performances of pieces of music in order to learn or compare the conducting style of different conductors. For this purpose, an efficient and seamless comparison within and between interpretations is desirable by means of real-time smooth cross-fading. It turns out that the key challenge in designing such user interfaces and in suitably preprocessing the music documents is to find an appropriate common mid-level representation for both music modes originating from different domains in order to compare and (inter-) relate the musical content. In Chapter 5, it is discussed how reductions of the various document types to such a common representation is possible.

3.5 Work-centric organization of music documents and metadata

In order to appropriately organize real-life data stocks, several data models were reviewed in the context of the development of the music DL, from which the most fitting one for music documents seems to be the FRBR [64] model. However, as the FRBR model fits for common and entire documents, some crucial insufficiencies have been identified regarding the special domain of music. For this reason, a data model accounting for these insufficiencies was developed, which adequately reflects real-life circumstances, and sufficiently meets music-specific practice requirements. The music DL framework incorporates a work-centric organization of music documents, based on an adoption and appropriate adaptation of the FRBR model. In this work-centric organization, all music documents belonging to the same piece of music are grouped together in a hierarchical, multi-level relationship structure. In practice, a metadata work entry reconciles all notations and performances associated to that work. In joint work between the Multimedia Signal Processing working group and the BSB, an appropriate entity–relationship (ER) metadata schema [38] was developed. In contrast to the widely used OPAC or Machine-Readable Cataloging (MARC)[3] models, this data model offers a more complex, favorable description of music documents. A key benefit is that it takes into account the complex interconnections between various expressions of the same musical work and parts of it, and makes them explicit by cross-referencing, resulting in a work-centric view on the music documents. The data model proved to be particularly useful especially in the context of multimodal music access, ranging from cross-modal indexing to the cross- and multimodal retrieval.

3.5.1 Metadata

Metadata or meta information are high-level descriptive information about music documents. In general, metadata are used to describe global characteristics and properties of musical data. For example, in the case of an audio recording that encodes an acoustic performance of a piece of music, metadata include time-invariant information such as performing artists, title, conductor, instrumentation, etc.

3.5.2 Metadata catalogs

Library-provided metadata catalogs typically contain a metadata record for each piece of music. This may be a movement or song available as either sheets of music contained in a sheet music book or acoustic performances contained on one or more CDs. Each metadata record includes textual information such as the name of the creator, title, and a catalog name and number, e. g., “opus 3 no. 2”.

[3] The MARC format is an international standard cataloging format that was developed under the aegis of the LoC in the 1970s.

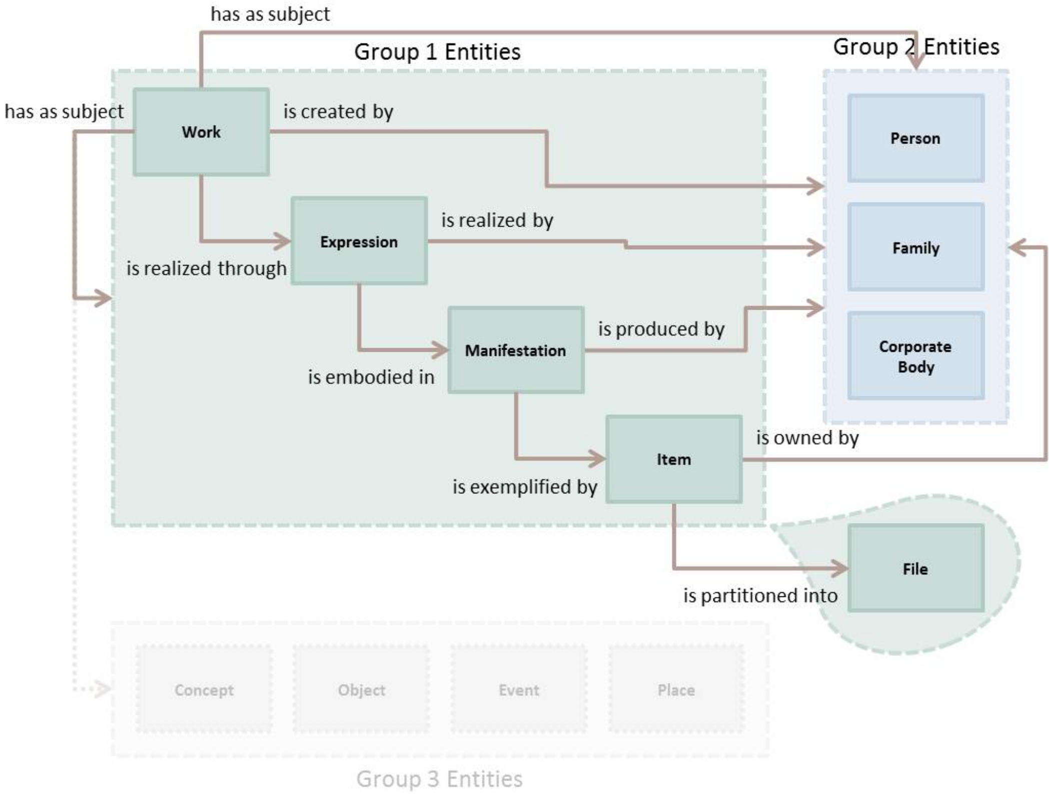


Figure 3.2 Overview on entities and relationships of the music DL data model, embedded in the FRBR model. The file entity does not belong to the original FRBR model.

3.5.3 Shortcomings of traditional library catalogs

Current library catalog systems pose many problems regarding the purpose of searching. One commonly cited shortcoming is the inability to find and collocate all versions of a distinct intellectual work that exist in a collection. Another is the inability to take into account known variations in titles and personal names [141]. Because catalog entries are organized in a non-hierarchical manner, related entries cannot be grouped together. One resulting consequence is that many entries are redundant. For example, for each particular audio recording belonging to the same piece of music, common information about that piece are multiply entered.

3.5.4 The FRBR model

To appropriately organize various representations of the same distinct intellectual work, the FRBR [64] model has been introduced in the library community. This model has been adapted within the music DL framework described in Chapter 4. FRBR is a standardized, widely used conceptual entity–relationship model for describing bibliographic documents or bibliographic metadata, developed by the IFLA. It attempts to address some of the shortcomings of traditional library catalogs by introducing the concept of multiple interrelated bibliographic entities [64]. It is built upon relationships between and among entities and reflects the hierarchical structure and interrelation of information resources[4] and other kinds of

[4] The term encompasses both physical and digital resources.

entities. An overview of the FRBR model (except for the file entity) is illustrated in Figure 3.2. "Relationships serve as the vehicle for depicting the link between one entity and another, and thus as the means of assisting the user to 'navigate' the universe that is represented in a bibliography, catalog, or bibliographic database" [64]. In particular, relationships between abstract intellectual works and the various published instances of those works are divided into a four-level hierarchy of works, expressions, manifestations, and items. The FRBR model represents a more holistic approach to the retrieval and access of entities, as the relationships between the entities provide links to navigate through the hierarchy. The FRBR model lays therefore the foundation for a new generation of cataloging systems that recognize the difference between a particular work (e. g., the bible), diverse expressions of that work (e. g., translations into English, German, French, and other languages), different manifestations of the same basic text (e. g., a particular paperback edition), and individual items (e. g., a copy of a paperback edition of the bible in a particular language on the shelf). In this formulation, each level in the hierarchy inherits information from the preceding level. Incidentally, this leads to reduction or even avoidance of redundancies.

The FRBR model comprises three groups of entities. Group 1 considers the following entities that describe intellectual and bibliographic units: (i) a *work* as a "distinct intellectual or artistic creation" [64] in its most abstract sense that only exists as an idea or imagination in the creator's mind and is intangible, (ii) an *expression* of a work as "the specific intellectual or artistic form that a work takes each time it is realized" [64] in physical or electronic form, e. g., an alpha-numeric, musical, or choreographic notation, a sound, an image, a video, a movement, and so on, or any combination of such forms, (iii) a *manifestation* of an expression as "the physical embodiment of an expression of a work. As an entity, manifestation represents all the physical objects that bear the same characteristics, in respect to both intellectual content and physical form" [64], and (iv) an *item* of a manifestation as "a single exemplar of a manifestation. The entity defined as item is a concrete entity" [64], i. e., a physical or electronic object that can be owned by a person. These entities build the foundation of the FRBR model. Alongside, Group 2 consists of the entities (v) *person* (as an individual) and (vi) *corporate body* (as an organization or group of individuals and/or organizations acting as a unit), and interconnects information resources with individuals and/or organizations. Group 3 includes the entities *concepts*, *objects*, *events*, and *places*.

The FRBR model promises to have a profound influence on the design of future catalog systems. The data model used in the context of the music DL framework (see Subsection 3.5.6) is on the one hand music-specific and hence somewhat restricted compared to the original FRBR model as described in [64], and considers only the entities (i) to (vi) from the first two groups. On the other hand, it extends the FRBR model for the special case of the music domain, where digital objects can be split in parts. Apart from this, it is fully compatible to the original FRBR model.

3.5.5 Shortcomings of the FRBR model from a musical perspective

The holistic approach of the FRBR model forms the basis of the data model used in the music DL framework. However, the requirements analysis identified three essential shortcomings regarding the specific case of digital music documents, especially concerning how pieces of music (expressions) are embedded within music documents (manifestations) found in practice. First, FRBR does neither consider a subdivision of a work into work parts such as individual

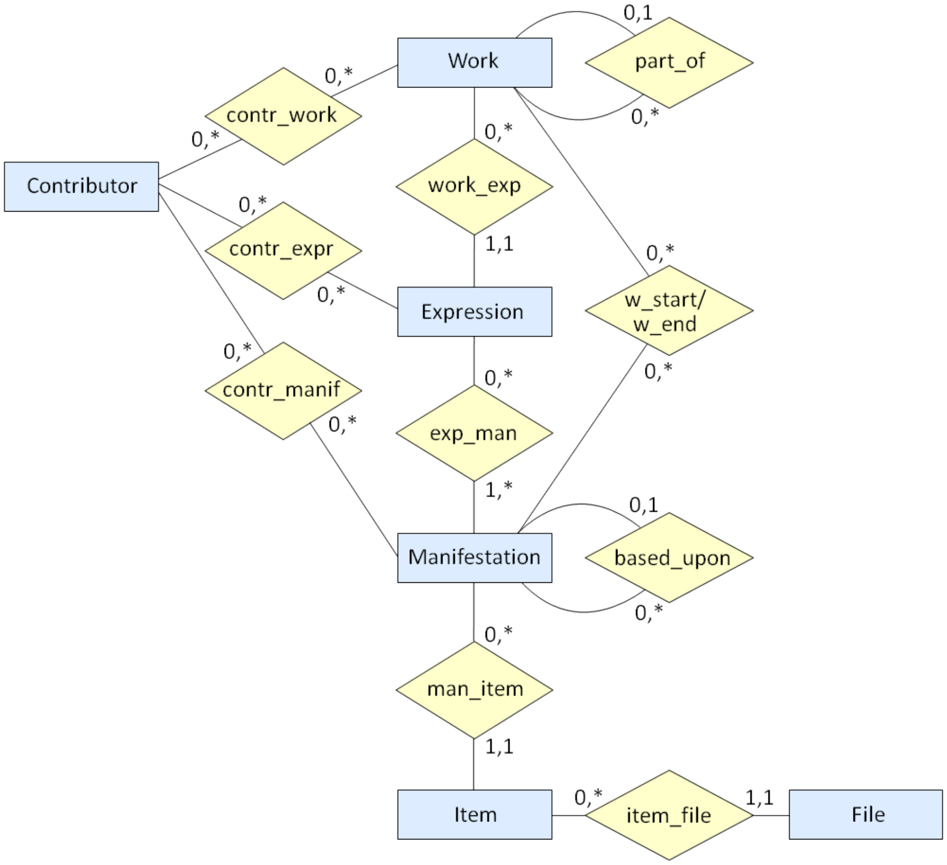


Figure 3.3 The ER diagram of the music DL data model. Adapted from [34].

movements, nor an aggregation of several works into one superordinate work. Second, FRBR does not account for an exact localization of individual expressions within manifestations—required segment boundaries corresponding to contained works are disregarded and hence cannot be referenced. In practice, sheet music books and audio CD boxes are usually composed of several different works that we want to be individually addressable. Third, FRBR lacks the concept of compound documents consisting of several self-contained parts or files. In practice, a digitization of, e. g., a sheet music book consists of the individual scanned pages of that book, each of which is stored as an image file. So, FRBR is insufficient in proper modeling digital music documents. To remedy those shortcomings, we propose to extend the model by one new entity and two new relationships. Furthermore, we restrict the entities to those ones being contained in the first two groups of the FRBR model. Entities of the third group are ignored in the music DL data model. An overview of the music DL data model and how it is embedded in the FRBR model is depicted in Figure 3.2.

3.5.6 The music DL data model

The ER diagram of the data model used in the context of the music DL framework is depicted in Figure 3.3. Here, entities are represented as rectangles and relationships are represented as diamonds. The ER diagram uses (min, max) notation of the Chen model [22], depicted at the

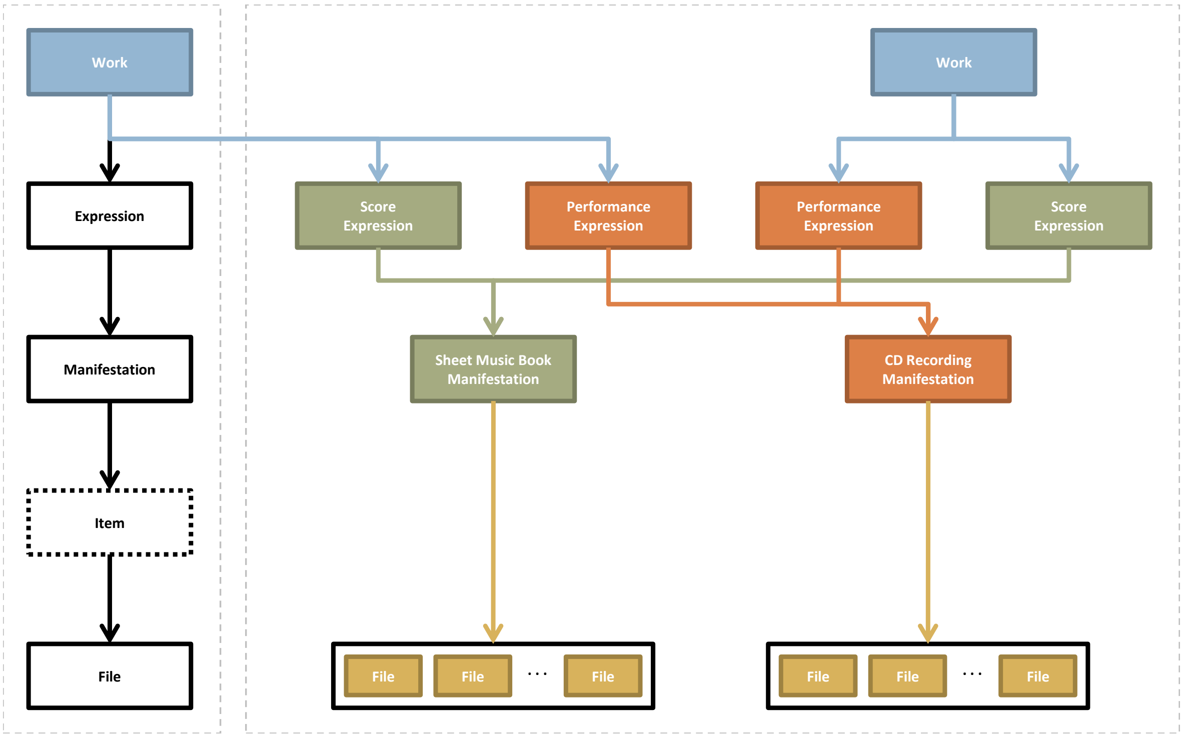


Figure 3.4 Conceptual music DL data model (left) and concrete example (right). The example illustrates the music DL data model-classification of involved files belonging to distinct pieces of music that are scattered over several manifestations.

connecting lines between rectangles and diamonds. This notation indicates the *cardinality*, which expresses the specific number of entity occurrences associated with one occurrence of the related entity. In this way, the occurrences of a relationship are specified, i. e., how many instances of an entity relate to one instance of another entity. More precisely, this notation captures both cardinality ratio and participation constraints, where each entity (rectangle) participates in at least *min* and at most *max* relationships (diamonds). To allow for an arbitrary number of relationships, an asterisk is assigned to *max*. For example, the $(0, *)$ cardinality on the work side of the work_exp relationship indicates that one work needs not to relate to any expression, but can relate to an arbitrary number of expressions. The $(1, 1)$ cardinality on the expression side of the work_exp relationship indicates that one expression relates to exactly one work, i. e., specifies a one-to-one relationship.

In addition to the original FRBR model, one new entity and two new relationships are introduced. Group 1 is complemented by the file entity. Relationships are supplemented by the part_of/whole relationship and the w_start/w_end relationship. In the following, this extension is discussed in detail from a musical point of view. In summary, the proposed reassimilation of the FRBR model ensures to meet the practical requirements for the targeted music DL w. r. t. the applicable organization of various types of music representations, their complex interrelations, and metadata [38].

3.5.6.1 Entities

In addition to the brief overview on the FRBR Group-1 and Group-2 entities described in Subsection 3.5.4, in the following a more detailed, music-oriented point of view on the entities used in the music DL metadata model is given, supplemented by examples. It is discussed how original FRBR entities are interpreted and adapted for the targeted scenario. All entity attributes are used for metadata indexing, searching, and browsing purposes. A concrete example of the music DL data model is depicted in Figure 3.4. Here, it can be seen that the typical real-life scenario is adequately adopted: score and performance expressions of two distinct pieces of music (works) are manifested as sheet music and CD-audio recordings, which are contained in a sheet music book as well as on an audio CD, respectively, each of which is embodied as a series of files.

Work A musical work represents the abstract composition of a piece of music, independent of any concrete notation or performance. It is purely a mental construct and resembles the intellectual creation behind that piece in the composer's mind. Every work is of a specific type. Examples for types are "piece of music", "movement", or "song cycle". Note that this is a proper generalization of the FRBR model without loosing any expressiveness. A work can consist of several subordinate works and, the other way around, several works can form a superordinate work. The relations between and among works and work parts form a complex dependency hierarchy. A work corresponding to a higher-level work that consists of several subordinate works such as a sonata, a concert, or a song cycle is referred to as root-level work. A work corresponding to a lower-level work being part of a superordinate work such as an individual movement or a single song, is referred to as leaf-level work. Examples for root-level works are Beethoven's "Symphony No. 5" or Schubert's "Winterreise". Examples for leaf-level works are the first movement of Beethoven's "Symphony No. 5" or the third song "Gefror'ne Thränen" of Schubert's "Winterreise". Each work is attributed with title, creation date, genre, type, and other distinguishing characteristics.

Expression A musical expression represents an intellectual or artistic realization of a particular work with a concrete arrangement in a defined form. Every expression has a specific form. In the context of this thesis, we concentrate on two separate forms of expression, score notations and acoustic performances. Score notations are the result of a concrete arrangement of a musical composition in a written form. Acoustic performances are the result of a coordinated sound production of an underlying work, performed by musical instruments. Note that each expression (transcription and performance, respectively) is unique and different from each other. An example for an expression is a score notation of the first movement of Beethoven's "Symphony No. 5" produced by Henle Verlag. An acoustic performance of that movement, performed by an orchestra conducted by Karajan, is another example of an expression.

In contrast to a work that refers to a distinctive composition, an expression refers to its arrangement, orchestration, or instrumentation. Two expressions are of the same work if they differ in their arrangement, key, or form, but share the same distinctive basic features of that work to a large extent, so that the work can be unambiguously recognized as such. This is the case for, e. g., different orchestration, instrumentation, or varying occurrences and arrangement of measures. In contrast, if an arrangement involves substantial artistic or creative contribution so that the piece of music cannot longer be clearly recognized, it refers to a newly created composition and thus a different work.

Each expression is attributed with contributors along with their roles, title, form, key, instrumentation, date and place of performance or notation, and other distinguishing characteristics. By means of using those attributes in combination, it could be easily figured out, e. g., in what performances a specified artist participated in as a singer during the time period from 1970 to 1980.

Manifestation A musical manifestation represents the physical embodiment of one or more expressions. Once a particular expression is captured in some container or its content is recorded on some carrier, it becomes a manifestation. Note that a manifestation can be a compilation of several different works, and additionally it can be partitioned into multiple separate volumes. This music-related, practical property is taken into account in the design of the music DL data model.

In order to record something, it has to be put on or in some container or carrier. Score notations are transcribed and manifested as sheets of music residing in sheet music books. Acoustic performances are recorded and manifested as tracks on CDs. So, manifestations appear in various carriers. As a whole, a manifestation bundles all the pieces of music that have the same physical characteristics and content, and are the result of an expression. Note that any change in the physical characteristics results in a new manifestation. Reduced to the considered expression forms and carriers, a manifestation represents a prototype (master copy) of a particular edition of either a sheet music book or a CD box. In actuality, a manifestation is itself an abstract entity, but describes and represents physical entities, i. e., all the items that have the same content and carrier. A bibliographic record typically represents a manifestation as the entirety of a publication. An example of a manifestation is a published recording of a CD album, which consists of many pieces of music (tracks), which in turn contain an audio signal. Another example of a manifestation is a print version of a sheet music book.

As in our scenario we deal not with physical but exclusively with digital music documents, a manifestation corresponds to a digital version of an item (described below)—a physical copy of a particular manifestation. Digital versions of music documents are created by digitization. The resulting digital copies are represented as binary data streams with each of them split across multiple self-contained parts or files (described below). A digital copy of a sheet music book is composed of the individual scanned pages, each of which is stored in an image file. A digital copy of a CD box comprises the individual CD tracks, each of which is stored in an audio file. Note that in contrast to a physical copy of a manifestation, which refers to a new item of that manifestation, every digitization of a manifestation constitutes a new manifestation, as well as every derivative made of it at a different sampling rate, format, and encoding.

Each manifestation is attributed with title, edition, publication date and place, publisher, form and dimensions of carrier, capture details, access restrictions, and other distinguishing characteristics.

Item An item represents an individual physical copy of a particular manifestation. It carries the actual content of a manifestation on a medium, which may be of physical but also immaterial form such as a bit string encoding a digital document. An item would be, e. g., an individual copy of a CD box, a printed sheet music book, or a digitized version of the latter comprising a set of files. Since in the context of the music DL framework it is dealt exclusively with digital instead of physical copies of music documents, all “physical copies”, i. e., items, of a manifestation are replications that are inherently identical and undistinguishable from each other and the original manifestation itself. So, the item entity is redundant and can be dispensed. For compatibility reasons, the item entity is nevertheless retained; technically, it just points to the manifestation entity. In the following we therefore work directly with manifestations and make no factual distinction between items and manifestations.

File A file represents a self-contained part of a particular item or manifestation. The introduction of this entity accounts for the fact that the concept of an item or manifestation as atomic unit is insufficient for modeling digital documents. As opposed to physical media, digital music documents are generally compound objects that comprise multiple self-contained parts. In practice, these parts are stored in files residing on a file system. Each file is attributed with order number in set, format type, codec, and a reference to the corresponding file resource. For example, a sheet music book consists of an ordered set or chain of image files in the TIF or JPG format, each of which encodes a scanned book page. A CD box consists of a chain of audio files in the WAV or MP3 format, each of which encodes a ripped CD track.

Contributor A contributor represents an individual, an organization or group of individuals, or organizations acting as a unit. This entity relates musical content with individuals or organizations. Each contributor is attributed with a role such as composer, performer, singer, conductor, and publisher.

3.5.6.2 Relationships

In the following, the two newly introduced, musically motivated relationships used in the music DL data model are described.

part_of/whole The part_of/whole relationship specifies the dependency relationship between (partial) works, i. e., what work is part of what other work. In this way, the possibly hierarchical structure of a work can be modeled appropriately. Examples are individual pieces of music belonging to a song cycle or individual movements making up an entire piano sonata. This relationship hence allows to model structural information of works by both partitioning a higher-level work into lower-level works such as individual movements, as well as grouping together several lower-level works being parts of a higher-level work. For example, the song cycle “Winterreise” by Franz Schubert is the superordinate work of the individual associated pieces of music. From a technical point of view, with the help of this relationship, the work entity is extended to not only include entire works but also parts and unions thereof.

w_start/w_end The w_start/w_end relationship specifies the start and end positions of a particular region in a manifestation, over which an expression spans. With the help of this relationship, a finer subdivision of a multi-expression manifestation into its individual expressions is possible, allowing for an exact localization of the boundaries at which individual pieces of music begin and end. In case of sheet music, a position indicates a page number, a grand staff line number, and a measure number. In case of audio recordings, a position indicates a CD number, a track number, and a time position. The introduction of this relationship overcomes the limitation of the FRBR model, wherein a precise localization of individual expressions contained in a manifestation is not possible. Note that this relationship also provides the basis for structural metadata on the level of manifestations. By means of these structural metadata, a navigation within manifestations is possible. Moreover, also a navigation across both different manifestations and manifestation forms is possible by interrelating all manifested expressions of all works.

exp_man An exp_man relationship specifies a captured expression that resides on one or more manifestations. In actuality, exp_man relates to a single-expression manifestation that may be part of a larger multi-expression manifestation. Each exp_man relationship is attributed with order number of the expression in a manifestation.

Chapter 4

Framework for a music digital library system

The availability of digital music collections inherently leads to the necessity of providing tools to automatically process, analyze and prepare these multimodal data for efficient and user-friendly dissemination. This chapter bridges the gap from theory to practice for the construction of a real-life music DL.

The main goal is the development of practical solutions for the preprocessing and dissemination of music data on a large scale that are integrated into a comprehensive framework. Thereby, a whole range of challenges arise from practical demands that have to be tackled in real life. Critical requirements are especially maximization of the degree of automation on the one hand, as well as most natural interaction with music data on the other hand. It turns out that suitable and systematic combination of novel and state-of-the-art technologies in a coordinated fashion leads to functional and sophisticated music DL services that offer efficient and intuitive access to music data in a work-centric and multimodal way, with a bunch of unprecedented possibilities concerning interaction modes. All of these services are utilized by means of a standard Internet browser, which constitutes a very modest demand on end-user's computing environments.

The following five main subjects have to be addressed to design a DL: conceptual system design and architecture; data acquisition, processing, and presentation. Accordingly, the present chapter is structured as follows. Section 4.1 introduces the reader to the initial situation, goals, tasks, and challenges that arise from practical demands, and outlines a proposed solution. Section 4.2 concerns the acquisition of music data. In Section 4.3, concepts and methods for a consistent preprocessing of music data are discussed. Section 4.4 presents resulting novel interaction modes with music data, accounting for a holistic music data model. In Section 4.5, a blueprint of the reference system architecture for delivering of a concrete music service is illustrated. Here, a mid-level description of all key components of the DL system is given, focusing especially on interrelations and mutual dependencies.

4.1 Outline

The framework describes the “big picture” encompassing goals, objectives, challenges, and practical solutions for a generic music DL incorporating the outlined state-of-the-art music IR techniques. Pursuing a holistic approach, this chapter discusses incorporated technologies, develops methods and systems, and presents a system architecture for the delivery of a music DL service. Thereby, all important aspects for the successful delivery of such a service are addressed by the framework, from which the main objectives are acquisition, process management, dissemination, as well as novel interaction modes and mechanisms with music documents. The approached solutions incorporate interdisciplinary knowledge from several scientific and software engineering areas, including music IR, librarianship, DL, enterprise architecture, business processes, and human–computer interaction. The music DL system is based on a SOA and uses Web services.

4.1.1 Application scenario and initial situation

In the following, a brief summarization is given on which types of music documents are available and what is assumed to hold in the considered real-life library scenario. In a practical real-life scenario, the most prominent music documents that are available in libraries are Western classical music documents in the form of sheet music books and either individual audio CDs or collections thereof, referred to as CD boxes. In addition, metadata are available for entire books and audio CDs, stored in electronic catalogs or databases.

Through the ongoing mass digitization that is taking place at various libraries during the recent years, a steadily growing volume of electronic versions or digital copies of the above music documents are available. As indicated in Figure 3.1, the resulting digital copies comprise large amounts of scanned sheet music pages and ripped audio CD tracks. In addition to traditional, library-hosted metadata catalogs that provide metadata on the level of books or audio CDs as a whole, metadata on the level of both CDs and individual CD tracks are available from external sources. Consequently, these are utilized for the supplement of metadata of audio recordings. They are gained from remote databases over the Internet.

4.1.2 Goals

Considering the background discussed in Subsection 4.1.1, the primary goal of this thesis is the creation of a fully operative music DL service delivering management and dissemination of music documents, that can be seamlessly integrated into preexisting utilization chains of current libraries. To gain practical insights from the librarians’ side, several practically relevant technical requirements and user demands have to be considered. The music DL service targets on the real-life library or commonly content provider scenario and incorporates the storage and access to digital music. For preservation purposes, digital copies of available music documents held by content providers such as audio recordings, sheet music, and other music-related materials are made. For indexing purposes, these digital copies are to be analyzed, annotated, grouped together, and semantically cross-linked based on their content and metadata using state-of-the-art music IR techniques (see Chapter 5). For dissemination purposes, the indexed data is to be searched, retrieved, browsed, navigated, and presented using services provided by a reference system architecture (see Section 4.5).

4.1.2.1 Assumptions and constraints

In the context of this thesis, we assume that music documents are of Western classical music, where sheet music books are notated in the Western standard notation system and are available as a print version (i. e., not in a handwritten form). Note that, however, also Western popular music documents can be successfully integrated, as first experiments showed. As a baseline scenario for a realistic system, it is assumed that no additional metadata are provided directly by content providers. However, for audio CDs and tracks, additional metadata is acquired through the Gracenote [56] service.

4.1.3 Approach

In order to achieve the stated goals, an analysis of the initial workflow within the targeted scenario had to be conducted to deduce the functional requirements for the music DL framework and its operation. As for planning purposes, problems that arise in a librarian's particular day-to-day business had to be identified through interaction with the partner on the library side (BSB). As for the entire music DL system construction, initial efforts focused on identifying practical demands of what a user expects from a music DL. Finally, knowledge transfer to determine what is possible using state-of-the-art music IR techniques had to be performed and comparisons had to be drawn between available technologies und practical requirements.

It is now discussed how the pursued objectives have been implemented by utilizing and applying concepts and technologies from the fields of software engineering and music IR.

4.1.3.1 Functional requirements

After a requirements analysis based on involving actual users (librarians and library customers) at the BSB, the following requirements were considered to be essential for a music DL:

- *Consistent concept of exploitation of cross- and multimodality.* As music documents are available as different media types, the cross- and multimodal collocation as well as interconnection of documents is to be established, using state-of-the-art music IR techniques. Moreover, a consistent cross- and multimodal querying-retrieval concept is to be incorporated.

- *Semi-automatic indexing of digital music documents with the lowest possible degree of required manual effort.* For this purpose, maximization of the degree of automation is targeted regarding preprocessing and ingestion of documents and associated metadata into the music DL repository. Because of quality assurance reasons, it turns out that a reduced degree of automation has to be accepted.

- *Conceptual design and development of an administrative workflow or preprocessing chain as a guidance of ingesting music documents into the repository with—by current state of scientific knowledge—the lowest possible degree of human–computer interaction.* The minimization of manual effort needed is to be derived from insights gained from practical requirements analysis.

- *Deployment of administration and client UI components for the intuitively operated management and user interaction with musical content utilizing a work-centric view* *on music documents*. Adequately designed user interfaces have to support modality-fused presentation and interaction with music documents to improve overall cognitive perception and experience of music.

- *Deployment of services for the querying, retrieval, and dissemination of indexed musical content, operated through a SOA supporting a SOAP interface-based public online access*. Musical content is to be downstreamed to clients.

4.1.3.2 Real-world challenges and applied music IR techniques

Music DLs pose several interesting challenges for DL research. For the purpose of an advanced presentation and interaction with music documents as discussed in Chapter 3, the delivery of tools that are capable of indexing, managing, distributing, and utilizing the documents is required. Indexing tools incorporating state-of-the-art music IR techniques such as content-based indexing and cross-linking of music documents are needed for preprocessing purposes. In addition, a workflow for document processing and adequate administrative user interfaces must be provided for management purposes such as creation and maintenance of content. In the context of this thesis, the term content refers to primary and secondary data, as well as to associated metadata. For the delivery of content, a service must exist that is able to distribute the content of a music DL repository. Here, the term repository refers to the entirety of archiving and server hardware providing access mechanisms for content distribution (cf. data layer and server layer in Subsection 4.5.2.1). Finally, user interfaces must be provided for content access and utilization, allowing modern and feature-rich handling of content (cf. presentation layer in Subsection 4.5.2.1). Subsequently, the real-world challenges for constructing the targeted music DL framework are discussed.

Discovering semantic interrelations on different levels One key task is to semantically cross-link all available documents for the same piece of music and relate them among each other. At this time, particular available document types are audio recordings and scanned sheet music. On different granularity levels, mapping and synchronization techniques are used to create semantic cross-references and alignments between meaningful entities within

- sheet music books and audio CDs (referred to as movement–track mapping),
- sheet music pages and time segments within audio recordings (referred to as score–audio synchronization),
- words and time segments of audio recordings (referred to as lyrics–audio synchronization), and
- time segments of different audio recordings of the same piece of music (referred to as audio–audio synchronization).

The movement–track mapping, i. e., the coarse-level identification and cross-linking of—in a musical sense—semantically equivalent document parts scattered over various documents, is a crucial sub-task. Not only is this necessary to gather all manifested expressions that belong to the same musical work in order to collocate them according to the music DL data model. It primarily serves as a prerequisite step for subsequent synchronizations on finer levels. In the context of this thesis, the following types of synchronization are used. The score–audio synchronization builds linking structures between sheet music and audio representations of the same work. Here, individual measures within sheets of music are mapped to corresponding time segments within audio recordings. The lyrics–audio synchronization builds linking structures between extracted vocal parts of sheet music and audio representations of the same work. For this sake, it maps individual syllables and words of the lyrics to corresponding time segments. The audio–audio synchronization builds linking structures between different acoustic performances of the same work. It links short time segments from one audio recording to corresponding time segments of another audio recording.

A more detailed view on the topic of extracting meaningful entities from scanned sheet music and its mapping to audio recordings is given in [62, 6, 51].

Cross- and multimodal indexing Another key task is to build up content-based search indexes in order to search for, e. g., lyrics phrases, melodies, score excerpts, and audio fragments. First, these are used for ordinary unimodal content-based searches. Second, in combination with the discovery of semantic cross-references, search and navigation capabilities across modalities become possible. Third, as a consequence of a unified presentation of musical works, also multimodal queries that are composed of individual unimodal queries are supported and offered to the user.

4.1.3.3 Working steps

From a task-oriented point of view both functional and architectural requirements have to be taken into account in order to realize the targeted functionalities. To realize a music repository providing functionalities described in Subsection 4.1.3.1 and deliver features described in Subsection 4.1.4.8, the following tasks have been tackled:

- *Construction of a database for metadata, referred to as metabase, incorporating the work-centric approach of the music DL data model.* Database records are adopted from both MAB catalog records and Gracenote-formatted files. Those are made available by the document management system (DMS) as suggestions to librarian operators with an option of manual correction. For this purpose, metadata from content provider-hosted metadata catalogs are extracted and assigned to the metabase scheme that fully fits the work-centric approach of the music DL data model. It turns out that utilized techniques are unreliable and thus the conversion process requires manual revision.

- *Work-centric organization of music documents.* To achieve an automated work-centric collocation of all the music documents or parts thereof belonging to the same work, the incorporation and extension of the FRBR model was approached. In this context, a content-based analysis of music documents prior to identification of corresponding parts within other documents embodying the same content in a musical sense is needed. For this purpose, common and comparable mid-level representations of music documents are needed. In particular, a mid-level representation based on so-called chroma features are used. It turns out that utilized techniques are fairly reliable, dependent on the accuracy of recognition and extraction of the symbolic content (notes, lyrics, instructions, etc.) of underlying sheet music documents (yielded by OMR processing). However, for reasons of quality assurance and improvement, recognition results should be optionally revisable by hand.

- *Semi-automatic macro-level indexing of music documents with the requirement of manual intervention for controlling and correcting erroneous and deficient information, partially extracted from potentially unreliable sources.* The identification and mapping of semantically interrelated parts of music documents within the same and across different types and modalities is needed in order to find semantically equivalent tracks and sections (cf. Section 4.1.3.2). It turns out that utilized techniques are slightly unreliable. Since this is a very basic processing step with a high impact on the overall preprocessing result quality, a mandatory by-hand revision of the processing result is recommended.

- *Fully automated micro-level indexing with a non-mandatory option of correcting potentially erroneously derived data in order to further improve quality (cf. Section 4.1.3.2).* For this purpose, alignment of semantically equivalent tracks or sections of music documents within the same and across different types and modalities, regardless of the representation type, are calculated by means of music synchronization techniques. It turns out that utilized techniques are reliable, but occasionally some subsequences are misaligned. To further improve the accuracy of calculated alignments, the latter should be optionally revisable by hand.

- *Establishment of a document processing chain that minimizes required manual effort for (a) the addition and revision of new digital music documents and associated metadata records to the repository, (b) the cross-modal indexing and multimodal fusion of them, and (c) the preparation of them for dissemination, i. e., the delivery to clients.*

- *Development of a GUI-based administrative control and intervention tool relating to each link of the document processing chain in order to support access to primary data as well as revision and correction of derived data in all stages of the preprocessing; e. g., for the purpose of later-on corrections of recognition results such as key signature detection and repeat detection in scores.*

- *Content-based indexing of all kinds of representations to allow for fast, content-based, and cross-modal searching capabilities (cf. Section 4.1.3.2).*

Discussion One key challenge is the reliable segmentation of sheet music books into individual pieces of music or movements. The segmentation especially relies on the detection of the musical form of individual pieces of music. The musical form refers to the overall structure of the piece, including the correct identification of repetitions, which poses a non-trivial problem. In conjunction with the musical synchronization of sheet music and audio performances expressing the same piece of music, the treatment of differences regarding the musical structure is another key challenge. The detection of the musical form of both sheet music and audio performance representations is important in order to match and align semantically equally parts within a particular manifestation. The first objective is to automatically detect repetitions and other kinds of structural information of sheet music notation in order to extract the course or progression sequence of underlying score. It turns out that the most critical issue is to automatically extract the correct score sequence from sheet music. An intrinsic property of sheet music notation is that the score sequence is compactly represented by means of marking repeated sections of notated score with the help of repeat signs, Dal segno, or Da Capo instructions. Therefore, the recognition of the latter is crucial to gain an "unfolded" representation of the score, that can be aligned to an acoustic performance. The second objective is to detect the structure of corresponding acoustic performances. Moreover, as repeat signs indicate that sections *should* be repeated, a concrete realization of the score may differ from the originally intended sequence, resulting in structural differences even in the case that score sequences are correctly extracted from sheet music. In order to handle such inconsistencies regarding different musical structures extracted from sheets of music and a corresponding audio performance, JumpDTW [46] is used. This is a crucial stage in the document processing chain, as structural inconsistencies have to be resolved in order to achieve a high accuracy and quality of the further steps of identification and synchronization. Therefore, the identification result, in particular the detection of start and end points of tracks or segments, can be corrected before the step of synchronization is performed.

Note that, in principle, the latter could also be fully automated without human intervention. Evaluations show this to achieve a maximum reliability, i. e., correct mappings, of nearly 80 % [46]. While on the one hand this degree of correctness may suffice for the processing of Web-scale data sets comprising millions of documents in favor of full automation, it does not suffice for the library context for quality assurance reasons. Libraries are responsible for the maintenance of high quality standards, thus the revision of the results is mandatory.

The extraction of musical symbols and lyrics from sheets of music is performed through OMR processing. As the OMR process does not work flawlessly, the recognition result, i. e., the extracted symbolic score and text siblings, is potentially error prone and should be corrected in order to improve the quality of further processing steps. The insufficient detection accuracy of the OMR process necessitates self-correcting post-processing steps, which also constitutes a key challenge. Techniques for the automatic detection and correction of some types of errors are proposed in [46], where solutions to these kinds of challenges are discussed.

To achieve the goals stated in Subsection 4.1.2, individual music IR techniques are exploited, suitably combined, and applied in an appropriate way. The utilized core techniques and concrete algorithmic tools for the semi-automatic processing that implement the general core technologies, are summarized as follows. For the recognition and extraction of meaningful entities within and across music documents, OMR, OCR, and audio signal processing techniques are used. For the creation of synchronization structures or alignments within and across music documents, dynamic time warping (DTW)-based techniques [70], more specifically MsDTW or FastDTW [94], and JumpDTW [46], are used. For the indexing of music documents, inverted file indexes [3], diagonal matching, and audio matching [79, 77] techniques are used. To allow for content-based search for music documents by means of specifying short query fragments of various forms such as melodies, score excerpts, lyrics, and audio snippets, several indexes are built up in a fully automated way. More precisely, the indexes are constructed from feature sequences extracted from the documents.

4.1.4 Contribution

The framework presented in this thesis increases the usability of services provided by libraries or other content providers. The employment of the framework enables content providers to deliver a value-added music service providing novel and feature-rich dissemination of digital music documents. The development of the framework was guided by insights gained from practice in the context of a cooperation between the Multimedia Signal Processing working group at the University of Bonn[1] and the BSB Munich. The framework is designed with the goal of being an integral part of an existing document processing chain of a content provider. It consists of several preprocessing tools incorporating modern music IR techniques for the ingestion and management of multimedia music document collections, a SOA-based system providing a prototypical implementation of a Web service, as well as various UIs for multimodal querying, retrieval, navigation, and presentation of music, with the primary goal of delivering value-added services.

A work-centric consideration of available music documents is taken into account, where particularly interrelations between various types of music documents are exploited. It turns out that, using state-of-the-art music IR techniques, those interrelations can be adequately cross-referenced, resulting in a modality-fused collocation of different document types in an integrated manner. This contributes to an increased convergence of music documents of different modes. Thereby, novel exploration and interaction modes with music documents can be achieved in order to increase overall music experience. For the purpose of an adequate indexing of multimedia music collections, a semi-automatic document processing chain is proposed that is controlled by means of a set of administrative tools, guided by a workflow which is gained from practical insights. The entirety of procedures, applied techniques, and tools is incorporated in a central management tool that is used to generate indexing and synchronization structures, and other kinds of secondary data.

4.1.4.1 Complete framework

Altogether, the framework provides concepts, methods, systems, and tools for the delivery of music DL services for the creation, management, and access of music repositories. For this, music documents are acquired, processed, indexed, and prepared for dissemination by means of a multi-stage document processing chain incorporating state-of-the-art music IR techniques. Musical content is managed by a workflow that relates to the document processing chain and controls the execution sequence by means of an administration user interface.

4.1.4.2 Consistent incorporation of cross- and multimodality

A key contribution is the consistent, large-scale indexing, cross-linking, and alignment of digital music documents stored in a music DL for a unified and multimodal access. In this context, the focus particularly concentrates on the cross- and multimodal querying, retrieval, presentation, and navigation of spatio-temporally linked sheet music books and corresponding audio CDs or CD tracks. This allows for novel and feature-rich applications that deliver new experiences in interacting with musical content. This is made possible by several music IR techniques described in Chapter 5. For example, by means of music alignment and advanced retrieval technqiues, music can be searched based on multiple modalities which also can be combined in order to improve retrieval performance. Another example is the score–audio synchronization, where retrieved original sheet music scans and semantically corresponding audio recordings are presented to the user in a synchronized, multimodal way (cf. Figure 6.4).

[1] `http://www-mmdb.iai.uni-bonn.de/`

4.1.4.3 Information system

Another key contribution is the development of an information system (IS). As generally ISs play a more and more important role in our society, the demands on these systems have also increased. Departing from their traditional role as simple repositories of data, ISs must nowadays provide more sophisticated support and incorporation of modern indexing and dissemination technologies. Therefore, the approached music DL system does not only provide simple access to musical content by means of viewing or listening to music documents in a unimodal way (i. e., unimodal streaming or download of a document). Moreover, the system offers intuitively operable, advanced access and interaction possibilities to content, detached from physical constraints. This is achieved by a sophisticated retrieval concept and work-centric presentation of music documents using all available modalities. In particular, the system provides users with tools that allow for the formulation of metadata- and content-based search queries that can also be combined in order to express search queries more precisely. This approach has been shown to significantly improve the retrieval precision. The system supports a unified access to retrieved documents that focuses on pieces of music. Here, all available music documents belonging to a particular piece are retrieved and presented in an integrated and interlocked fashion, delivering a new user experience regarding consumption and interaction with music.

4.1.4.4 Document processing and workflow

Regarding required preprocessing techniques, main contributions are a highly automated document processing chain, a workflow, and a GUI-based DMS. The semi-automatic document processing chain for cross-modal indexing and organization of music documents is controlled by a well-defined workflow (requiring manual steps) using the highest possible degree of automation. Exploiting and combining state-of-the-art music IR techniques, a highly automated document preprocessing can be achieved. The preprocessing is guided by the workflow, developed on the basis of insights gained from an evaluation of the processes at todays libraries. The objective is an automated document indexing and warehousing, controlled, supervised, corrected, and prepared for dissemination by a content-managing person. The workflow is applied using the DMS, with which repository content is managed. The intuitively operated tool is designed to support especially non-technical content-managing persons in their daily business for streamlined operating cycles.

4.1.4.5 Core capabilities of the music DL system

The music DL system comprises a set of automated subsystems that together provide a comprehensive capability to manage and disseminate digital copies of music documents held by content providers. Specifically, the capabilities of the music DL system concerns the following areas: capture or creation of content, content-based indexing, cross-linking, and metadata cataloging, storage and Web service-based distribution, as well as work-centric and multimodal search and presentation of content. Musical content and metadata exist in multiple formats and on different types of media, each of which with specific technical challenges regarding indexing, processing, and dissemination. On several levels, the music DL system architecture (see Section 4.5) shows how capabilities are realized and related. It describes how business processes or functions are realized, how technology components fit together, and how they interact with each other. Those functions, components, and inter-relationships are reduced to a concrete software and hardware reference implementation, which leads to an operational, prototypical music DL system.

4.1.4.6 Semi-automatic acquisition and assignment of metadata

Yet another contribution is the integrated support of external metadata sources for the acquisition of metadata associated to audio recordings on both CD- and track-level, and the assignment in the music DL data model. Metadata are acquired from Gracenote-formatted CD metadata that are gathered from the Gracenote service.

4.1.4.7 Highly automated indexing of music documents

On the backend, the framework provides a GUI-based DMS that automates the process of ingesting and maintaining music documents as much as possible by means of state-of-the-art music IR techniques. For the purpose of cataloging and indexing, each document is passed through a document processing chain which consists of several stages. Each of these stages performs particular music IR techniques which collectively recognize, extract, interrelate, and align meaningful semantics obtained from the documents. However, at particular stages manual intervention is possible or necessary. For a streamlined document management, an administrative workflow has been established that guides the course of manual interactions with the DMS. This workflow is gained from practical insights and is optimized to administer operating tasks efficiently. A more detailed view on document preprocessing is given in Section 4.3.

4.1.4.8 Sophisticated user interaction

The resulting benefits on the frontend are advanced capabilities for cross-modality searching, browsing, presenting, and navigating in audio-visual music content. Here, all individual modalities are mutually interlocked such that each content-interaction in one modality is equivalently reflected in the other modalities. In particular, the following functionalities are available for end-users:

- Cross- and multimodal retrieval of music content. Both metadata- and content-based retrieval of music content within the same and across different modalities is supported, based on the consistent integration of the aspect of multimodality in every stage of the querying–retrieval process.

- Time-interleaved playback of music documents of the same representation type, allowing for interactive crossfading between different performances of a particular piece of music during playback. For example, the user can set the "listening focus" to one of those performances and jump at any time to another one while maintaining the musical position. This can be used for drawing local comparisons between different interpretations.

- Spatio-temporal synchronous playback of music documents of different representation types belonging to the same piece of music. This includes:

  - Multimodal presentation of audio recordings and associated sheet music representations, respectively, where the visual modality is used to highlight the currently played part of the audio recording. This can be used for the score-following purposes while listening to associated performances of a piece of music.
  - Multimodal presentation of audio recording and associated lyrics, respectively, where the visual modality is used to highlight the currently sung syllables and words of the lyrics.

- Score-based navigation in performances with the capability to resume playback at arbitrary positions. Here, individual positions in performances are identified through the more suitable visual modality, where positions are expressed in music and not in time units.

4.2 Acquisition of music documents

In the course of establishing a music DL, the first consideration to be made is the acquisition of musical content. This section presents available sources of primary data and associated metadata, which constitute the baseline of building up a music DL.

4.2.1 Data sources

4.2.1.1 Content provider-hosted data

Being a large-scale library, the BSB holds a huge amount of music documents, ranging from handwritten and printed sheets of music over acoustic performances recorded on several media types up to biographies and other accompanying music-related materials. In the context of the PROBADO digital library initiative, the framework was developed focusing particularly on practical conditions in such a large-scale content provider. Because of their size, the development of streamlined processes is a demanding issue. Fundamental considerations on the real scenario at the BSB constitute the basis for learning from business practice, and the development of practically applicable concepts.

4.2.1.2 Public-domain data

Most of the music documents hosted by content providers cannot be publicly made available due to copyright infringements. For this reason, in order to properly demonstrate the concepts of a developed music DL, also non-copyright protected music documents were acquired from the Internet and other sources to create a music repository that is driven by a legal, publicly accessible music service[2], hosted at the University of Bonn.

4.2.2 Primary data resources

Considered primary data resources are scans of printed sheet music books and rips of audio CD boxes. In addition, associated metadata records stored in catalogs are considered. Beyond

[2] http://probado.iai.uni-bonn.de:8080/

those, another group of data resources that may be available and particularly interesting for music research consists of accompanying information such as biographies. However, such data are out of the scope of this thesis and not being considered. In the following, considered primary data resources are described in more detail.

4.2.2.1 Scans of sheet music books

A scanned sheet music book is an ordered set of image files, each of which is a digital copy of a single page. The set comprises all pages of the book, including cover and backside pages. All pages are numbered consecutively in ascending order. The numbering starts with 1, meaning the page number 1 is assigned to the cover page. Besides the cover and the backside page, a book usually contains several additional pages of non-score information. Usually, these reside at the beginning and the end of the book and include publishing information, foreword, table of contents, and pictures.

4.2.2.2 Rips of audio CD boxes

An audio CD box is an ordered set of audio files, each of which represents an individual track on a CD included in the box. The set comprises all tracks of all CDs of the box. The CDs are numbered consecutively in ascending order, starting with 1. The individual tracks on each CD are also numbered consecutively in ascending order, starting with 1. Each track is assigned both the number of the CD within the box and the track number on that CD. This allows for an exact localization of each track of the collection.

4.2.2.3 Content provider-owned metadata catalog

In the context of this thesis, we assume the availability of content provider-owned metadata, stored in traditional, non-hierarchically organized metadata catalogs. Those metadata catalogs consist of metadata records, each of which corresponds to exactly one manifestation and contains global information about that manifestation (most notably, title and publisher). In general, metadata are exclusively available on the level of manifestations as a whole—a finer subdivision, which is particularly useful for music documents, is disregarded here. This most common case found in practice means that metadata associated to individual expressions are missing. In other words, metadata for individual pieces of music, movements, or tracks contained in sheet music books or on CDs are not directly available from metadata catalogs.

4.2.3 Secondary data resources

Considered secondary data resources are symbolic scores derived from OMR-processed sheets of music and external metadata. In the following, considered secondary data resources are described in more detail.

4.2.3.1 Symbolic scores

In addition to the scanned pages of sheet music books, from each single page that contains actual score content, the contained symbolic score is made available through OMR processing.

The OMR processing is performed by the commercially available OMR software package SharpEye Music Reader 2.68 [68]. A page-wise OMR processing results in a set of MRO files, each of which explicitly encodes the underlying symbolic score content of a single page of a sheet music book.

4.2.3.2 Additional metadata

In the context of this thesis, additional metadata for CD tracks are gathered from the Gracenote [56] service. By means of these, in addition to catalog-provided metadata about a CD as a whole, also metadata are available on the level of individual tracks contained on that CD. These additional metadata are available as plain text files, one for each CD, wherein respective track titles are represented by character strings, one title per line of text. Typically, the format of those text files is inconsistent, i. e., available text files are heterogeneous regarding naming conventions and contain errors. Throughout this thesis, this format is referred to as Gracenote format.

4.3 Cross-modal indexing of music documents

This section covers the overall processing of music data from a business process-oriented view. Thereby, it lays the methodological foundation for a highly automated indexing of musical content. For the purpose of indexing musical content, music IR techniques are suitably combined and systematically applied using a consistent workflow. In the following, appropriate workflow-supporting tools that deliver streamlined ingestion and maintenance of music documents are described in detail.

4.3.1 Semi-automatic metadata acquisition and assignment

One objective is to minimize necessary manual cataloging effort and to automatically generate appropriate metadata wherever possible. The main source of existing metadata being acquired in the context of the PROBADO music DL is the BSB cataloging database. It contains metadata for the entire library collection in the MAB[3] format, the German counterpart to the Anglo-American MARC format. For this, the MAB records related to music documents have to be converted into FRBR-based metadata records. This process is also referred to as *FRBRization* [141].

In this way, all information about a work are centralized in one record. Records for subsequent expressions of that work would add only the information specific to each expression. For example, an audio recording of an orchestral performance of Beethoven's “Symphony No. 5” conducted by Karajan does not need to repeat the fact that the work was written by Beethoven. This approach has certain inherent advantages for collections with many versions of the same works. Newly published work-related material can be cataloged more quickly, and records can be stored and updated more efficiently.

In the context of the PROBADO music DL, much work has gone into rethinking what information should be contained in catalog records, how the records should relate to each

[3] The MAB (Maschinelles Austauschformat für Bibliotheken) is used mainly in the German librarianship, where it serves as a common interchange format for metadata.

other, and how to automatically convert existing (traditional) catalog records. It turns out that not all parts of the conversion process can be automated. Especially the extraction of metadata on both work and expression levels requires manual intervention. Although it will be possible to do this cataloging work manually for small collections, further automation of this process is required for the future augmentation of bigger collections.

In addition to the BSB cataloging database there are other data sources that we consider for obtaining metadata. Besides Gracenote metadata that are used for the preferably consistent labeling of audio CD tracks, also the German-wide authority files for personal names (PND) and corporate bodies (GKD) are used for the contributor entity.

4.3.2 Highly automated music document preprocessing

Prior to accessing musical content in the desired fashion, it has to be indexed on the metadata level and has, on the basis of its substance, to be stored and prepared for dissemination by means of appropriate tools. For this purpose, both a preprocessing workflow and a DMS were developed that take the role of the administrative or authorial part of the music DL framework. The DMS features a document processing chain that is controlled by means of an intuitively operated UI.

To obtain an efficient and user-friendly process for the generation of all required data that are utilized by the music DL system, the automation of all required preprocessing steps is desirable. As a practical solution towards full automation, a workflow was developed for the indexing of music collections comprising sheet music scans, CD-audio recordings, and metadata catalogs. The DMS aims at providing an intuitive UI to support the preprocessing steps required to generate this data, including primary data, indexing and synchronization structures, and other secondary data such as album cover images. Figure 4.1 depicts an overview on the document preprocessing that is subdivided into several subsequent stages of the approached document processing chain.

In the context of this thesis, the following distinction between the concepts of music documents and musical content is made. Content is the broader term for primary and secondary music data that has been indexed by running through the document processing chain, see Subsection 4.3.2.2. A document denotes a single self-contained entity of primary music data that is being submitted to the document processing chain. Here, it is analyzed, indexed, converted, and cross-linked to other documents, where cross-linking takes place on several levels of granularity and across modalities. After passing the document processing chain, resulting index data is stored in the repository and, among other things, is made available for search. Documents also refer to the entities that retrieval engines report in their results.

The document processing chain is controlled and monitored through the DMS, which delivers a UI-based administration tool for the management of music documents. The DMS provides functionality for adding, manipulating, and removing work entries and metadata, as well as for ingesting documents into a repository by passing them through the document processing chain. The content-managing person controls particular stages of the document processing chain through MACAO (acronym for *"Music management tool for content-based analysis and organization"*), guided by the workflow. The workflow relates to both the document processing stages involved and the order of their execution, and defines the course of actions to be carried out by hand. MACAO catches processing failures, whereupon appropriate messages are presented to the content-managing person, whereupon the latter must intervene. In the

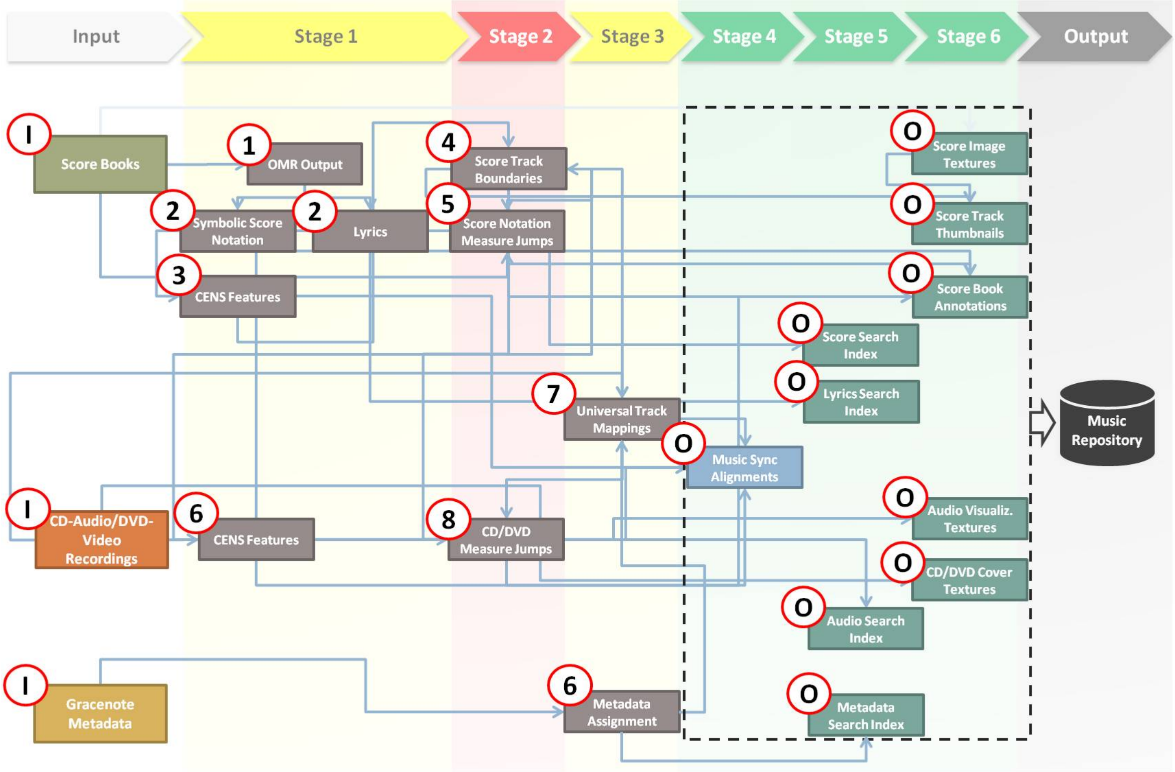


Figure 4.1 Overview on the highly automated document preprocessing.

following, a more detailed description on involved preprocessing components is given, including the workflow, the document processing chain, and the DMS.

4.3.2.1 Workflow

The workflow describes a sequence of manual steps to be performed in order to ingest, maintain, and prepare documents in a music DL repository for the purpose of dissemination. The main objective of the workflow is the supervision of the document processing chain. The workflow models the actual (manual) work and business processes, and it defines processes and execution parameters. Prescribing a definite course of manual steps, it controls and supervises the execution and information flow between individual stages of the document processing chain. By applying the workflow, new documents can be efficiently added to the repository and properly prepared in a streamlined fashion, ready for dissemination. The lower part of Figure 4.1 gives an overview on the workflow and shows individual administrative tasks that are interconnected with corresponding stages of the document processing chain at the top.

4.3.2.2 Document processing chain

To insert documents into a repository, they are processed by the document processing chain—the core component at the backend of the framework. Documents are sent through the document processing chain that consists of multiple stages, as depicted in Figure 4.1. The stages are executed sequentially and transform data from one into another representation.

When a document runs through the document processing chain, it passes each individual stage that involves reading input data, performing calculations based on this data, and output calculation results in a suitable manner such that it is accepted as an input of a subsequent stage. Prior to the first stage of the document processing chain, the data format of the input primary data is detected. On this basis, a conversion into a suitable format accepted by the first stage of the document processing chain takes place. Subsequent main tasks are feature extraction, intra-modal and cross-modal similarity analysis, mapping, and synchronization of music data.

The generation of intermediate results can be individually controlled for each of the processing stages. The shown arrows indicate the mutual dependencies between the various stages. Due to the chosen partition of the document processing chain into individually controlled processing stages, the manual manipulation of intermediate results is possible prior to further processing stages. To this end, some of the stages generate error-prone data. At these stages, a manual intervention is crucial for a successful further processing. Further details on this issue are addressed in Subsection 4.3.3. From a process-oriented point of view, three different data types can be distinguished. The input data (labels I) are at the front of the document processing chain and currently consist of scanned sheet music book pages and audio recordings. The intermediate data (labels 1–8) contain derived data that are required for the processing chain but do not contain information required by the service runtime environment of the music DL system. The output data (labels O) are at the back of the document processing chain and comprise derived data that are directly used by the service runtime environment. In particular, these data comprise index structures for content-based retrieval, compressed audio and image texture data for rendering audio recordings and sheet music book pages, synchronization data for fusing documents, and metadata.

The stages of the document processing chain are classified into three classes that correspond to the colors of a traffic light system, cf. Figure 4.1. The classes identify different grades of automation, depending on reliability and accuracy of calculation results w. r. t. partial processing stages. Whether and in what way individual stages require manual intervention is indicated by one of the following classes:

- Class 1 (red): the result of this stage might be not as reliable as required to meet high quality demands. Its outcome is likely to be of mediocre or insufficient quality. Therefore, a manual revision is mandatory. This stage cannot be completed without manual intervention.
- Class 2 (yellow): the result of such a stage may be somewhat unreliable. Its outcome is likely to be of high or at least sufficient quality but may optionally be improved where necessary by manual revision. However, in principle, such a stage can be completed without manual intervention.
- Class 3 (green): computations performed in such a stage rely exclusively on prior processing results. The result of such a stage cannot be revised. In case of an unsatisfactory outcome, the reason for this is to be sought in preceding calculations or stages. Such stages are completely processed in a fully automated fashion without any opportunity for manual intervention.

For reasons of quality assurance, in the first three stages of the document processing chain the documents run through, manual intervention and modification of individual processing results is possible or required where information is insufficient or missing. Particularly, externally provided metadata and automatically generated OMR results potentially yield insufficient or error-prone data. Since these are very basic processing steps, the content-managing person is forced to review the generated data prior to further processing. Therefore, corresponding stages belong to the first class.

The results of each stage are logged for administrative purposes. Here, major decisions, errors, and warnings are logged. Log entries contain stage, time, and date for error tracing purposes. As the logging is verbose, the advantage of logging is that the preprocessing of particular documents can be traced and analyzed.

4.3.2.3 Document management system

For the management of music content at the backend, the conception and development of a DMS incorporating a combination of semi-automatic and fully automated processes towards indexing and organization of content is to be targeted. To ensure high quality standards[4], intervention possibilities are featured for critical processing tasks with possible degraded or insufficient output quality.

DMSs are solutions and tools that originally arose from the need to provide administrative functions to be able to handle fast growing document collections. Therefore, DMSs equipped with highly automated processing chains become increasingly important for the management of documents. For example, in the music DL scenario, the discovery and management of interrelations between entities within musical content is no longer affordable by manual effort. Therefore, the DMS developed in the context of this thesis provides a widely automated document processing chain along with a corresponding workflow, that seamlessly integrate in the document lifecycle and support librarians and other content-managing persons in their daily work.

The document lifecycle describes the whole process that documents typically run through in the music DL, from document creation over the indexing and retrieval up to the deletion or storage in a long-term archive. The beginning of the document lifecycle marks the point where documents are put into the DMS.

The DMS is a key component of the music DL framework and refers to the authoritative part of the document lifecycle. More precisely, it is to be understood as the entirety of methods, procedures and processing tools for the ingestion, indexing, management, control, and storage of music content. It integrates a multitude of music IR techniques and manages a central music repository consisting of scanned sheets of music, audio recordings, and metadata. Information derived from documents such as document type, storage location, and extracted features are stored as database-driven metadata and by using specialized content-based indexes. By means of this information, documents can be efficiently searched and retrieved, based on attributes such as author and title, but also by means of content-based query fragments consisting of text, score, audio material, as well as a combination of them. Note that the actual retrieval process is carried out in the service runtime environment and the Web interface, where search queries and results are properly handled by the latter.

The main tasks of the DMS are the mapping, controlling, and monitoring of the document processing chain by means of the workflow. The DMS provides alerts when processing stages

[4] Quality assurance is a crucial issue in the real-life library scenario, particularly in the context of cultural heritage applications.

fail and manual interaction becomes inevitable. Essentially, the DMS concerns the following items that are typically found in a DMSs: (a) indexing methods and structures, (b) storage structure and archiving policies, (c) descriptors and attributes used for both formal and content-based indexing, (d) conventions for used attributes (naming etc.), and (e) business processes and process management (workflow and document processing chain).

4.3.3 Document processing

This section gives a task-oriented description on the particular steps required for the semi-automatic indexing of music documents in order to ingest them into the music DL system. Figure 4.1 depicts the document processing chain and shows individual stages of the document processing chain the music documents run through. In the following, each of the six stages of the document processing chain is described in detail.

4.3.3.1 Macro and micro processing

The second and the third stage of the document processing chain are classified as *macro processing* as they operate on macro-level granularity. Here, large coherent regions within music documents, referred to as sections, are considered. In case of sheet music, a section spans over one or more consecutive pages of sheet music books, embodying notations of whole songs or movements. In case of audio recordings, a section spans over one or more consecutive CD tracks, embodying performances of whole songs or movements. Note that in both cases, songs or movements may begin or end in the middle of a page or track. In case of metadata, global, static information are available about notations or performances of whole songs or movements that are manifested within sheet music books or CD boxes, respectively.

The rest of the stages of the document processing chain are classified as *micro processing* as they operate on micro-level granularity. Here, small fragments of music documents, referred to as snippets, are considered. In case of sheet music, a snippet refers to a small number of consecutive notes of the contained score. In case of audio recordings, a snippet refers to a short excerpt of the contained audio signal.

4.3.3.2 Processing stages

In particular, the six stages of the document processing are:

1. Score, lyrics, and feature extraction.

2. Work identification, cross-modal mapping, and fusion.

3. Metadata assignment and classification in the hierarchical music DL data model.

4. Music synchronization.

5. Building up search and browsing indexes.

6. Preparation of visualizations and audio coding.

The produced results of the first stage do not need revision, but for improving results, this stage offers optional manual intervention. The second and the third stage require revision of produced results and thus force mandatory manual intervention. The results produced by the last three stages cannot be revised, hence these stages do not require any manual intervention. Each of these stages is explained in detail in the following.

Stage 1: Score, lyrics, and audio feature extraction In this stage, several types of information are extracted from the music documents.

First of all, the score content is extracted from sheet music documents by using OMR, prior to further processing. This process results in a symbolic representation of the sheet music documents. As this process is quite error-prone, post-processing techniques are used to improve the recognition result, including the application of semantic interpretation of the extracted symbols. Especially the detection of the musical form or structure under difficult conditions such as false detection of bar lines is a demanding research task. This and other challenges have been tackled in [46] and it has been shown that the recognition result can be significantly improved as compared to original OMR results yielded by SharpEye. This process can be done in a fully automated fashion and is fairly reliable in conjunction with the post-processing—the score extract is reasonably accurate. A revision of the result is only needed in rare cases. As these rare cases may nevertheless affect the quality of the music synchronization results, a manual revision might be of interest in order to improve the synchronization accuracy. Therefore, the score extraction process is done in a fully automated fashion with an option for manual intervention.

In addition, a MIDI-like representation is derived from the extracted symbolic score content. This representation is a sequence of simple note events, each of which is composed by a pitch and an onset time value. A post-processing automatically corrects some minor OMR issues and quantizes the onset time values on a 64th grid. This conversion is done in a fully automated fashion with no possibility of manual intervention as it is directly derived from the extracted score content where manual corrections were possible.

By incorporating an additional OCR component, the OMR process also includes the recognition of textual content, especially syllables belonging to lyrics. This allows for the extraction of the lyrics of a musical work. In a post-processing step, individual syllables are merged into whole words which in turn are afterwards matched against a dictionary and corrected if necessary. Note that in most cases lyrics are externally available, e. g., from online resources—the actual advantage of the integrated OCR is to provide explicit synchronization information between text and notes. In most cases, subsequent text passages of a song are distributed among several lines of a score sheet, reflecting the repetitive structure of the musical work. That is, originally the lyrics are available in a “folded” textual format. Exploiting the recognized musical structure gained by the score extraction process, the lyrics text can then be “unfolded” to finally yield a linear word sequence. In rare cases, text fragments not belonging to the lyrics are extracted by the used OMR software and incorrectly assigned to the lyrics. Such text fragments could be conducting and tempo instructions or edition information. This can also be corrected by a suitable post-processing of the OMR result. With the help of external lyrics text files—if available—and application of DTW between the latter and the extraction result, a further improvement of the accuracy can be achieved. Either or both of the auto-correction strategies can be applied. Exploiting knowledge regarding the offset position of each syllable within a measure combined with the measure-wise synchronization of the score content and corresponding time segments, also the times can be accurately estimated of when and which word is sung in individual performances. To enable accurate lyrics–audio synchronizations, we propose to associate linearly interpolated times to the individual words. The lyrics extraction process is done in a fully automated fashion. Although the extraction quality is quite high and a manual correction can be omitted, the system nevertheless offers an option for manually correcting the extracted lyrics.

The extraction of feature sequences from both score and audio content on the one hand builds the basis for the identification, the mapping, and the music synchronization processes. As a direct comparison between score and audio content is inherently impossible, identifications and mappings or music synchronizations are actually done by calculating alignments between feature sequences that are comparable regardless of document types, gained from different music documents and types. On the other hand, the extracted feature sequences are used to build up the various content-based indexes. That said, the feature extraction process forms the core link for incorporating the aspect of cross- and multimodality. A feature-based representation contains too many data points and is—semantically—in no suitable format to be accessed and edited by humans. Therefore, the feature extraction process is fully automated.

Overall, this stage is processed in a fully automated fashion with several options of manual intervention at certain points in order to improve the quality of selective intermediate results.

Stage 2: Work identification, cross-modal mapping, and multimodal fusion In this stage, all manifestations of musical works scattered over various documents, and document types as well, are identified based on their respective underlying musical content. Aiming at a work-centric collocation, all relevant parts of music documents that are mutually related to the same musical work have to be identified. This procedure includes the segmentation of sheet music books into consecutive parts corresponding to expressions of songs or movements and the subsequent cross-linking to semantically corresponding audio tracks. For this, boundaries of expressions within manifestations are calculated. More precisely, for the case of scores these are start and end measures within sheet music books. For the case of performances, these are start and end times of tracks on CDs.

After determining the boundaries, the latter are suggested to the content-managing person. Large-scale evaluations of this procedure, applied to a wide range of Western classical music documents, showed an overall identification accuracy of roughly 80 % [46]. As this stage lays the foundation for the subsequent music synchronization process, the boundaries must be manually revised for the purpose of quality assurance.

Hence, this stage is processed in a semi-automatic fashion with mandatory manual revision, as it is slightly unreliable and the correct determination of track boundaries is crucial prior to further processing stages.

Stage 3: Metadata assignment and classification in the music DL data model In this stage, work-level metadata from content provider-hosted metadata catalogs are extracted and assigned to the metabase according to the music DL metabase scheme. This procedure is done for recycling reasons to reduce the needed manual effort of entering metadata on the work-level. It turns out that not all records of library-hosted MAB-formatted metadata catalogs can be converted into the music DL metabase format in a fully automated fashion [38].

Supplementary to the recycling of content provider-hosted metadata catalogs, additionally provided external metadata are used for the acquisition of track-level metadata associated to audio CDs. As external metadata may vary in formatting and naming conventions, they have to be revised manually.

By utilizing the identification and mapping of music documents across type and modality, additional audio metadata is being automatically assigned to corresponding sheet music pages. Exploiting this together with both metadata sources, sufficient information is available to fully fit the work-centric approach of the music DL data model. However, assigned metadata are potentially error-prone. Therefore, they are presented to the content-managing person as suggestions with the choice of being confirmed or corrected.

Overall, as the extraction and assignment of metadata is somewhat incomplete and unreliable[5] due to the partly inferior quality of available resources. This stage is hence processed in a semi-automatic fashion.

Stage 4: Music synchronization In this stage, all collocated music documents or parts thereof belonging to the same musical work are aligned among each other, based on their respective feature representation. In particular, the synchronization of semantically corresponding parts of music documents produces a linking structure across document parts both within the same and across different modalities. As a result, a substantial cross-linking on a fine-grained level is established.

As for robustness and application-specific reasons, music synchronization is performed at several different resolution levels that depend on the music documents to be aligned. The coarsest bar-wise resolution level is used for the synchronization of score and audio content. A finer resolution level is used for the word-wise synchronization of lyrics and audio content. The finest resolution level of a tenth of a second is used for the synchronization of two audio recordings. Note that for the purpose of switching between different acoustic performances of the same musical work, the resolution level for the synchronization of pure acoustic content must be fairly high, as the human auditory system is very sensitive in noticing latencies produced by a low time resolution and thus inaccurate synchronization. In contrast, a lowered resolution level for the synchronization of graphical and acoustic content does not affect the perception that much.

The quality of the synchronization process relies on preceding processes (OMR, identification and mapping, detection of the musical form, and feature extraction) which in majority offer the possibility of manual correction of processing results. At this time, the linking structure generated by the synchronization process is not directly revisable. Synchronization errors of score and audio content can potentially be corrected by modifying extracted score (see Stage 1). Assuming a reasonable quality of the prior processing stages, the calculation of music synchronizations is reliable in almost all cases. In case of conformity regarding the musical structures of two expressions of the same musical work, a failure, however, has only a local impact, meaning that only a few score bars are misaligned.

Therefore, this stage is processed in a fully automated fashion with no possibility of manual interaction.

[5] Note that the classification of metadata catalog records in the music DL data model is not completely machinable.

Stage 5: Build-up of search and browsing indexes In this stage, several search index structures are constructed, built upon score, lyrics, and feature sequences as well as metadata, extracted in prior stages. These structures are then used for both metadata- and content-based retrieval purposes for the various available query types and modalities. Besides, various tree structures are constructed from metadata for browsing purposes.

For the purpose of simple metadata-based retrieval, metadata of manifestations are enriched by superordinate metadata from the expression and work levels, and indexed as a whole string in a full-text search index structure. Although this approach might be redundant, it is pursued due to speed advantages—a considerably faster retrieval justifies the redundancy. The advanced, fielded metadata search is performed by directly using Structured Query Language (SQL) statements. Hence, for this kind of search no special index structure is used. Furthermore, the metadata are hierarchically organized in tree structures according to certain criteria, in order to browse manifestations according to different criteria as described in the following. In one tree structure, manifestations are organized such that first-order tree nodes contain composer names, each of which points to second-order child nodes containing all the works created by the respective composer. In another tree structure, first-order tree nodes contain interpreter names, each of which points to second-order child nodes containing all the works containing interpretations where the respective interpreter has contributed to. In yet another tree structure, first-order tree nodes contain musical work titles, each of which points to second-order child nodes containing all involved persons related to that work, including creators, performing artists, singers, conductor, etc.

For the purpose of content-based retrieval, several of the prior extraction results are indexed. From score contents, MIDI-like note event sequences are derived and indexed in the score index. From audio contents, feature sequences are extracted using signal processing techniques and indexed in the audio index. From lyrics contents, word sequences are extracted and indexed in the lyrics index.

This stage is processed in an automated fashion with no possibility of manual intervention or revision.

Stage 6: Dissemination-ready image and audio data preparation In this stage, the creation of compact textual, visual, and acoustical data derived from primary data is prepared. The resulting data is ready for dissemination to the frontend of the music DL.

From sheet music documents, OpenGL-formatted image textures are extracted and stored in the file system. From the extracted lyrics, for each audio recording that contains the lyrics, an XML-formatted lyrics file containing words and associated, appropriate timestamps within the audio recording are constructed and stored in the file system. From the audio recordings, a frequency-binned spectrogram consisting of a sequence of short-time spectra, where each spectrum has a resolution of 88 frequency bins, is calculated, converted to OpenGL image textures, and stored in the file system. Additionally, the audio content is encoded using an MP3 encoder. This is done in order to save bandwidth while streaming acoustic data to the frontend of the music DL.

This stage is processed in a fully automated fashion.

4.4 Multimodal retrieval of music documents

This section covers the retrieval of music data. Thereby, it lays the foundation for a consistent retrieval concept, where music IR techniques are suitably combined and integrated to allow for a work-centric and modality-fused interaction with music documents. For a user-centric view on the cross- and multimodal interaction with music documents, see Chapter 6.

4.4.1 Searching and browsing strategies

A collection of music documents can be searched using different strategies, criteria, and modalities. First, documents can be searched implicitly by browsing hierarchically organized metadata catalogs. Second, documents can be searched by explicitly specifying search queries. Here, two distinct approaches are supported: traditional metadata-based searches and content-based searches. Metadata-based searches may be applied on the level of works, manifested expressions, and multi-expression manifestations (i. e., whole books or CD boxes). Content-based searches apply on the level of raw data of individual music documents.

4.4.2 Query specification and formulation

As mentioned, in order to search explicitly for music documents, a search query must be formulated. This is done by specifying descriptors that can be compared to or matched against indexes built from music documents and the metabase. These descriptors are formed of metadata attributes, content-based features, and any combination thereof. That is, queries are composed of multiple descriptors of potentially different modalities. While metadata attributes are expressed in a single (the textual) modality, content-based information are expressed in several modalities. Accordingly, the query formulation can take place by using different modalities and paradigms as well. Firstly, free-hand queries can be formulated by specifying words, phrases, melodies, and complex scores in several adequate forms by means of text and notes. Secondly, applying the query-by-example paradigm, queries can be specified on the basis of manifestation excerpts by graphically selecting particular regions within manifestations. Since selected regions that are to be queried concern multiple modalities, different modalities can be queried altogether or apart from each other. To facilitate the formulation of composite queries consisting of text, scores, and sound, a bag-of-queries is iteratively compiled with partial queries from individual forms or selected regions.

4.4.3 Retrieval result organization and document surrogates

As a result of entering a query in the music DL system, the system retrieves a set of results that match the query specification in some sense in at least one manifestation or one metadata attribute, and displays the results to the user in the form of document surrogates in a work-centric manner. The results are presented in form of a list, where each list entry corresponds to a work comprising documents that relate to the query. The list entries are ranked so that those results being most relevant to the query are shown first. The results can be browsed by means of a navigation control that allows for scrolling back and forth in the list.

A single search result provides several textual information about the matching work, including composer's name, work title, matching modalities, and, if available, lyrics excerpts. The lyrics excerpts are the top-3 matching positions of retrieved query terms, shown within their respective context and with search terms highlighted. A search result includes links to manifestation parts that match the query. It can be chosen to load corresponding manifestations and jump to the top matching part. All matching parts within a particular manifestation are highlighted. For an overview, these are additionally arranged on a timeline, where they can be chosen from in order to navigate, view, and listen to respective matches.

4.4.4 Multimodal presentation of music documents

Retrieval results are presented in the context of their respective containing manifestations in order to make the result set more understandable. This includes presentation of the relationship of individual retrieval results to the query, showing descriptive metadata, and showing occurrences within document structures. The approach discussed in the following allows for a unified, work-centric presentation of retrieval results.

4.4.4.1 Work-centric document organization and display of document surrogates

As it is the most common way to visually present the results for a particular query, matching works or manifestation parts are sorted in the result list in descending order w. r. t. their computed relevance to the query. Works and manifestation parts that match on a metadata level and have the same relevance, e. g., in case of a field-based metadata search which is based on Boolean retrieval, are sorted by their natural occurrence order in the manifestations. Thus, individual motives belonging to the same work or songs belonging to a song cycle are placed in the correct order. Each entry of the result list consists of a summary of a matching work. This type of result representation, where documents are in aggregate form and shown as a single result list entry, is referred to as a document surrogate. In particular, a result list entry contains the work's composer name and title, as well as a subset of important metadata such as matching modalities (why have the documents been found in the repository?), if available an excerpt of contained lyrics, and a control for showing all works that belong to the composer. Alongside the title, a numerical score (normalized relevance accounting for percentage in results) is shown, indicating an estimated probability of relevance, i. e., the ranking value.

Choosing a document surrogate will bring up a detailed view of the work itself in the multimodal music player, including all available documents that embody that work. From here, all query occurrences in manifestations can be examined in detail. On a more detailed level, they are arranged on a timeline for a visual summarization at a glance. On the most detailed level, they are highlighted in the manifestations.

4.4.4.2 Matching positions within documents

At the frontend of the music DL system, the user is enabled to view or listen to all work-containing manifestations of a currently retrieved work. Since manifested works are not isolated but reside in whole sheet music books and CD boxes, they are presented to the user in their manifestation context. Here, because of the document sizes, it is useful to scroll all manifestations' views to the first passages containing relevant parts regarding the selected work. Note that this kind of presentation also allows for navigating through the context of passages possibly related to a work.

On a fine-grained level it is useful to highlight all occurrences of terms or descriptors that match those ones of a content-based query. The highlighting is thought to help drawing the user's attention to those parts of the manifestations that are most likely relevant to the query. It should be mentioned that the highlighting of document parts matching the query has been found to be a useful feature in the UIs of IR systems at several occasions [86, 3]. In consequence, the frontend supports a consistent concept for highlighting query occurrences or matches in contrasting colors in two different, simultaneously used fashions. Firstly, occurrences are visualized by semi-transparent colored regions within individual, different manifestation views. More precisely, corresponding (a) bar sequences within the score view, (b) segments within the audio view, and (c) words or phrases within the lyrics view are highlighted. Secondly, occurrences are depicted as colored boxes or markers on the timeline bar of the respective view. Here, those ones that represent matches contained in the active manifestations (i. e., those ones currently selected for playback or visualization) are colored, where primary color and intensity encode the matching modality and relevance (ranking value), respectively. The ones representing matches within inactive manifestations (i. e., all other ones belonging to the work, but are currently not selected for playback or visualization) are grayed out. Besides from just giving an overview on the arrangement of query occurrences over time, the boxes are also to be used for navigation purposes. In this way, we have a consistent concept that runs through all modalities and views.

4.4.5 Synchronous, multimodal playback, visualization, and navigation

With the help of semantic linking structures, see Section 4.1.3.2, both within and across different documents and document types, advanced playback and navigation capabilities become possible. The synchronous, multimodal playback and visualization of different expressions of pieces of music is one of various appealing application scenarios. Here, the benefit is a modality-fused overall user's perception that triggers an increased cognitive perception and experience of music. Moreover, a cross-modal navigation in audio recordings by means of corresponding sheet music can help in finding particular positions in audio recordings. All this functionality is realized in the multimodal music player.

4.4.5.1 Score–audio synchronization

A score–audio synchronization enables the user on the one hand to visually track the currently played measure within the sheet music representation of the concrete audio recording he is listening to. On the other hand, the sheets of music can be used to change the playback position within the audio recording. In this sense, interactions regarding one modality are reflected within the other modality.

4.4.5.2 Lyrics–audio synchronization

A lyrics–audio synchronization allows for a karaoke-like application where the user can see which word is currently sung within the audio recording he is listening to. Again, the linkage can be utilized to change the playback position to a specific word.

4.4.5.3 Audio–audio synchronization

An audio–audio synchronization enables the user to switch between different interpretations (audio-recordings) while retaining the actual playback position in a musical sense. Among others, this allows to draw local comparisons between different interpretations belonging to the same piece of music.

4.5 System architecture

This section covers the system architecture of the music DL. The system architecture conceptually models the structure, behavior, and views of the system considering technical aspects. In particular, the system architecture describes the structure of the system which comprises system components (technology-enabling core entities), externally visible properties of those components, and their complex inter-relationships (e. g., the behavior) between them at different abstraction levels. From a high-level view, the music DL is a distributed system, wherein subsystems interact with each other in various ways. Based on the system architecture, subsystems can be developed and interconnected, which will work together to implement the overall system. Thus, the system architecture delivers a construction plan for setting up the targeted music DL service. For this, functional requirements, non-functional requirements, and technical requirements that affect the former ones, are taken into account. While the functional requirements refer to the set of functions the system is expected to offer, the non-functional requirements refer to the manner in which these functions are performed, and concern qualitative aspects such as service driven, scalability, and extensibility. The technical requirements pertain to the technical aspects which concern technology to be used and that will advise the development of the system, such as performance-related issues, reliability issues, and security issues. The details of the system architecture are presented in the following.

4.5.1 Design aspects

First, we focus on how the music DL system is specifically designed to satisfy functional requirements specified in Subsection 4.1.3. As the core business of libraries and other content providers is to help their customers with their information needs, there is a corresponding increase in expectations on performance, uptime requirements, and in the need to remain both flexible and scalable. In such a demanding scenario, design and implementation are critical for an appropriate system architecture. In the following, the specific system design is described, covering components, modules, interfaces, and data for that system, based on a preceding requirements analysis, considering certain design aspects (see Subsection 4.5.1.2). As the system architecture affects all aspects of software design and engineering, a variety of key issues must be considered in order to construct an effective architecture, from which the most important ones are application complexity, integration and interfacing, as well as network infrastructures. In addition, the system architecture also strongly affects flexibility and maintenance aspects.

4.5.1.1 Conceptual overview of the music DL system

In order to realize a music DL system under the stipulated aspects, the following main fields need to be addressed. System architecture affects all aspects of software design and engineering.

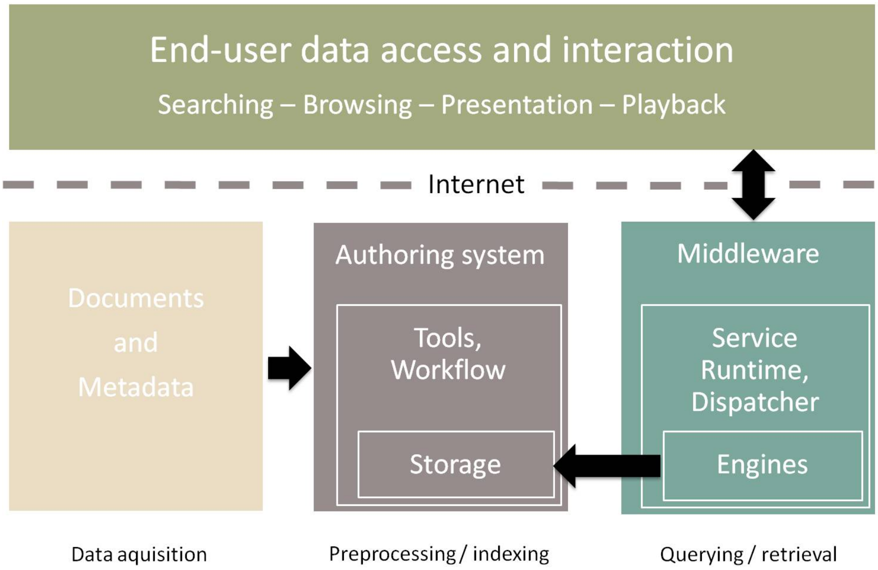


Figure 4.2 Conceptual overview of the music DL system. Documents and metadata are authored and semi-automatically processed at the backend by means of the DMS. The middleware of the service-oriented IS delivers musical content to the frontend and receives client requests therefrom.

Aspects such as the complexity of the application, the level of integration and interfacing required, the number of users, the properties of relevant networks, and the overall transactional needs of the application are to be considered. The design of the system architecture affects development time, future flexibility, and maintenance of the music DL system. The conceptual overview of the music DL system is illustrated in Figure 4.2.

Management tools and workflow The DMS covers all important issues of digital content provision. Principal function of the DMS is to centrally index and store musical content from particular audio recordings, sheets of music, associated metadata, as well as other music-related material, and to make it accessible to end-users. Metadata records for works and associated expressions are stored in an SQL database, together with segment boundaries within all corresponding manifestations. Actual manifestations or documents are stored in the file system and are referenced by further entries in the metadata records.

The workflow for document processing is carried out with the administrative UI components of the DMS. Music content managed by the DMS can either be automatically transferred to other repositories (e. g., for the usage by other operating sites) or they are manually input by a content-managing person.

Dissemination services and clients Regarding user–data interaction, the conception and development of middleware services including a broker for the retrieval, delivery, and interaction with musical content has to be focused. Here, the compliance with commonly designed interfaces that meet common software standards is an important issue to take into account in order to embed the system as building block in a superordinate architecture. The component-based, modularly designed service runtime environment incorporates a generic query processing engine framework featuring a plug-in mechanism for the extension of functionalities. Individual functionalities are encapsulated in single query engines and provided to the service runtime environment. Particular query engines are triggered by corresponding queries, e. g., search queries, requested by clients. With this, a standardized and open communication infrastructure over the Internet is being achieved.

Regarding human–computer interaction, the conception and development of intuitively operable UIs that support new ways of music content interaction using established Web 2.0 techniques is focused.

4.5.1.2 Key principles, aspects, and paradigms

The music DL system is designed for the dissemination of music content over the Internet and for usage from within standard Web browsers. The system architecture is based on the client/server model of distributed systems. This model determines how resources are partitioned among software components and distributed on a network. Numerous approaches of partitioning resources exist, of which the most recognized are the two- and three-tier architectures. These architectures enable a single server or a family of servers, commonly referred to as middleware, to provide services for a number of clients at the same time. In particular, the construction of the music DL system architecture takes into account the following considerations regarding non-functional requirements, including design principles, aspects, and paradigms.

- *Service driven.* The architecture is driven by the music DL services it provides. Therefore, the architecture is technically based on a SOA. The reference implementation uses a Web service hosted by an application server (currently, Apache Tomcat) and an RMI-over-Internet (RMIoI) service hosted by a repository server for the delivery of music DL services.

- *Open architecture.* The architecture is open and supports interoperability among heterogeneous, distributed systems through the Web service. All functionality is partitioned into a set of well-defined services which are accessible via well-defined protocols, specified in the Web Service Definition Language (WSDL) [135] and the Java language.

- *Modularity.* The architecture represents a modular, component-based approach. It further promotes interoperability by different technology-enabling subsystems and components that are able to communicate over well-established communication protocols. Particular subsystems and components are loosely coupled and communicate through stable interfaces. Hence, they can partially be replaced by other implementations without side effects.

- *Practicality.* The architecture represents a flexible and practical approach that takes into account economic aspects. It is built on industry standards, including standardized protocols, and the reuse of established software frameworks. This should drastically decrease software maintenance efforts.

- *Scalability.* The architecture is robust, scalable, and reliable even at high transaction rates by means of adequate scheduling and threading mechanisms.

- *Client support.* The architecture supports a baseline level of services, including querying, retrieval, and browsing, which can be accessed with common desktop hardware and software configuration. These services are accessed through the Simple Object Access Protocol (SOAP) [59] using any JavaScript-enabled Internet browser. Certain higher-level services, including cross-modal and content-based querying, streaming, as well as multimodal presentation and navigation, require proprietary clients. Those services are accessed through the Java Remote Method Invocation (RMI) [58] protocol using any Internet browser that is capable of running Java applets. As client reference implementation, the services can all be accessed through the AJAX (acronym for *"Asynchronous JavaScript and XML"*)-based Web interface, in which context several Java applets are being executed.
- *Session based.* The architecture provides session management for stateful data exchange between clients and services. For each client, an individual session, which temporarily stores client states, retrieval results, and other intermediate data, is created on initial request and subsequently reused for that client. A session expires after a certain time span of client inactivity, whereupon it is removed.
- *Security.* The architecture is sensitive to security issues and incorporates secure access mechanisms. It uses login credentials and signed applets for trustworthy data exchange between clients and services. It further allows for both public (anonymous) and restricted (individual permission settings) access to baseline- and higher-level services and resources.
- *Distribution.* The architecture incorporates services to ingest, process, store, maintain, retrieve, and disseminate multimedia content and metadata of audio-visual music content along with textual metadata, where this multimedia data is transferred in chunks to clients on demand using streaming mechanisms.
- *Uploads.* The architecture incorporates services to upload multimedia content from clients for the purpose of content-based retrieval.

4.5.1.3 Web application

The conception of the client was intended to keep technical requirements on the user side at a minimum in order to achieve a high acceptance of the music DL system. The repository client is based on a mixture of Web browser scripting languages and the Java language and runs in a Web browser. Through the design of the system as Web application, a wide range of target platforms is supported using cross-platform Web technologies and Web browser plug-in technology. As baseline functionality, it provides the search of musical content by means of metadata, lyrics, and content-based data. This functionality is realized through the usage of HTML, CSS, DOM, and JavaScript, that the Web interface is based on. Extended functionality is realized through the usage of Java applets running in the context of the Web interface.[6] The used technologies meet the minimum requirement to get the concept work by enabling the client to bidirectionally communicate between HTML/JavaScript code that is directly rendered by the Web browser, and Java applets including special query interfaces and the multimodal music player, that are executed by a Java virtual machine.

[6] The use of the extended functionality requires the Java Web browser plug-in to be installed on target machines.

The reason for using a mixture of technologies (Web and applet technologies) is discussed in the following. On the one hand, the repository client is intended to be used with standard Web browsers with a minimum set on technical prerequisites. Nevertheless, the exclusive use of standard Web technologies (HTML, CSS, and JavaScript) entails some serious limitations. The first issue concerns the connection between the browser and the Web service, which uses the HTTP/SOAP that is not connection oriented. This results in added overhead and potential security problems, as the communication state must be passed back and forth between the Web service and the Web browser. The second issue concerns the responsiveness regarding user interaction. Due to network latencies and the nature of scripting languages, the Web interface—especially the multimodal music player—is far not as responsive as OpenGL-supporting applets, because every time an interaction is made by the user, a new request needs to be generated, transmitted to the repository, and processed, whereupon a response is generated which is sent back to the browser, where it is used to update the client state.

4.5.2 Infrastructure

The music DL system architecture is a distributed architecture, consisting of loosely coupled subsystems. In addition, the subsystems are modular and both internal and external components interact with each other through stable, abstract interfaces hiding implementation details. As a consequence, this means that individual components can be seamlessly replaced by other implementations on different abstraction levels without side effects. The system is based on a SOA, where modules (agents) communicate with each other over the Internet through well-defined, publicly available communication interfaces using the SOAP.

The core subsystems are the repository server, the Web service, the application server hosting the Web service, the Web interface containing applets, and the DMS along with MACAO. Further subsystems are the MySQL server housing metadata and the file server storing primary and secondary data. The application server manages and hosts the Web service and provides requesting clients with the Web interface for service utilization.

4.5.2.1 System layers

The functionalities presented in Subsection 4.1.3.1 that are featured by the music DL are realized and implemented within a layered system as depicted in Figure 4.3. At the top-level view, the architecture breaks down in three physically separated layers that are intended to be operated on different, network-connected locations. This classical three-tier architecture consists of the *data layer*, the *presentation layer* and the *services layer*, from which the services layer will be examined in detail in the remainder of this chapter. Further details on the presentation layer from a user-centric view are to be found in Chapter 6.

Search indexes, annotations, and linking structures between different modalities are obtained in a preprocessing step which is carried out offline in the data layer. The access to index structures and synchronization data, as well as the delivery of musical content to the user, takes place in the services layer. The presentation layer consists of UI 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. Further details of individual layers are explained in the following.

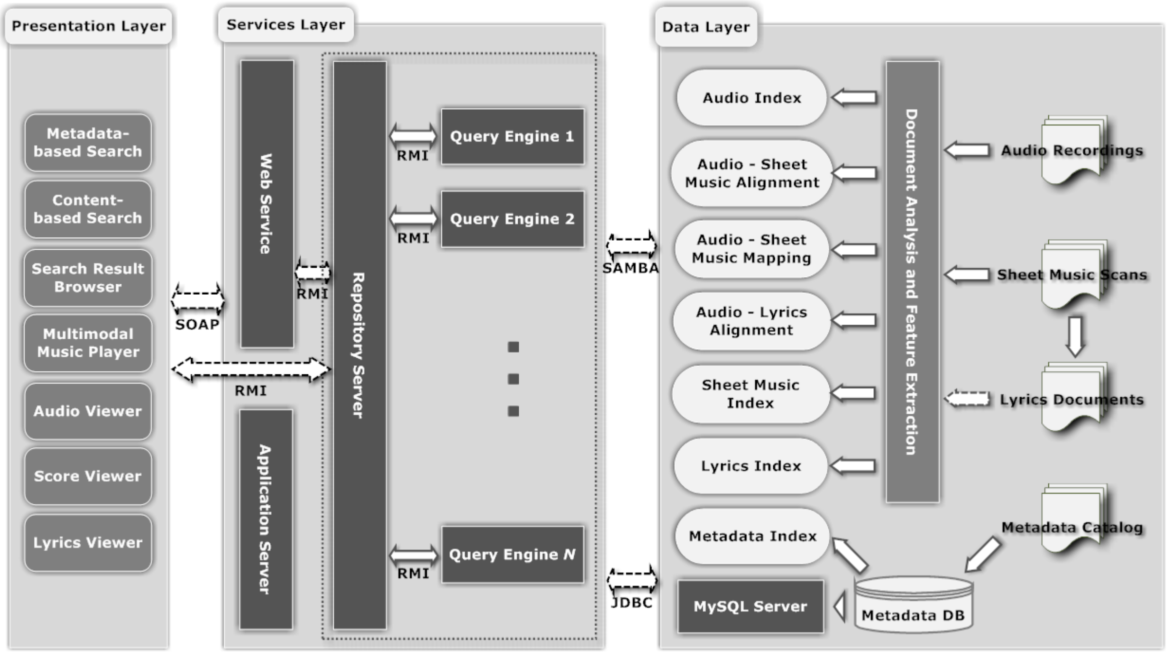


Figure 4.3 Overview of the music DL system architecture.

Data layer: centralized preprocessing and storage This layer consists of repository contents at the *backend* of the system, including primary and secondary data, and associated metadata. Furthermore, the layer comprises management tools for automatically processing, mapping, and aligning several types of data.

Presentation layer: web-based user–data interaction This layer comprises various UI components at the *frontend* of the system, by which end-users interact with repository contents. In particular, the layer comprises UI components for multimodal music access and interaction, including search, presentation, navigation, and browsing. Central functionality of the Web interface is the retrieval of music documents. Here, several kinds of metadata- and content-based searches can be employed for querying. Retrieved results are presented to the user in the result list, which mainly provides functionalities for viewing, browsing, and choosing musical works for in-depth view, as well as for query reformulation and refinement.

Services layer: service-oriented, distributed runtime environment This central layer connects the data layer with the presentation layer by using a *middleware* which transparently organizes the transport of data (messaging), brokers function calls between remote components (RPCs), and provides transaction reliability. First, the services layer has direct access to all repository data contained and generated in the data layer. Second, this layer is concerned with receiving and handling various requests from the presentation layer. The core components of the services layer are the application server and the repository server. The task of these servers is to handle communication with the presentation layer and to schedule incoming user queries to a set of available query engines, each of which offers a particular, dedicated functionality (e. g., content-based retrieval using audio matching or delivery of data via streaming).

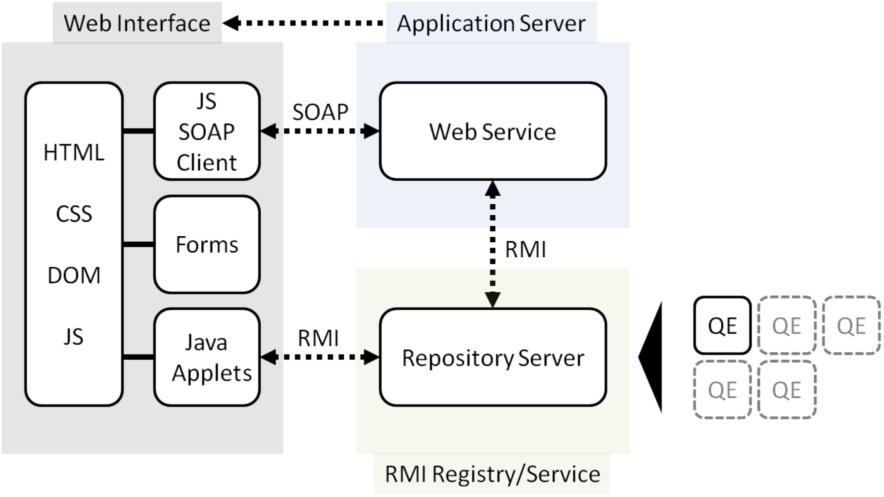


Figure 4.4 Functional interaction between frontend and middleware components.

4.5.2.2 Web application components, interfaces, communication infrastructure, and protocols

The Web application is driven by particular frontend and middleware components, whose functional interaction is illustrated in Figure 4.4.

Technical base From a technical perspective, all interaction with the repository from the outside, i. e., communication between the presentation layer and the services layer, is provided by a SOA [95, 96, 61] using Web services [81, 15] standards that have gained broad industry acceptance, and utilizes both the SOAP and the Java RMI protocol, collectively referred to as Repository Data Interchange Protocol (RDIP). SOAP is a network protocol for remote procedure calls (RPCs) and is used for the implementation of Web services. Web services provide a standard means of interoperating between different software applications, running on a variety of platforms and/or frameworks. For data interchange, human-readable Extensible Markup Language (XML) [17] messages are passed between instances on different machines using the Hypertext Transfer Protocol (HTTP) [43] over the Transmission Communication Protocol (TCP) [108]/Internet Protocol (IP) [107].

Access The RDIP is mainly used for searching and browsing purposes supporting both manual access for direct utilization through the Web interface (cf. Section 4.5.4.1) and mechanical access. The latter allows for the provision of open-access services that can be seamlessly integrated as closed building block in greater, embedded system architectures. Java RMI is a Java-based technology for network interactions between remotely distributed Java objects. For data exchange between the objects, there exists a name service, the Java RMI registry, which provides remote access to public methods of registered objects, and that is called like local methods by the RPC mechanism provided by the Java RMI protocol. This protocol is used, besides other, for the downstreaming of musical content and supports access security mechanisms.

Interfaces The RDIP provides a number of fundamental operations such as doQuery and getStream, which provide metadata- and content-based retrieval given a user query and the downstreaming of content, respectively. It provides clearly defined, open interfaces for the repository that allow third parties to write clients and higher level interfaces. The internal organization of the repository is not made public and can therefore be changed transparently to the client. This means that internal modifications do not affect the way clients interact with the repository, thus clients using the repository do not need to be modified. This is achieved by using stable communication interfaces that are implemented by the Web service and the Java RMI registry.

Security Security is an integral part of the system architecture and the RDIP. Rights and permissions can be individually configured on several levels. Besides public access to the services which enables anonymous clients to utilize baseline-level services, access to higher-level services is restricted to authenticated clients that either provide login credentials or have particular IP addresses. The usage of signed applets verifies that the applets come from a reliable source and can be trusted to run. If applets have the wrong or no signature, access to services will not be granted. Cross-site interfacing with applets is prevented by requiring both Web interface and applets to be hosted at the same location and run within the same client context in order to function properly.

4.5.3 Repository structure

4.5.3.1 Backend: persistent data storage

Primary and secondary data in the repository is held in the persistent storage. For the persistent storage, a MySQL database and a file server are used at the backend. The implementation of the persistent storage is completely hidden from the outside and the Web service gives a common access protocol regardless of the concrete implementation of physical data access.

4.5.3.2 Middleware: service runtime environment

The Web application business logic for the retrieval and dissemination of preprocessed data predominantly takes place in the middleware. Particular involved middleware components are considered in more detail in the following.

Application server For the implementation of Web applications, a so-called application server is needed. The application server takes the server role of the Web application. The application server hosts the deployed Web service and is publicly reachable through an Internet address. The interchange of data between Web interface and application server takes place by sending and receiving XML messages using HTTP and SOAP as transport mechanisms.

Web service The Web service is the part of the architecture that interfaces with the outside world by following the request/response paradigm. It is mainly intended to act as the gateway between the outside world and middleware-/backend-internal components and resources. The Web service implements the SOAP and converts between internal and external forms of user-specified search queries, retrieval results, and other data structures. It provides and manages user sessions for caching client state, search results, and other temporary data. The Web service is implemented in Java, where provided methods are defined in the WSDL and made available by a contract-first approach. XML-formatted messages and Java objects are cast back and forth. More precisely, the Web service accepts incoming XML-formatted messages, extracts contained method names along with associated parameters, casts the data in Java-complying data structures, and finally calls appropriate Java methods with according parameters. After finishing the execution of called methods, results are cast back into XML-formatted messages as responses.

Repository server The repository server provides the mapping between digital objects and their physical locations, and the system services required for reliable operation. The repository server provides a Java RMI-based interface between the services provided by the persistent storage and the functions required by the Web service.

The architecture has been designed to be general. In order to support many scenarios the repository server has been partitioned into core and extension modules. The core modules provide a minimum set of architectural building blocks and functionality. The extension modules allow for more specific or individual functionality. For each particular system interaction such as querying and retrieving search results as well as accessing musical content, a dedicated module is responsible, referred to as *query engine*. Query engines are dynamically loaded into and operated within the service runtime environment, an evolved redesign of the server component of the SyncPlayer framework [78, 45, 30]. Examples for query engines are search engines (metadata- and content-based) and streaming engines.

Database server The metadata are stored in a MySQL database and accessed through metadata engines.

File server The primary and secondary data are stored in a UNIX file system, accessed by the service runtime environment core over Samba.

4.5.4 User interfaces

4.5.4.1 Frontend: repository clients

The repository client is used to access repository content with any current Internet browser. It provides several techniques for complex browsing and handling of repository content. As reference client, it supports all RDIP methods implemented for the repository. It provides forms, mechanisms, and views for searching, browsing, retrieving, and interacting with musical content. The repository client is implemented as set of client services implemented in Hypertext Markup Language (HTML), JavaScript, and Java (cf. Figure 4.4).

Components The repository client is used to interact with the repository. It consists of two main components, the Web interface and the multimodal music player. Furthermore, the repository client concerns two areas: operational user interfaces for human–computer interaction and technical client services for communication with the repository.

Web interface The Web interface layouts query forms and result views, and it embeds the applets. It directly communicates with the Web service using the SOAP and is responsible for managing connections and user requests, and for maintaining a session with the repository. It indirectly communicates with the repository server over the Web service using Web interface scripts. The Web interface scripts are implemented in HTML, Cascading Style Sheets (CSS), and JavaScript.

Applet The applet communicates directly with the repository server using Java RMI and manages downstream and actual interaction with musical content. It communicates directly with the Java RMI service/registry using the Java RMI client library. The applet is implemented in Java in order to run as a browser-embedded cross-platform Java applet, downloaded on demand over the JNLP.

Services The client services can be separated into two different functional sets, namely JavaScript scripts and a Java RMI proxy. The first set is embedded within the Web interface scripts (JavaScript) and is responsible for communicating remotely with the application server (AJAX) and locally with the embedded applet (JSObject). The second set is independent of the browser and is responsible for establishing connections to the Java RMI service/registry using the Java RMI protocol. This architecture provides considerable flexibility since the JavaScript Web service caller function (belonging to the first set) can be interfaced or intercepted by third party user interface scripts. This feature allows for future repository clients to have multiple choices of interfaces to communicate with the repository. The supported interfaces to the repository are Web services over HTTP/SOAP (port 8080) and Java RMI over TCP/IP sockets (ports 8081 and 8082).

4.5.4.2 Backend: repository management tools

The DMS provides all the features required by content-managing persons and system administrators to manage music content. The UI is implemented as stand-alone Java desktop application, referred to as MACAO, and delivers a set of administrative tools and services providing mechanisms to create, index, interrelate, synchronize, and maintain musical content and corresponding metadata.

Chapter 5

Cross-modal music information retrieval techniques

In this chapter, a detailed description of the applied techniques used in the stages of the document processing chain is given, including feature extraction, audio indexing, as well as music identification and synchronization. For an overview, we refer to Figure 5.1. One of the key contributions of this thesis is the holistically multimodal querying–retrieval concept. The concept provides consistent integration of multimodality into every stage of the querying–retrieval chain. This chapter describes underlying preprocessing, indexing, and retrieval techniques. In order to compare and relate music data of various types and formats, the objective is to establish inner- and cross-modal linking structures that reveal the semantic correspondences of musical events in the various data streams. The extraction of small local information fragments embodying musical entities serves as a basis for an automated organization, semantic cross-linking of, and searching in digital music documents. To this end, the idea is to transform the various music representations into a common feature representation that allows for a direct comparison of the different types of data. Feature sets are extracted from the music documents on whose basis an objective, short-term comparison of the underlying musical information content is to be completed. In this context, chroma-based music features have turned out to be a powerful mid-level representation [6, 62, 88], which will be introduced in Subsection 5.2.1. In particular, we show how these features can be obtained from audio recordings using digital signal processing methods, as well as from scanned sheet music using OMR.

The features extracted from the audio documents are further processed and suitably organized by means of an inverted file index structure (Subsection 5.3.4). This audio index can then be used for both (a) identifying and annotating individual pages of scanned sheet music by means of available annotated audio material (Subsection 5.2.2), as well as (b) content-based music retrieval (Chapter 6). The identification task for sheet music consists of assigning each scanned sheet music page to a particular audio recording, as illustrated in Figure 5.2. For each audio recording, we group the corresponding pages of sheet music to establish a global correspondence between the audio and the sheet music data on the level of individual tracks, i. e., songs or movements. Finally, using the mid-level chroma representation and DTW, the two representations are synchronized, which results in a linking structure that links the visual with the acoustic domain (Subsection 5.2.3), see also Figure 5.3. This structure lays the

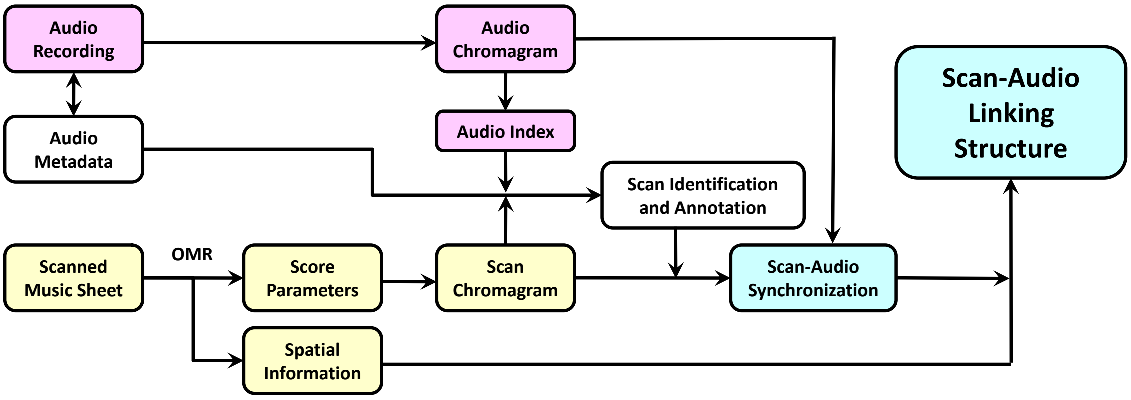


Figure 5.1 Overview of the workflow for automatic cross-modal document processing. Two different modalities are considered, which concern the visual (scanned sheet music) and the acoustic (audio recordings) domain. Adapted from [34].

foundation for a time-synchronous presentation of sheet music and audio recordings by means of the multimodal music player (Section 6.4).

5.1 Extraction of musical and textual entities from sheet music

As the scans of sheet music are available as images, the score information is "hidden" in the sense that graphical elements need to be recognized and interpreted in order to achieve a machine-readable symbolic representation of the sheet music contents. Before the sheet music data can be further processed, it has to be transformed from the image domain to the symbolic domain. This is done with the help of OMR, a key technology to extract symbolic information from these images. In principle, OMR works similar to OCR, but is more involved both from a computational and a pattern recognition point of view. Primarily, all the musical symbols that build up the scores are extracted. Besides, also syllables belonging to lyrics are one of the recognized data fragments.

5.1.1 Symbolic score extraction

OMR processing is a key component in document processing. It serves as a chain link in the document processing chain and is applied to each scanned page of sheet music in order to get from its pictorial representation to a symbolic representation that is initially required for further processing. Subsequent semantic analysis on related symbolic content of consecutive sheet music pages yields higher-level semantics such as expression boundaries within the underlying manifestation. It also yields other information required for the purpose of deriving a MIDI-like representation of that expression. The latter type of representation, mainly consisting of note onset times, durations, and pitches, is then subsequently used for two purposes. First, it is used for score-based indexing and retrieval. Second, it is used for mapping and synchronization of interrelated scores and performances. Note that although symbolic score data created through OMR is used in the process of indexing and organizing content, but also for calculating synchronizations, it is never directly presented to the user—it serves exclusively for the generation of secondary data.

5.1.2 MIDI-like note extraction

For the purpose of a score-based retrieval, the symbolic score is transformed to a MIDI like representation. Here, a subset of the entire score content is extracted consisting of a set of note events. A note event is a triple $[p, t, d]$ with a *pitch* p, an *onset time* t, and a *duration* d. The pitch p is in the range of 88 semitones of the well-tempered piano, the onset time is a 64-th of a measure that consists of either one whole, 4 quarters, sixteen 16-th, or sixty-four 64-th notes.

5.1.3 Lyrics extraction

The OMR process does not only provide the recognition and extraction of musical symbols. If available, also lyrics are recognized and extracted along with their respective coordinates. This additional functionality is exploited for two purposes: the display of spatial location within the scanned image and interaction with the lyrics in a karaoke-like fashion (as another view within the multimodal music player) and the provision of a full-text index-based search in lyrics. In order to feature a fast, fault-tolerant, and word-based search, on the one hand, a search index has to be built up, and, on the other hand, individual extracted syllables are to be assembled to whole words. For the syllable extraction and word-assembling process, all MRO files in the repository are to be analyzed.

Due to the cross-linkage between image regions of score scans with time segments of associated audio recordings belonging to the same work yielded by the synchronization process, not only spatial information is available, where syllables are located within the score scans, but also timing information is available specifying at which time a certain syllable is sung.

Exploiting further the detected repetition structure of the scores containing the lyrics, an “unfolding” of the horizontally arranged lyrics lines can be achieved. That is, the lyrics lines or words are arranged successively in their correct order as they arise in the timeline of the musical work. This process, referred to as linearization or unfolding of the lyrics as sequence of words, is needed to feed the lyrics view of the multimodal music player in order to play back the lyrics or individual words time-synchronously to their occurrence in an audio recording. Note that this is not required for the purpose of indexing or searching because the correct sheet music page and position therein can still be determined without unfolding. However, without unfolding, in the retrieval scenario the visual arrangement or placement of the occurrences of a query on the timeline bar of the multimodal music player could not be possible. For different performances of the same work, there are potentially different timings for syllables and words which may be of special interest to be highlighted to the user on the timeline bar of the multimodal music player.

In order to get the onset times of lyrics, the following approach is applied. The sheet music books are processed page by page on the level of measures. For each of the measures, individual syllables are extracted. For these syllables the respective onset and offset positions within their measures are known. In conjunction with the synchronization of the measure to an appropriate segment of an audio recording, the corresponding syllables’ time positions within the audio recording, in turn, can be estimated heuristically. Individual syllables are merged into whole words, each of which is associated with both a surrounding bounding box or region within the sheet music page and a time segment within the audio recording. The merging of individual syllables into whole words is based on the assumptions that a word’s prefix and infix syllables are hyphenated It turns out that almost all hyphens are recognized and extracted by the OMR process. So, this approach is a practical solution for our purposes. Subsequently, the extracted words are matched against a dictionary in order to automatically detect and correct wrongly extracted letters and thus words. To further improve the extraction accurracy, this process may be combined with OCR techniques. In this scenario, the pre-calculated bounding boxes are utilized for an OCR-based extraction of the contained syllables. As sophisticated OCR systems focus on the reliable extraction of text, it is likely that they can be used to improve the lyrics recognition results over the OCR algorithms included in OMR systems. However, due to the incorporation of fault tolerance mechanisms, this is less critical for the purpose of searching than for the proper display of the lyrics in the lyrics view of the multimodal music player. It remains to mention that the extracted time information is additionally stored in the lyrics index in order to extend the searching capabilities, see Subsection 5.3.2.

5.1.3.1 Utilization of external lyrics text files

In case of availability of orthographically correct external lyrics text files, these can be exploited for the annotation as well as enriched with timing information from the error-prone lyrics extract gained by the OMR process. On the one hand, the error-prone lyrics extract contains flawed words along with corresponding timings of when each word is sung in a particular audio recording. On the other hand, associated orthographically correct text files do not provide these timings, but are flawless. These two kinds of information sources can be combined so that the timings of the flawed lyrics are transferred to the flawless lyrics. For this purpose, a DTW-based approach is used to align the flawed word sequence and the orthographically correct lyrics word by word. In this process, partially flawed words are matched and aligned against their orthographically correct pendants, whereupon individual, associated timing information is transferred. The method of utilizing external lyrics that are available as text files finally leads to high-quality lyrics with automatically calculated timings.

5.2 Cross-linking of semantically interrelated entities

Given various representations of musically relevant information, e. g., as encoded by sheets of music or as given by a specific audio recording, the identification of semantically interrelated events is of great relevance for music retrieval and browsing applications. Here, we will discuss the problem of score–audio synchronization, which refers to the problem of finding fine-grained linking structures that spatio-temporally align sheet music and audio recordings based on their musical content. Such linking structures can then be used to, e. g., highlight the current position in scanned sheet music during the playback of a corresponding audio recording, or navigate in the audio recording by means of selecting a certain measure within the scanned sheet music. This not only enhances the listening experience, but also provides the user with tools for intuitive and multimodal music exploration.

5.2.1 Common mid-level feature representation

To make the various music representations comparable, one needs to find a suitable common mid-level feature representation that satisfies several critical requirements. On the one hand, such a feature representation has to be robust to semantic variations as well as to transformation errors. Furthermore, the various types of data should be reducible to the same mid-level representation. On the other hand, the features have to be characteristic enough to capture distinctive musical aspects of the underlying piece of music. In the context of matching and synchronization of music, *chroma-based music features* have turned out to constitute a good trade off as they do achieve those—to some extent conflicting—requirements to a high degree for certain classes of music, especially for the case of 12-tonal Western classical music. Here, the twelve *chroma* correspond to the twelve traditional pitch classes of the equal-tempered scale [6]. In Western music notation, the chroma are commonly indicated by C, C♯, . . ., B consisting of the twelve pitch spelling attributes. Chroma-based features are well-known to reflect the phenomenon that human perception of pitch is periodic in the sense that two pitches are perceived as similar by the human auditory system if they differ by one or more octaves [6].

For an audio recording, the digitized signal is transformed into a sequence of normalized 12-dimensional chroma vectors, where each vector reveals the local, normalized energy distribution among the twelve pitch classes. Hence, the chroma sequence closely correlates to the harmonic progression of the underlying piece of music. Based on signal processing techniques, a chroma representation can be obtained by using either short-time Fourier analysis in combination with binning strategies [6] or multirate filter bank techniques [88]. Such a representation, in the following also referred to as *audio chromagram*, absorbs variations in parameters such as dynamics, timbre, and articulation, and closely correlates to the rough harmonic progression over time of the underlying audio signal. Figure 5.3 (c) shows an audio chromagram for the first few measures of an audio recording of the third movement of Beethoven's Piano Sonata Op. 13 ("Pathethique").

The transition from a sheet music representation to a chroma representation consists of several steps. In the first step, musical score symbols such as notes including onset times, pitches, durations, clefs, key signatures, and time signatures are extracted using OMR, see [18, 23]. This process is similar to the well-known OCR, where textual information content is extracted from an image scan of a text document. In the context of the music DL framework, the commercially available OMR software package SharpEye Music Reader 2.68 [68] is used to extract the musical score symbols from sheet music scans. Note that the OMR extraction step is error-prone and the recognition accuracy strongly depends on both the quality of the input image data and the complexity of the underlying score. In the context of the music DL system, high-quality scans of sheet music at a resolution of 600 dots per inch (DPI) and 1-bit color depth (black/white) are considered. In addition to the musical score symbols, the OMR process also provides spatial information. In particular, the exact 2-dimensional position parameters, i. e., pixel coordinates, of the extracted notes as well as bar line information are available from the scanned images. This allows for localizing all extracted musical symbols within the sheet music.

In the second step, based on the OMR output, a sequence of normalized 12-dimensional chroma vectors is derived, which is also referred to as *scan chromagram*. To this end, note events specified by musical onset times, pitches, and note durations are created from the extracted musical symbols. Assuming a constant tempo of 100 beats per minute (BPM), the explicit pitch and timing information can be used to derive a chromagram essentially by identifying pitches that belong to the same chroma class. To this end, a temporal window is slid across the time axis while adding energy to the chroma bands that correspond to pitches that are active during the current temporal window. Here, a single temporal window equals a single

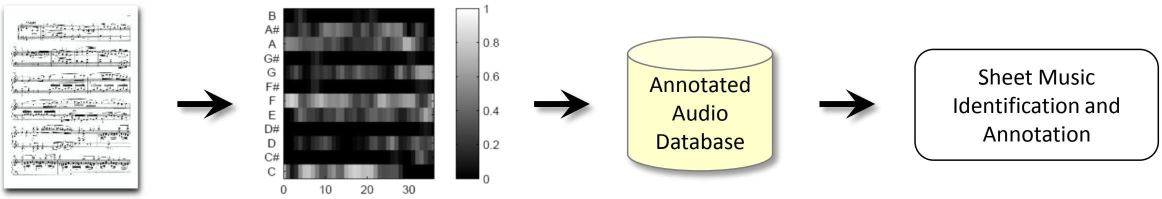


Figure 5.2 Overview of the matching procedure for automatic identification and annotation of scanned sheet music using an annotated audio database. The first page of the second movement of Beethoven's piano sonata Op. 2 No. 1 and the resulting scan chromagram are shown. Adapted from [34].

chroma vector. A similar approach has been proposed in [62] for transforming MIDI data into a chroma representation. Note that the particular choice of 100 BPM in our assumption is not an essential restriction, because differences in tempo will be compensated in the subsequent matching and synchronization steps. Figure 5.3 (b) shows a scan chromagram obtained from (a) a sheet music representation for an excerpt of our "Pathethique" example. Note that a scan chromagram is obviously, in general, much "cleaner" than an audio chromagram as it is derived from relatively few, discrete events, cf. Figure 5.3. However, the OMR software often produces serious note extraction errors, which are only partially absorbed by the chroma features. Depending on the complexity of the underlying score, for the majority of sheet music scans of pieces of music considered in the context of this thesis it turns out that the OMR quality or accuracy is sufficient to obtain reasonable matching and synchronization results in the subsequent processing stages (cf. also the closing discussion of Subsection 5.2.3).

Both the identification of scanned sheet music pages (Subsection 5.2.2) and content-based audio retrieval (Subsection 5.3.4) rely on a mechanism for efficient *audio matching* [77]. Here, given a short query music clip in form of an excerpt taken from an audio recording or in form of some bars of music taken from scanned sheet music, the goal is to automatically retrieve all excerpts that musically correspond to the query from an audio database. As opposed to classical audio identification [1], audio matching allows for semantically motivated variations as they typically occur in different interpretations of a piece of music. The methods for audio matching introduced in [77] work on the basis of chroma representations. As it has been shown recently, the chroma features generated from symbolic music representations, e. g., those obtained by the above OMR process, are compatible with audio chromagrams. Therefore, chroma features can be used to perform both audio matching [79] and synchronization of music documents both within and *across* the domains of symbolic music and audio recordings [51].

5.2.2 Identification and annotation of scanned sheet music

After the digitization process, the digitized documents need to be suitably annotated before they can be integrated into the holding of a digital library. In the case of digitized audio recordings, one has to assign metadata such as title, interpreting artist, and performers to each individual recording. Besides the labor- and cost-intensive option of manual annotation, one may exploit several available databases that specialize on various types of metadata such as Gracenote [56] or DE-PARCON [72]. Note that in spite of the existence of such databases, it may *not* in general be assumed that the acquisition of metadata is a trivial task because existing databases are frequently incomplete w. r. t. old recordings, they lack particular types of requested metadata, or contain errors and inconsistencies. An improvement may be achieved by, e. g., exploiting and merging multiple data sources. However, this is out of the scope of this work and we rely on the availability of metadata of sufficiently high quality. This is not a serious restriction as an improvement of both quantity and quality of metadata is to be expected over time due to the fact that the supply of high-quality metadata is the business concept of some commercial service providers (e. g., Shazam [132]). Ultimately, the libraries usually have high-quality metadata and are responsible for the quality assurance of them. For the purpose of this thesis, we assume that suitable annotations for the audio recordings are readily available, cf. Figure 5.1.

After the digitization of scanned sheet music—a process that can be done by scan robots—each page has to be annotated separately. This annotation is usually done in a manual process. Following [51], we now describe how this annotation process can be performed automatically, see also Figure 5.2. In our scenario, we assume the existence of an audio database containing annotated audio recordings for all pieces of music to be considered in the sheet music digitization process. In a preprocessing step, we transform the audio documents into corresponding audio chromagrams and build up an audio index structure. Then, in the annotation step, each scanned page of the sheet music is converted into a separate scan chromagram. Using each scan chromagram as query, we compute the top match within the audio documents as described in Subsection 5.3.4. Here, we assume that each page is fully contained in a single audio document. Note that this assumption does not generally hold, since a single page may refer to several short pieces or may contain the end and the start of two consecutive movements corresponding to different audio documents. Under our assumption, the top match usually identifies the musically corresponding audio recording with high probability. As our experiments show, this holds with an even higher probability in case that there are no severe OMR errors. Upon identification, the scanned page can then be automatically annotated by the metadata already attached to the corresponding audio recording, see Figure 5.2. Furthermore, the first few top matches usually consist of all passages within the audio recordings that musically correspond to the page. This additional property is exploited for the retrieval and browsing applications as described in Chapter 6.

In order to establish a correspondence between sheet music pages and audio recordings on the level of individual tracks, content-based comparison is used. This can be realized using various strategies. Firstly, the score is searched for indented grand staffs. Such indentations usually indicate the beginning of a new movement or musical work. Using this information, the scan chromagrams created from pages including such an indentation can be divided at the beginning of the indented grand staff to account for the expected track change. Secondly, title headings that have been recognized in the scores may be used as indicators for the beginning of movements and musical sections as well. Furthermore, the recognized text of these headings can be compared to the known titles of the tracks in the audio database.

Using suitable heuristics, some of the OMR extraction errors can be corrected in a post-processing step prior to the matching step. For example, in the case of piano music, different key signatures for the left and right hand lines in a grand staff can be assumed to be invalid and easily corrected by considering neighboring staff lines. Furthermore, similar to the strategy suggested in [18], one can simultaneously employ various OMR extraction results obtained from different OMR software packages to stabilize the matching result. First experiments show that, based on these strategies, a significant improvement of the identification rates can be achieved [46].

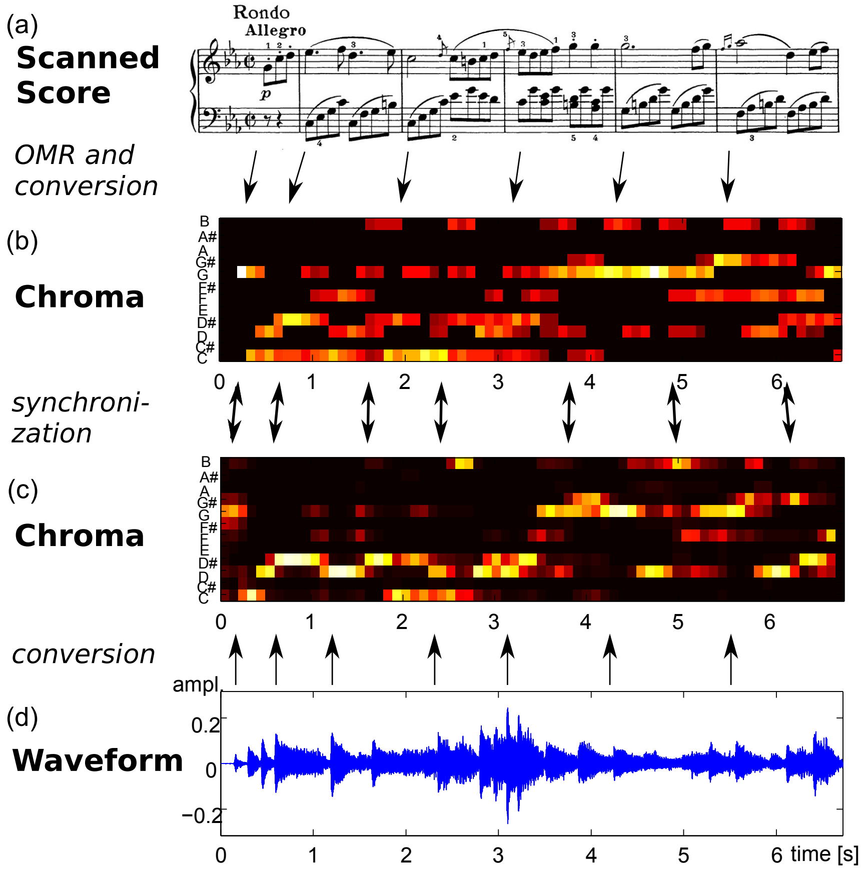


Figure 5.3 Data types involved in automatic document processing for the first few measures of Beethoven's Piano Sonata Op. 13 ("Pathethique"), Rondo (third movement): (a) scanned sheet music, (b) scan chromagram, (c) audio chromagram and (d) audio recording (waveform). The scan–audio linking structure (double-headed arrows) is obtained by synchronizing the two chromagrams. Adapted from [34].

5.2.3 Scan–audio synchronization

Once having identified scanned pages of sheet music and corresponding audio recordings, we automatically link semantically interrelated note events across the two types of musical expressions. In the general context, various alignment and synchronization procedures have been proposed with the common goal of automatically linking multiple types of music representations, thus aligning the multiple information sources related to a given musical work [2, 62, 88, 93, 99, 120, 127]. In our specific scenario, the problem is referred to as score–audio synchronization, where the objective is to link regions (given as pixel coordinates) within the scanned images of given sheet music to semantically corresponding time segments within an audio recording. Such a procedure has been described in [80].

The basic idea is to convert both the scanned sheet music of a given piece and the corresponding audio recording into chromagrams. In case of the scanned pages, we utilize the identification results described in Subsection 5.2.2 to construct the chromagram for the underlying sheet music representation by appropriately grouping the features obtained from the individual pages.

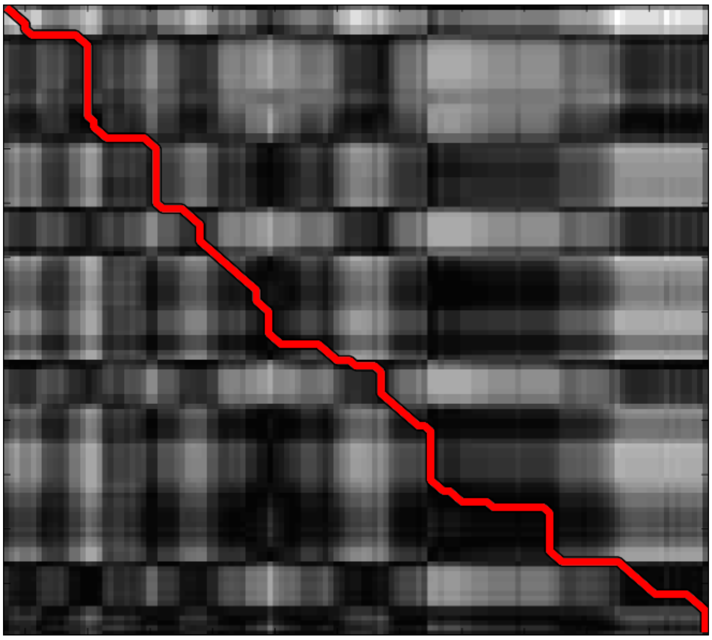


Figure 5.4 Cross-similarity matrix and optimal alignment path for feature sequences.

The resulting scan- and audio-chromagrams are then synchronized based on standard alignment techniques such as DTW [88]. Here, one builds up a cost matrix by computing the pairwise distances between each scan chroma vector and each audio chroma vector. More precisely, denoting the feature sequence derived from the sheet music by $v := (v_1, v_2, \ldots, v_N)$ and that of the audio file by $w := (w_1, w_2, \ldots, w_M)$, one builds an $N \times M$ cross-similarity matrix by calculating a similarity value for each pair of features (v_n, w_m), $1 \leq n \leq N$, $1 \leq m \leq M$ (cf. Figure 5.4). Here, one needs a suitable local distance measure for determining the distance between two chroma vectors. In our implementation, we use the cosine measure between normalized chroma vectors. Then, an optimum-cost alignment path is efficiently determined from this matrix via dynamic programming. In order to handle global tuning shifts in the audio recordings, *chroma cyclic shifting* is performed, see [90] for details. The resulting optimal path through the matrix encodes a temporal alignment of the two chroma sequences. For details on DTW, we refer to the literature [88]. Now, the spatial information of the OMR output allows for assigning each scan chroma vector to a corresponding region within the scanned sheet music image. Combining this spatial information with the score–audio synchronization result, one can subsequently derive a linking structure between the scanned images and the audio recording. The importance of such linking structures has been emphasized in the literature [40]. Note that while Figure 5.3 suggests a spatio-temporal alignment on the note-level, the chroma features that are currently used, usually do not yield such a high time resolution. To avoid possible corresponding visualization errors in the multimodal music player presented in Section 6.4, a coarser alignment resolution on the granularity level of musical *measures* is used. For mapping the resulting scan–audio alignment to a resolution level of measures, the symbolic information is used on the location of measures gained from the OMR step. An example of the discussed score–audio synchronization is shown in Figure 5.3, where the resulting linking structure is indicated by the double-headed arrows. In Chapter 6, UI components are introduced that exploit the scan–audio alignment in order to facilitate cross-modal music presentation and navigation.

We conclude this section by discussing some challenges that arise in the music synchronization context and give rise to future research. Local differences in content between audio and score representations may have a negative influence on the final synchronization result. As mentioned in [49], the quality of the resulting synchronization depends on several factors. One of these factors are structural differences such as missing or additional repetitions of musical sections. For example, the score might contain a section that is not played in the audio recording or the audio recording might contain an extra repeat that is either not present or not recognized in the score. Such differences in structure may be caused by OMR errors or stem from the fact that a performance is not required to strictly follow the structure suggested by the musical score. Furthermore, for a given audio recording it is not guaranteed that the contained performance is actually based on the particular score edition that it is to be synchronized with. Differences in structure can violate the boundary or monotonicity assumptions made in DTW, see [88]. Such differences may be handled in a manual preprocessing step or by partial matching strategies [89]. In [50], a novel variant of DTW is introduced, referred to as JumpDTW, which allows jumps and repeats in the alignment and significantly improves synchronization results. This approach has also been incorporated in the music DL framework. However, the general problem of handling partial similarities between music representations to be synchronized still poses many open research problems.

Further dissimilarities of more local nature are musical events in the audio and sheet music representations with deviating pitch or duration. Problematic are also note ambiguities in the score such as arpeggios, trills, grace notes, or other ornaments. Generally, differences of this class tend to have little impact on the overall synchronization result as long as they stay local and are enclosed by sections without mismatches and errors. Significant differences in tempo may also cause problems in the synchronization procedure. Recall that for computing a mid-level representation from a symbolic music representation, one needs to decide on the tempo to be used in the transformation from musical onset times like beats and bars to physical onset times measured in seconds and milliseconds. Since tempo directives of music notation are often not output by OMR systems (as it is also the case with SharpEye), the tempo then has to be estimated. For Western classical music, the tempo can vary over a wide range from about 25 to 200 BPM. Local differences between the estimated tempo and the actual tempo of the audio recording are usually compensated by a DTW-based alignment strategy. However, DTW starts to loose flexibility and accuracy when the tempo differences become too large. The limits for which an alignment is calculated reliably, are reached at a tempo deviation factor of approximately three [46]. For a detailed examination of such issues, we refer to [47, 46]. A beat tracker may help to estimate the tempo of a specific audio recording to be synchronized with in order to yield more accurate synchronization results. Currently, this approach is not applied.

Another challenge pose orchestral works when performed with transposing instruments. In such a scenario, the relative detunings of involved instruments lead to significantly altered chroma vectors. This, in turn, complicates the task of content-based retrieval as well as the alignment of that performances to sheet music or other performances where instruments are not transposed. This issue is approached in [124].

The practical impact of the issues discussed above are heavily data-dependent and unforeseeable in general. This holds especially in the case of great structural differences between the score and associated audio recordings, combined with other issues such as a high degree of polyphony as well as pitch and tempo deviations. In the worst case, synchronization results may become so degenerate that large parts of the calculated positions do not match the actual positions (in a musical sense). However, the testbed used in the context of the PROBADO music DL showed that serious errors affecting the synchronization accuracy occur only rarely for large parts of the processed data set. With an increasing degree of polyphony, a small degradation of the synchronization accuracy was experienced. While synchronization results for piano sonatas are virtually flawless, some of the more complex orchestral works need to be corrected manually before the synchronization is calculated. For this purpose, the DMS of the music DL framework supports the manual intervention and possible correction of errors at the corresponding stage in the document processing chain.

5.2.4 Lyrics–audio synchronization

Similar to the score–audio synchronization, where musical symbols or whole measures are assigned to musically corresponding time segments of audio recordings, the lyrics–audio synchronization is a word-based alignment of individual words mapped to both a 2-dimensional region within the sheet music where they have been extracted from, and to corresponding time segments of audio recordings. Note that usually there is more than one audio recording of a musical work.

In contrast to other approaches that try to align lyrics and audio recordings by means of audio signal processing methods, the approach in this thesis exploits the score–audio synchronization to extract individual word's onset times. The extracted words' respective localizations within the sheets of music are gained likewise by means of OMR processing.

5.3 Indexing and retrieval techniques

With the help of indexing and retrieval techniques, documents can be searched and found in large document collections. For this purpose, various search indexes are constructed from metadata and content extracted from all documents. In common, the index structures consist of sequences of small digests of document fragments embodying their essences, on whose basis matches can be detected between search queries and documents. As key function, a search may be performed based on multiple modalities. Here, multiple indexes are used in parallel for query processing, whereupon returned result lists are fused to a single modality-comprehensive result list.

5.3.1 Metadata-based retrieval

Metadata are stored in an SQL database according to the music DL metabase scheme. The SQL database consists of various tables from which the most important ones are work, expression_has_manifestation, and manifestation (cf. Figure 3.3).

For the simple full-text search in metadata, a search index of all relevant metadata contained in the SQL database records is created. In order to efficiently find both inherited and directly attached metadata associated to manifestations, for each work, all metadata from all levels are merged into a single character string that is indexed and associated to the work. Standard text retrieval techniques are used to retrieve all works contained in the search index whose manifestations relate somehow to a user's query.

For the advanced fielded search in metadata, no special search index is used, as this search can rely on the basic functionalities provided by the DBMS of MySQL. That is, instead of setting up a special search index, the internal database index is directly used. A fielded search in the metabase's underlying SQL database records is performed by means of SQL statements. Currently, the fielded search supports the explicit search of composer, title, opus, contributors, instrumentation, and creation date or period, where the latter is actually a range search (e. g., 1900 to 1920). It may be of interest to search for other dates or periods such as creation or publication dates, which are available for most manifestations. In principle, there are no restrictions to support more fields. The search is fault-tolerant in the SQL LIKE-sense, i. e., a queried string is found if it is a substring of another string stored in the metadata records. In case of searching by means of several fields, the total result consists of the intersection of all of the individual results. Note that search terms entered in a specific field are only searched in metadata records that correspond to that field.

5.3.2 Lyrics-based retrieval

In the following, the index-based search method for lyrics-based retrieval is described. The music DL system employs lyrics-based retrieval as described in [30, 92].

We assume that the lyrics for our collection of N audio recordings are stored in N text files $\mathcal{L} := (L_1, \ldots, L_N)$, where a file L_i consists of a sequence $(t_{i1}, \ldots, t_{in_i})$ of terms. The indexing technique uses inverted files which are well known from classical text retrieval [139]. In lyrics-based retrieval, users are likely to query catchy phrases as they frequently occur in the chorus or hook line of a song. Therefore, our basic indexing strategy presented next is designed to efficiently retrieve exact sequences of query terms. Later on, this basic strategy is extended to allow fault-tolerant retrieval.

In a preprocessing step, for each term t an inverted file $H_{\mathcal{L}}(t)$ is constructed from our text files. $H_{\mathcal{L}}(t)$ contains all pairs (i, p) such that t occurs as p-th lyrics term within text file L_i, i. e., $t_{ip} = t$. Using inverted files, query processing may then be performed simply by using intersections of inverted files. Assume a query is given as a sequence $q := (t_0, \ldots, t_k)$ of words. Then, with $H_{\mathcal{L}}(t) - j := \{(i, p - j) \mid (i, p) \in H_{\mathcal{L}}(t)\}$, the set of *matches*

$$H_{\mathcal{L}}(q) = \bigcap\nolimits_{j=0}^{k} H_{\mathcal{L}}(t_j) - j \tag{5.1}$$

can be easily shown to contain all pairs (i, p) such that the exact sequence of terms q occurs at position p within the i-th document. To make this basic matching procedure robust towards errors such as misspelled or wrong words, we introduce several methods for incorporating fault tolerance. To account for typing errors, we preprocess each query term t_j and determine the set T_j of all terms in our dictionary of inverted files having a small Levenshtein distance [83] to t_j. Then, instead of only considering the exact spelling t_j by using $H_{\mathcal{L}}(t_j)$ in 5.1, we consider the union $\cup_{t \in T_j} H_{\mathcal{L}}(t)$ of occurrences of all terms which are close to t_j w. r. t. their Levenshtein distance. To account for term-level errors such as inserted or omitted words, we first preprocess all word positions occurring in 5.1 by a suitable quantization. This amounts to replacing each of the inverted files $H_{\mathcal{L}}(t)$ by a new set $\lfloor H_{\mathcal{L}}(t)/Q \rfloor \cdot Q$, where each $(i, p) \in H_{\mathcal{L}}(t)$ is replaced by a quantized version $(i, \lfloor p/Q \rfloor \cdot Q)$ for a suitably chosen integer Q (currently, $Q = 5$ is used).

Furthermore, we replace $H_{\mathcal{L}}(t_j) - j$ of 5.1 by $H_{\mathcal{L}}(t_j) - \lfloor j/Q \rfloor \cdot Q$ prior to calculating the intersection. The latter yields a list $(m_1, \ldots, m_\ell)$ of matches which is subsequently ranked.

For each match m_i we obtain a ranking value r_i by combining classical ranking criteria (r_i^1, r_i^2, and r_i^3 in what follows) with criteria accounting for the peculiarities of the lyrics-based retrieval scenario (r_i^4 in what follows). As for the classical criteria, each match m_i is assigned a ranking value r_i^1 that essentially measures the deviation of the query terms occurring in m_i from their correct ordering as specified by q. To account for term-level mismatches, a ranking value r_i^2 counts the percentage of query terms occurring in m_i. Note that r_i^2 may be obtained efficiently by using a dynamic programming technique [25] while simultaneously calculating the set of matches (5.1). A further ranking value r_i^3 accounts for the total Levenshtein distance of the query terms to the terms matched in m_i. Exploiting the lyrics–audio synchronization corresponding to m_i, we obtain a ranking value r_i^4 by suitably weighting the temporal distance (within the audio recording) of the first and the last query term occurring in m_i. Finally, an overall ranking value for each match is obtained as $r_i := \sum_{j=1}^{4} w_j r_i^j$, where $w_1, \ldots, w_4$ denote some suitably chosen real-valued weighting factors.

Beyond these ranking criteria, it may be interesting to include even more music-specific knowledge into the ranking procedure. As an example, one might exploit available information about the structure of the audio recording to give lyrics terms more weight if they occur in structurally salient passages such as in the chorus sections.

The proposed methods for indexing and retrieval can be realized by properly adapting well-known text retrieval techniques. As described in [25], retrieval using the proposed inverted file-based approach can be performed very efficiently in both of the cases of exact and fault tolerant retrieval including the proposed ranking. The efficiency of the proposed methods has been tested on a database consisting of 120 000 lyrics files. Allowing a maximum term-mismatch rate of $\frac{1}{2}$ and a maximum Levenshtein distance rate of $\frac{1}{3}$ per term, it turns out to perform efficiently in a few tenths of a second.

To conclude this section, we note that in the above we have assumed that the alignment information for linking the lyrics to the audio recordings is stored in some suitable secondary data structure that can be accessed efficiently. In our implementation, we use an additional file format to store this information which turns out to perform sufficiently well.

5.3.3 Score-based retrieval

In the following, we describe the index-based search method for score-based retrieval. The music DL system employs score-based retrieval as described in [24, 26]. The appropriate retrieval engine aims at finding a queried melody in a collection of sheet music. Here, the user formulates a query consisting of notes building up a melody that is to be searched in the collection. A query is matched against an index that is built upon extracted MIDI notes yielded by a prior OMR process. The search result consists of a set of localizations within the document collection, where the query has been found.

It is assumed that the extracted MIDI notes[1] $[t, p] \in U = \mathbb{Z} \times [0 : 127]$ from the sheet music documents are stored in N score files $\mathcal{D} := (D_1, \ldots, D_N)$. A document $D_i \subset U$ consists of a set $([t_{i1}, p_{i1}], \ldots, [t_{in_i}, p_{in_i}])$ of notes, each of which consists of an onset time and a pitch. Note that durations are not taken into account, only a note's onset time and pitch are considered.

[1] The infinite set of all possible MIDI notes (the note universe) from which score files are composed of.

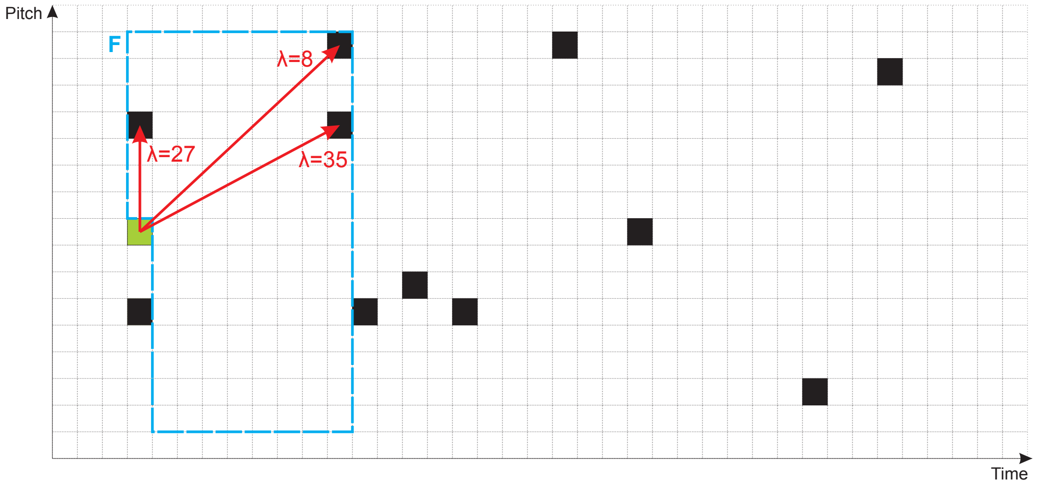


Figure 5.5 Illustration of legal note pairs. For a given note, legal note pairs are built from notes within the individual frame. An individual arrow shows the corresponding legal note pair index λ. As durations are not considered, all note lengths are equal to 1. Adapted from [26].

By neglecting the duration of notes, an unnecessary constraint is removed as particularly duration timings are likely to vary and are relatively unimportant in most cases. Without this convention, the set of matches would be too restricted and thus the usability of the search likewise reduced. The indexing technique uses inverted files which are well known from classical text retrieval [139] and is a specialization of a generic retrieval concept that is based on identifying object constellations in a general object universe [25]. Here, both score documents and queries are modeled as finite subsets of U. Finding note constellations should be invariant under time- and pitch-shifts. As in score-based retrieval, the user's queried melody or short score excerpts may be contained in the database $\mathcal{D}$ in other tunings, i. e., all query's notes are constantly shifted in pitch against note constellations contained in the database. Such a transposition-invariant retrieval is realized by considering note pairs instead of single, isolated notes, because pairs express the relative position among neighboring notes, regardless of their absolute pitch. Accordingly, both the documents and queries are likewise preprocessed prior to indexing or retrieval, respectively. The basic indexing strategy presented next is designed to efficiently retrieve exact score excerpts disregarding relative transpositions of queries against note constellations in the database. Later on, this basic strategy is extended to allow fault-tolerant retrieval.

In a preprocessing step, each legal note pair $([t,p],[t',p']) \subset D_i$ is assigned a note pair index $\lambda := \Delta(t'-t, p'-p), \lambda \in \mathbb{N}$, obtained from an injective partial mapping $\Delta : F \rightarrow \mathbb{N}$ that is well-defined for legal note pairs and undefined otherwise . Legal note pairs are defined within a certain neighborhood $F \subset U$ in time and pitch. In particular, two notes $[t,p]$ and $[t',p']$, $t \leq t'$, build a legal note pair if for suitable $T_0, P_0 \in \mathbb{Z}_{>0}$ either $p < p' \leq p + P_0$ for coinciding onset times $t = t'$ or $t < t' \leq t + T_0$ and $p - P_0 \leq p' \leq p + P_0$ for different onset times does hold. This defines the frame $F(P_0, T_0) = (\{0\} \times [1 : P_0]) \cup ([1 : T_0] \times [-P_0 : P_0])$ that spans a compact plane, wherein partner notes $[t',p']$ must reside for a given note $[t,p]$ in order to be identified as legal note pairs, cf. Figure 5.5. In the context of the music DL, $P_0 := 8$ and $T_0 := 32$, yielding 552 frame cells identifying pairwise distinct possible note pair constellations referring to $\lambda \in [0:551]$. For each note pair index λ an inverted file $H_\mathcal{D}(\lambda)$ is constructed from the score files. $H_\mathcal{D}(\lambda)$ contains all pairs (i,t) such that the relative note pair constellation between $[t,p]$ and $[t',p']$ corresponding to λ occurs at the t-th onset time within score file D_i, i. e., $H_\mathcal{D}(\lambda) := \{(i,t) \in [1:N] \times \mathbb{Z} \mid [t,p],[t',p'] \in D_i : (t'-t, p'-p) = \lambda\}$. Using inverted files, query processing may then be performed simply by using intersections of inverted files. Assuming a query is given as a sequence $Q := ([t_0,p_0],\dots,[t_{n-1},p_{n-1}])$ of notes that is preprocessed in the same fashion as documents are. Hence, $Q \mapsto Q' := (\lambda_0,\dots,\lambda_{m-1})$ is computed by the same mapping Δ as above for all legal note pairs of the query Q prior to the retrieval step. Then, denoting time-shifted versions of note sequences D by $\tau + D := \{(i,\tau+t) \mid (i,t) \in D\}$, the set of *transposition-invariant matches*

$$H_\mathcal{D}(Q) := \{(i,\tau) \mid \tau + Q \subseteq D_i\} = \bigcap_{[t,p],[t',p'] \in Q : \Delta(t'-t,p'-p) = \lambda \in Q'} H_\mathcal{D}(\lambda) - t \qquad (5.2)$$

can be easily shown to contain all tuples (i,t) such that the exact relative note constellation pattern among neighboring notes Q' in Q occurs at onset position t within the i-th document, i. e., $H_\mathcal{D}(Q) = \{(i,t_{ik}) \mid \exists\ j,k : \Delta\ (t_{ij} - t_{ik}, p_{it'} - p_{it}) = \ \lambda\}$. Note that note pairs form constellations that are conserved by constant pitch shifts. Therefore a queried melody can be identified independently from a constant shift in individual note pitches.

Since the OMR process is prone to errors, notes to be extracted are often wrongly or even not at all recognized. The former case means that extracted notes often have inaccurate onset times or false pitches. Besides that, user-formulated queries are mostly inaccurate. Consequently, the search has to be fault-tolerant to some extent against such local inaccuracies. In order to make the basic matching procedure robust against mismatching notes, the following modified version of the approach towards fault tolerance described in [24] is incorporated. For a query $Q \subset U$ and a value $k \in \mathbb{N}$ let $H_{\mathcal{D},k}(Q) := \{(i,\tau) \mid |(\tau+Q) \setminus D_i| \leq k\}$ denote the set of matches with at most k mismatches. The latter can be evaluated efficiently by means of the following algorithm. With $Q = \{q_0,\dots,q_{n-1}\}$ and $Q' = (\lambda_0,\dots,\lambda_{m-1})$ obtained by calculating $\Delta(t_j - t_k, p_j - p_k) = \lambda_\ell$ for all legal note pairs $([t_j,p_j],[t_k,p_k]) \in Q$, $H_{\mathcal{D},k}(Q)$ may be calculated by first determining $H_\mathcal{D}(\lambda_\ell)$ for all $\ell \in [0:m-1]$ and then counting multiplicities in the multiset of matches $H^*_\mathcal{D}(Q) := \dot{\bigsqcup}_{\ell=0}^{m-1} H_\mathcal{D}(\lambda_\ell) - t_\ell$. More precisely, $\forall\ \ell \in [0:m-1]$ we first calculate $H_\ell := H_\mathcal{D}(\lambda_\ell) - t_\ell$ in $O(|H_\ell|)$ operations. We note that as the $H_\mathcal{D}(\lambda_\ell)$ are sorted, also the H_ℓ are sorted. We then construct a minimum heap by using the m minimum (i. e., first) elements of the m lists $H_0,\dots,H_{m-1}$. We iteratively remove the minimum element from the heap, insert it into $H^*_\mathcal{D}(Q)$, insert the next minimum element of the list from which the removed element is from into the heap, and afterwards restore the heap conditions. This process is done until no residual list elements are left and the heap is empty. As the H_ℓ have a total of $\sum_0^{m-1} |H_\ell| =: L$ elements, and the heap has height $\lfloor \log m \rfloor$, processing of the heap requires $O(L \log m)$ operations. Obviously, while processing the heap, multiplicities can be counted as the elements of the lists occur in sorted (ascending) order. Let $(i,\tau) \in_\kappa H^*_\mathcal{D}(Q)$ denote that (i,τ) is contained in $H^*_\mathcal{D}(Q)$ with multiplicity κ. Then finally, retaining only elements in $H^*_\mathcal{D}(Q)$ that occur less than $n-k$ times results in the set of matches

$$\{(i,\tau) \mid (i,\tau) \in_\kappa H^*_\mathcal{D}(Q), \kappa \geq n-k\} = \{(i,\tau) \mid |(\tau+Q) \setminus D_i| \leq k\} = H_{\mathcal{D},k}(Q) \qquad (5.3)$$

that contains exactly those matches which correspond to matches in $H_\mathcal{D}(Q)$ with at most k mismatches. In total, this algorithm can be performed with computational complexity $O(L) + O(L \log m) = O(L \log m)$.

The multiplicity values are used for relevance ratings in the subsequent ranking of the set of matches. The relevance of a match is obtained by counting the total number of matched or mismatched note pairs, using the k-mismatch algorithm. The more mismatched note pairs are counted, the less important a query result is and therefore the lower its assigned ranking value shall be. Each match $(i, \tau) \in_\kappa H^*_\mathcal{D}(Q)$ is assigned a ranking value $r_{i\tau} := \kappa/n$ that essentially measures the deviation of the query notes occurring in D_i from position τ on from their corresponding notes as specified in Q. The set of matches is sorted by descending ranking values.

Using the described methods, a time-efficient and fault-tolerant retrieval can be achieved. In our implementation, the time efficiency has been tested on a database containing slightly above 100 000 sheet music pages. Allowing a mismatch rate of $\frac{1}{2}$, i. e., $k := |Q|/2$, it turns out to perform well in less than a second.

5.3.4 Audio-based retrieval

Both the identification of scanned sheet music pages, used in the subsequent preprocessing stages, and the content-based audio retrieval described in Subsection 6.1.4 rely on a mechanism for efficient *audio matching* [77]. Given a short audio snippet, the goal of audio matching is to automatically retrieve all excerpts from all recordings within a database of audio recordings that musically correspond to the query. As opposed to classical *audio identification* [1], audio matching allows for semantically motivated variations as they typically occur in different interpretations of a piece of music. The methods for audio matching introduced in [77] work on chroma features extracted from audio recordings.

As mentioned before, the key idea we exploit for automatic document analysis is to reduce the two different types of data (visual and acoustic music data) to the same type of representation (chromagram), which then allows for a *direct* comparison *across* the two modalities on the feature level.

To also allow for an *efficient* comparison, we further process the chroma features by quantizing the chroma vectors using semantically meaningful codebook vectors as described in [77]. According to the assigned codebook vectors, the features can then be stored in some inverted file index, which is a well-known index structure frequently used in standard text retrieval [139].

In our system, we employ audio matching as described in [77] as an underlying engine for the various music retrieval and identification tasks. The basic matching approach works as follows. Each music document of the repository is converted into a sequence of 12-dimensional chroma vectors. In our implementation, we use a feature sampling rate of 1 Hz. While keeping book on document boundaries, all these chroma sequences are concatenated into a single sequence $(d_0, \ldots, d_{K-1})$ of chroma features. Similarly, a given query music clip is also transformed into a sequence $(q_0, \ldots, q_{L-1})$ of chroma features. This query sequence is then compared to all subsequences $(d_k, d_{k+1}, \ldots, d_{k+L-1})$, $k \in [0 : K - L]$, consisting of L consecutive vectors of the database sequence. Here, we use the distance measure $\Delta(k) := 1 - \frac{1}{L} \sum_{\ell=0}^{L-1} \langle d_{k+\ell}, q_\ell \rangle$, where the brackets denote the inner vector product. The resulting curve $k \mapsto \Delta(k)$ is referred to as *matching curve*. Note that the local minima of Δ which are close to zero correspond to database subsequences that are similar to the query sequence. Those subsequences will constitute the desired *matches* for content-based retrieval, see Chapter 6. Because of the bookkeeping, both document numbers and exact positions of matches within each document can be easily recovered.

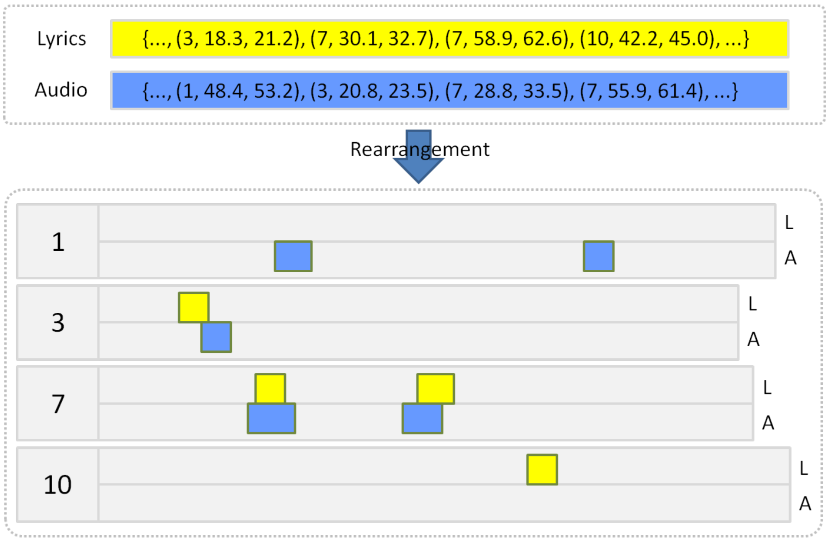


Figure 5.6 Rearrangement of retrieval results for two modalities (lyrics and audio). List elements are gathered and put into corresponding multi-element lists according to their individual associated work ID.

So far we have not yet accounted for possible temporal differences between the query clip and corresponding temporal regions within the audio documents. For example, interpretations of the same piece of music often reveal significant local and global differences in tempo as a result of the freedom a musician has in interpreting a piece. Also, when converting a score representation into a feature representation, one needs to assume a tempo that may deviate from a corresponding audio recording (see, e. g., our assumption in Subsection 5.2.1). To handle such tempo deviations, one can use subsequence variants of DTW [88], or one can employ the technique of multiple querying with various chromagrams at different sampling rates [77]. In particular, the latter technique can be supported by the above mentioned index structure facilitating an efficient computation of the audio matches. For the technical details, we refer to [77].

5.3.5 Multimodal retrieval and advanced ranking strategies

The multimodal retrieval strategy [35] utilizes several search indexes, where individual retrieval results obtained by corresponding query engines are adequately fused and ranked. As query engines act independently from each other, individual querying can be parallelized. Each of the query engines delivers a retrieval result list after processing, where list entry or match consists of a document ID, the position of the matching segment, and a ranking value $r \in [0, 1]$. In the 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 timestamp and an end timestamp. Thus, all segments are directly comparable, which will be exploited in the subsequent combined ranking and merging.

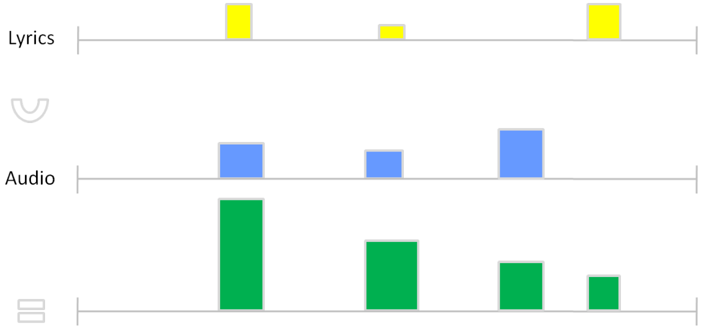


Figure 5.7 Merging of unimodal matching segments (lyrics and audio, respectively) corresponding to the same work into multimodal matching segments. New ranking values are assigned to segments that are collected in the final result, sorted in descending order of ranking values.

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 the ID of a corresponding piece of music together with related segment lists (cf. Figure 5.6). 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 (cf. Figure 5.7). 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 start timestamp b_i^k and an end timestamp e_i^k as well as a ranking value r_i^k. 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. First, 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. Second, 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 into L. Note that overlaps do reflect simultaneously arising hits and for this reason, we generally want them to obtain a higher rank. 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 globally constant 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 $[0, 1]$. Note that the final merging result does not depend on the merging order. This algorithm can be implemented in a straight-forward manner with a computational complexity linearly in the list lengths, as long as each list L^k is sorted ascending

w. r. t. the start 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.

5.3.6 Spatial organization of retrieval results

An alternative to a 1-dimensional organization of retrieval results on the basis of a single ranking value, which can hardly reflect complex similarity relationships, is a multi-dimensional organization on the basis of multiple considered values that can be used for comparison. In this section, we describe an approach for a spatial organization of retrieval results on a 2-dimensional plane, which allows for a more differentiated presentation of retrieval results. Here, retrieval results are arranged on the plane such that individual Euclidean distances between objects express dissimilarities among them.

Similarities between objects are expressed in terms of their individual characteristics. For example, considering, in addition to the ranking value, a second value that is used for comparison like the name of the composer of a retrieval result, retrieval results with the same composer name are aligned on an axis-parallel line of the plane. This kind of retrieval result organization can be directly grasped visually, where retrieval results on the same line are perceived as group or cluster. Therefore, a finer distinction between individual retrieval results is possible, which helps to give a better orientation in retrieval results. Motivated by this example, we want to incorporate several additional values for determining a well-differentiated similarity. In general, we consider n values, each of which quantifies an arbitrary individual characteristic.

5.3.6.1 Multiple characteristics of retrieval results

All the characteristics of a retrieval result form a feature vector that can be regarded as point in $\mathbb{R}^n$. The closer points are located, the more similar they are. By using a suitable metric, distances $\|s_i - s_j\|_n \approx \delta_{i,j}$ corresponding to dissimilarities between points $s_1, s_2 \in \mathbb{R}^n$ can be determined, where $\|\cdot\|_n =: d^{\mathrm{E}}$ is a norm in $\mathbb{R}^n$. Here, the Euclidean distance is used as norm to calculate distances between s_1 and s_2. Note that in fact any distance function can be used.

To calculate the distances between single corresponding feature vector entries, for scalar entries the norm is used, and for character strings (metadata) $t_1, t_2 \in \Sigma^*$, the normalized Levenshtein distance $d^{\mathrm{L}}(t_1, t_2) / \max(|t_1|, |t_2|)$ between them is used. Note that evaluating the norm is performed in constant time, while the computational complexity of the Levenshtein distance is quadratic in string lengths.

5.3.6.2 Dimension reduction of feature vectors

While for one, two, or three considered characteristics the interpretation of an arrangement of retrieval results on a 1D axis, 2D plane, or 3D space, respectively, is intuitive, a higherdimensional arrangement of retrieval results considering more characteristics cannot be directly interpreted due to our restricted visual thinking. A practical solution is based on the idea of reducing dimensions, where points in a high-dimensional space are projected into a lower-dimensional space.

In general, two problems are concerned with dimension reduction methods that are able to perform such a projection. First, the projection should map points into low-dimensional spaces with the least possible distortion of original individual distances between mapped points. Second, the "curse of dimensionality" rapidly leads to computationally intensive calculations to perform the projection for an increasing number of dimensions.

An appropriate dimension reduction method that offers favorable properties w. r. t. the raised issues is the multi-dimensional scaling (MDS) [125], which is used here. MDS is a special case of ordination, a method from multivariate analysis to order (multivariate) objects that are characterized by values on multiple variables. There exist many ordination methods, including the well-known principal component analysis (PCA) and independent component analysis (ICA) that are often applied in statistics.

MDS attempts to find an embedding of high-dimensional points in lower-dimensional spaces, while preserving individual distances between objects. The MDS is performed by building up a matrix consisting of dissimilarities between points, each of which is assigned a location in a p-dimensional space, where p is specified a priori. In particular, for m objects, an $m \times m$-dimensional *dissimilarity matrix*

$$\Delta := \begin{bmatrix} \delta_{1,1} & \cdots & \delta_{1,m} \\ \vdots & \ddots & \vdots \\ \delta_{m,1} & \cdots & \delta_{m,m} \end{bmatrix} \tag{5.4}$$

is constructed, whose entries (i, j) are composed of the respective distances $\delta_{i,j} := d^{\mathrm{E}}(s_i, s_j)$ between points $s_i, s_j \in \mathbb{R}^n$ or $\delta_{i,j} := d^{\mathrm{L}}(t_i, t_j)$ between character strings $t_i, t_j \in \Sigma^*$. Note that the dissimilarity matrix is real and symmetric, and has zeros along the diagonal and positive elements everywhere else. Now the goal is to find to find m vectors $x_1, \ldots, x_m \in \mathbb{R}^p$ such that $\|x_i - x_j\|_p \approx \delta_{i,j} \; \forall \; i, j \in [1 : m]$, where $\|\cdot\|_p$ is a norm in $\mathbb{R}^p$. For our purposes, $p = 2$ and the Euclidean distance is used as norm to calculate distances between x_i and x_j. Note that in fact any distance function can be used. In order to determine the vectors x_i, various approaches exist. A usual approach is to find $(x_1, \ldots, x_m)$ that minimizes a certain cost function, e. g., $\min_{x_i} \sum_{i<j} \left(\|x_i - x_j\|_p - \delta_{i,j}\right)^2$. A solution can be found using numerical optimization methods.

The solution of the MDS is referred to as configuration. Typical configurations are estimated in two or three dimensions to ease interpretation. Besides the spatial configuration, i. e., the projection of feature vectors, the MDS yields additional values (e. g., stress factor) on whose basis the embedding quality w. r. t. the distortion aspect can be evaluated. In our context, for each dimension of the matrix, scaling factors are assigned. Doing so, dimensions of the source space are individually considered as differently important.

Classical MDS There exist several kinds of MDS methods, among which we concentrate on non-iterative, metric MDS. The objective of the classical (metric unweighted) MDS [28, 75] is to arrange m points in a high-dimensional space $\mathbb{R}^n$ with distances $\delta_{i,j}$ in a lower-dimensional space $\mathbb{R}^p$ such that the relative distances of the m points among each other are preserved with the lowest possible degree of distortion (error). For this, the classical MDS essentially relies on two key operations. First, distances are transformed into similarities by means of "double centering" the dissimilarity matrix, i. e., translating matrix vectors[2] to coordinates in a new Cartesian system whose origin is at the centroid of all matrix vectors. Second, spectral decomposition is performed using singular value decomposition for the purpose of reducing dimensions. For an explanation of other kinds of MDS, see [42, 74, 106, 111, 118] and the references therein.

The calculation of the MDS is performed in four steps:

1. Construct a matrix $\Delta = (\delta_{i,j})$ with $\delta_{i,j} := d^{\mathrm{E}}(s_i, s_j)$ between points $s_i, s_j \in \mathbb{R}^n$ or $\delta_{i,j} := d^{\mathrm{L}}(t_i, t_j)$ between character strings $t_i, t_j \in \Sigma^*$.

2. Construct a matrix $\Theta = (\theta_{i,j})$ with

$$\theta_{i,j} := \delta_{i,j} - \frac{1}{m}\sum_{k=1}^{m}\delta_{i,k} - \frac{1}{m}\sum_{\ell=1}^{m}\delta_{\ell,j} + \frac{1}{m^2}\sum_{k=1}^{m}\sum_{\ell=1}^{m}\delta_{k,\ell}. \tag{5.5}$$

3. Determine eigenvalues λ_i and corresponding eigenvectors $\gamma_i = (\gamma_{i,j})$ of matrix Θ with property $\sum_{j=1}^{m}\gamma_{i,j}^2 = \lambda_i$.

4. The coordinates of the scaled points $x_i \in \mathbb{R}^p$ are then obtained from the eigenvectors corresponding to the p largest eigenvalues:

$$x_i = \sqrt{\lambda_i}\gamma_i. \tag{5.6}$$

Note that Θ is the Gram or kernel matrix [115] $Q^{\mathrm{T}}Q$ consisting of inner products (squared Euclidian distances) between vectors in Δ. That is, Θ is symmetric and positive semidefinite, since Δ is a Euclidean distance matrix, and thus can be decomposed. Further note that the calculated distances are insensitive against uniform scaling[3], rotation, and reflection. Using Euclidian distances as norms leads to an intuitive geometric interpretation of the result, since the Euclidian distance is structure-preserving and has a direct meaning: it corresponds to perceptually shortest distances between particular points. Besides that, using other distance measures as norms may lead to negative eigenvalues and may be not structure-preserving in general. Finally note that the computational complexity of most metric MDS methods exceeds $O(N^2)$. Therefore, these are not appropriate for large sets of feature vectors. In consequence, the spatial organization is calculated only for small subsets of retrieval results. However, there have been proposed several scalable MDS algorithms [42, 105] which drastically decrease space and computational complexity, and can be used with large sets of feature vectors.

5.3.6.3 2D visualization of retrieval results

In the current setting, with $m = 50$ the top-50 retrieval results are considered, which turns out to perform in hundreds of a second on current computers. Regarding the number of characteristics, we use $n = 5$, where dimensions correspond to ranking value, work title, composer name, performing artists, and year, respectively. To individually control the influence

[2] Note that row vectors are equal to column vectors, since the dissimilarity matrix is symmetric.

[3] A uniform multiplication of distances results in the same coordinates due to the property $\sum_{j=1}^{m}\gamma_{i,j}^2 = \lambda_i$.

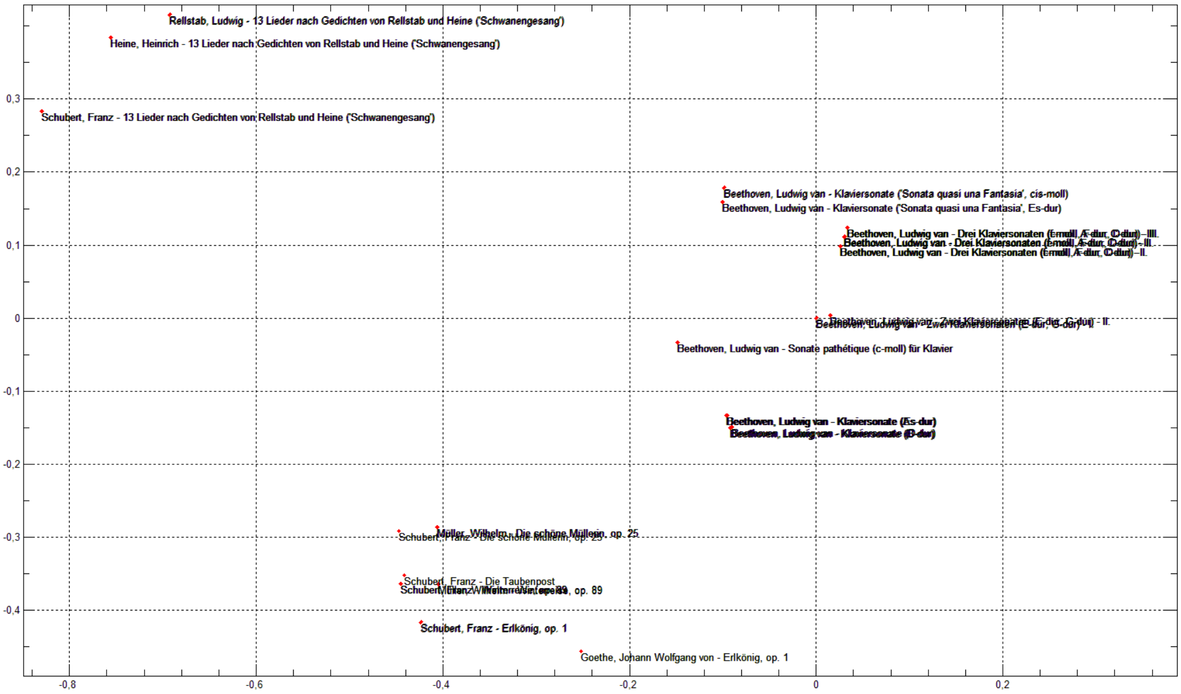


Figure 5.8 2D visualization of retrieval results. Spatially arranged points on the plane correspond to the 50 top-ranked results, for which respective work's composer name and title are shown as labels.

of particular characteristics on the placement of retrieval results onto the plane, we use weights for corresponding feature vector entries. For a feature vector s_i let $s_{i,j}, i \in [1:n], j \in [1:m]$ denote the j-th vector entry. Then, weights $w_j \in [0,1]$ with $\sum_{j=1}^{m} w_j s_{i,j} = 1$ are used to scale the "degree of participation" of each dimension on the singular value decomposition evaluated by the MDS.

In the current setting, we use $w_1 := 0.4, w_2 := 0.2, w_3 := 0.3, w_4 := 0.05$, and $w_5 := 0.05$ so that the ranking value gets the most influencing factor in the placement of retrieval results onto the plane. In this way, marking the the top-ranked retrieval result as the center of the plane, retrieval results are positioned such that less similar results are placed farther away from the center. In Figure 5.8, a configuration of 50 top-ranked results of a retrieval result is illustrated. As it can be seen, very similar results are mapped on almost the same positions, resulting in a superposition of results. It can further be seen, how particular results are being clustered using current weightings of dimensions. The results can roughly be divided into three clusters (top left, middle right, and bottom center) with each consisting of results that principally belong to the same composer (Rellstab, Beethoven, and Schubert, respectively).

However, it makes sense to parameterize these weightings so that the user can decide on the importance of particular dimensions or characteristics to obtain a presentation of retrieval results that fits well on his preferences. As final note, it should be mentioned that it may make sense to incorporate other kinds of features like, e. g., audio thumbnails [6], into projections; see also [105].

Chapter 6

Cross- and multimodal music retrieval and interaction

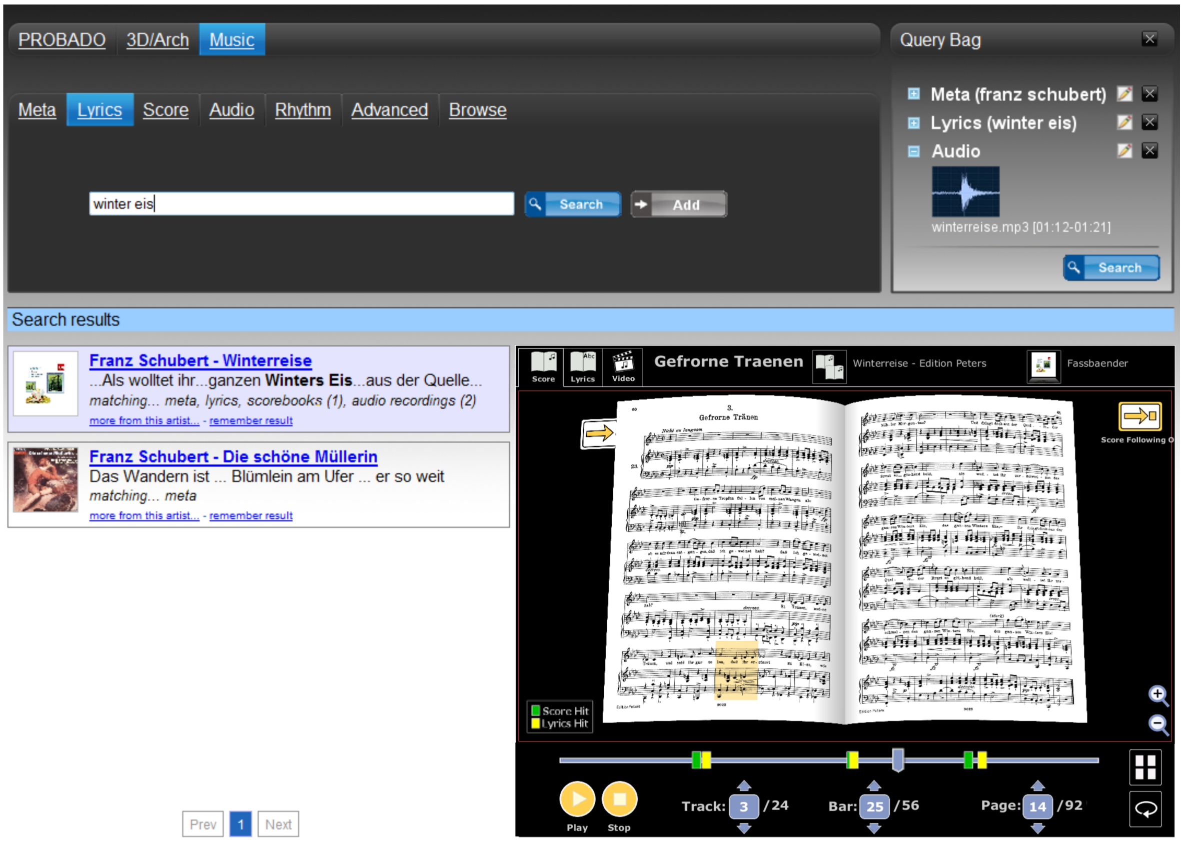
In this chapter, UI components for accessing musical content are presented. One of the key contributions is the multimodal presentation and cross-modal navigation in music documents. For this purpose, a special document renderer that handles the rendering of music documents of diverse formats, referred to as multimodal music player (see Section 6.4), provides various views and modes for the user to operate on music documents (cf. Figures 6.4, 6.6, and 6.7). In particular, it firstly provides the simultaneous, synchronized playback of audio recordings and various types of visualization, including scores, short-time spectra, lyrics, and videos. Secondly, it allows for a cross-modal navigation in music documents. For this, the multimodal music player utilizes synchronization data that links semantically meaningful musical entities of one representation to corresponding ones of another representation of the same piece of music. This is done in a cross-modal way, e. g., the linking of image regions within scanned sheets of music and time segments within audio recordings. In the case of sheet music and audio recordings, the user is on the one hand enabled to visually track the currently played measure of a piece of music within its sheet music representation while listening to an associated audio recording. On the other hand, he has the option to navigate through the sheets of music and select a specific measure in order to change the playback position within the audio recording accordingly. This can be useful, since the sheet music representation gives a more suitable possibility to search for specific parts within a piece of music than auditory searching within the audio recording.

As a major contribution of this thesis, it is proposed how to incorporate a consistent concept for combined multimodal queries into the typical stages of a querying–retrieval chain, particularly query formulation (Section 6.1), content-based retrieval and ranking (Subsection 5.3.5), presentation of query results (Section 6.3) and mechanisms for user feedback and navigation (Section 6.5). For this purpose, our system enables the user to formulate a query that may consist of different modalities, including the textual, visual and auditory modality. In particular, the user is enabled to query a combination of metadata, lyrics and audio fragments. For this, he formulates single, unimodal queries and adds them successively to his search. These queries are gathered in a special structure for representing sets of queries, referred to as bag-of-queries or short query bag. After the query bag is submitted to the retrieval system, the user is presented a list of pieces of music that match his query in at least one modality. To organize the result list, the ranking approach introduced in [35] that is based on a combination of multiple result lists is employed, ensuring that pieces of music containing more matching modalities are given a higher rank, see Subsection 5.3.5. The results, i. e., pieces of music, may then be examined in detail and played back with the multimodal music player. Additionally, both the result list and the multimodal music player can be used for querying, query refinement and document navigation, see Sections 6.1 and 6.5. To give a user-friendly and intuitively operable interaction environment, our approach was to incorporate the look-and-feel of widely accepted Internet search platforms. The UI components for the retrieval, browsing, playback, navigation, and exploration of musical content are completely Web based and run in virtually every state-of-the-art JavaScript- and Java-enabled Web browser. Figure 6.1 shows a snapshot of a typical system configuration. Similar to popular existing query engines, the top part of the UI 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 multimodal music player.

The chapter is structured as follows. In the following, an overview on the Web-based interaction environment and incorporated technologies is given. Subsequently, a further in-detail look at each particular stage of the query–retrieval chain is given (Sections 6.1, 6.2.1, 6.3, and 6.5). Here, all the UI components are considered in detail. A detailed view on the multimodal music player and its particular audio-visual playback and navigation capabilities is given in Section 6.4. Throughout the whole chapter, the music work “Gefror’ne Thränen” belonging to the song cycle “Winterreise” by Franz Schubert will serve as running example.

6.1 Unimodal query formulation and retrieval

This section gives some more detail on the provided querying functionality. As mentioned before, a key task in the context of the targeted music DL is to enable content-based search using lyrics, score, and audio 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. That 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, three distinct options for content-based querying are available, lyrics-based retrieval as proposed in [92], score-based retrieval as proposed in [26], and audio-based retrieval using audio matching as proposed in [77]. All three approaches use indexing techniques to achieve a high retrieval efficiency. Besides identifying a particular manifestation that contains the user’s query in some sense, the content-based search techniques are capable of determining the exact matching positions or regions of the query results within the manifestations. The matches are returned as an ordered list of matches sorted in descending order of relevance. In the following, a detailed look at raised unimodal querying options is given (Subsections 6.1.1, 6.1.2, 6.1.3, and 6.1.4). Subsequently, multimodal query formulation and its UI is exposed in detail (Section 6.2).



winter eis

Meta (franz schubert)

Lyrics (winter eis)

Audio

winterreise.mp3 [01:12-01:21]

Franz Schubert - Winterreise
...Als wolltet ihr...ganzen **Winters Eis**...aus der Quelle...
matching... meta, lyrics, scorebooks (1), audio recordings (2)
more from this artist... - remember result

Franz Schubert - Die schöne Müllerin
Das Wandern ist ... Blümlein am Ufer ... er so weit
matching... meta
more from this artist... - remember result

Figure 6.1 Web-based UI for multimodal and combined query formulation and browsing (top left), the query bag (top right), aggregated display of search results (bottom left), and the multimodal music player (bottom right). Adapted from [34].

6.1.1 Metadata-based retrieval

The metadata-based retrieval allows the search of records in the metadata catalog. A query is formulated in the textual modality in order to find manifestation and work entries that match the query's specified metadata descriptors. This can happen in two different ways, either as simple full-text or as advanced fielded search, each of which has its specific advantages. The usefulness of a particular way depends on the user's individual preference and purpose. While professionals in music may want to formulate more specific queries using the advanced search, non-professionals may want to formulate less specific queries using the simple search. A query for the simple full-text search can be formulated by entering one or more metadata descriptors in a single text field provided for this purpose. A query for the advanced fielded search can be formulated by explicitly entering certain metadata descriptors in various respective designated fields. Entering the query in the system, the system in turn retrieves all works that match the query's specified metadata descriptors in their respective metadata records of the metadata catalog. The simple search implicitly searches the entered metadata descriptors in all fields of the metadata catalog records. Therefore, a specified metadata descriptor will be found (if available) regardless of the metadata catalog records' fields the descriptor is found in. In contrast, the advanced search explicitly searches the entered metadata descriptors in the respective fields of the metadata catalog records. Thus, the advanced search is more constrained than the simple search in the sense that each search criterion must match a corresponding metadata catalog record's search criterion of the same type. For example, while an advanced search for the phrase "piano sonata" entered in the metadata "title" field results in all works that include this phrase in some form in their respective titles, the same terms entered in the metadata "creator" field most likely leads to no results.

The simple search utilizes an indexing technique that is based on inverted files, see Chapter 5. The search is fault tolerant w. r. t. misspelled or omitted words in both the query as well as the metadata records. The advanced search utilizes a database search technique that is based on the MySQL LIKE operator.

Depending on its quality, each result of a simple search is assigned a ranking value $r \in [0, 1]$. In contrast, the advanced search is based on database search techniques and is thus a Boolean search. Hence in this case there is no ranking value assigned to the results—either a query does match or not. For those results a ranking value of 1 is assigned.

6.1.2 Lyrics-based retrieval

The lyrics-based retrieval allows for formulating a content-based query in the textual modality in order to find exact positions within audio recordings where the words are sung. This can happen in two fashions, applying either the free-form or the query-by-example paradigm. First, a query can be formulated by entering a sequence of words or phrases in a provided text field by means of the keyboard. Second, applying the query-by-example paradigm, a query can also be formulated in the visual modality. This is done by either selecting a portion of a sheet music page, more precisely one or more consecutive measures, that contain lyrics syllables, or by selecting one or more consecutive words from the lyrics view. Since the query is actually passed through the query form, i. e., the text field, when entering the system, it can also be manipulated from there. This enables the user to verify and possibly adjust a query and correct wrongly extracted lyrics prior to entering the query in the retrieval system, as the query is constructed from lyrics extraction (OCR results) which can contain errors. Based on the corrected query, subsequently a new search can be initiated, in case that query results are not satisfactory. Entering the query in the retrieval system, the system in turn retrieves all occurrences of the selected music excerpt within the indexed lyrics that have been obtained from the post-processed OMR results. Note that the sheets of music are images obtained from scanned book pages and thus the actual textual content is expressed in the visual modality. Therefore, the textual content has to be obtained from the images in a preprocessing step. This is done by OCR techniques [18, 68]. Although the output of the OCR process is somewhat error-prone, it is sufficient for matching purposes as fault-tolerance mechanism can be applied. The mapping of positions within extracted lyrics or syllables to time segments within associated audio recordings is performed by exploiting the fact that onset times of individual words or syllables are implicitly given by means of the musical context. In particular, both bar-wise synchronization structures between score–audio pairs and heuristics of mapping syllable placements within score bars to time offsets are utilized to determine or estimate the individual onset times. This information, in turn, is then used to synchronize the lyrics to audio recordings. The subsequently used indexing technique is based on inverted files which are well known from classical full-text retrieval [139] 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 [92].

6.1.3 Score-based retrieval

The score-based retrieval allows for formulating a query as either a monophonic melody or a polyphonic score excerpt. This can, again, happen in two fashions, applying either the free-form or the query-by-example paradigm. First, a query can be formulated by entering notes in a Java applet-based form provided for this purpose. This can be done by means of both keyboard or mouse. Second, applying the query-by-example paradigm, a query can also be formulated in the visual modality by selecting a portion of a sheet music page, more precisely one or more consecutive measures. Since the query is actually passed through the query form when entering into the retrieval system, it can also be manipulated from there. This enables the user to adjust a query and correct wrongly extracted notes by the OMR process prior to entering the query, as the query is constructed from symbolic scores which potentially contain many extraction errors. Based on the adjusted query, subsequently a new search can be initiated, in case that query results are not satisfactory. Note that the sheets of music are images obtained from scanned analog pages and thus the actual musical contents or semantics are expressed in the visual modality. Therefore, the images have to be transformed to a symbolical representation in a preprocessing step. This is done by OMR techniques [18, 68]. Although the output of the OMR processing is quite error-prone, it is sufficient for matching purposes, using fault-tolerance techniques. After entering the query into the retrieval system, the system in turn retrieves all occurrences within the indexed symbolic scores that have been gained from post-processed OMR results that are similar to the selected music excerpt.

6.1.3.1 Free-hand query form

The free-hand query form allows to create new melodies or to adjust melodies extracted from a score selection. The free-hand query form offers two different views that allow to express and edit scores in two distinctive ways, operated by means of the mouse and the keyboard.

A somewhat simplified way to create and edit score content is offered by the *piano roll view* that provides the graphical formulation of notes on a 2-dimensional grid plane, as depicted in Figure 6.2. For non-musicians this may be the medium of choice for entering musical notation as this view does not rely on knowledge about Western music notation and thus might be easier to operate with than the score view. For details, we refer to Section 6.1.3.1.

A more common way to create and edit score content is offered by the *score view*. It provides a means for entering notes in symbolic notation following the Western music standard notation system, as depicted in Figure 6.3. For musicians, this is the medium of choice for formulating melodies and score excerpts as it is the most common way to write down notes. A further explanation is given in Section 6.1.3.1.

Besides the visual revision, the free-hand query form for score editing also provides the playback of formulated score contents, which allows for an acoustic review. As the two views describe the same abstract score content, the views are synchronized so that the user can freely switch between them while formulating a query. The synchronization ensures that a melody edited in one view is updated or translated accordingly in the other view. Actually, an abstract representation structure for notes is commonly used from which each view is rendered, created, and updated. Vice versa, modifying the notes or the score in a view, the common structure is updated accordingly. Besides key signatures that directly affect the pitches of notes, a score mainly consists of a set of individual notes, each of which is given by a triple $[t, p, d] \in U$ defined in Chapter 5.

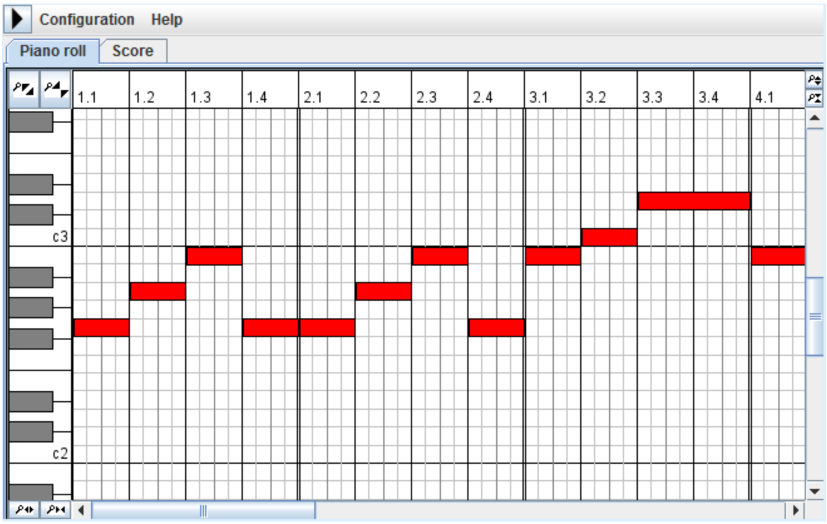


Figure 6.2 Piano roll view of the score-based query formulation form.

Piano roll view In the piano roll view, notes are represented and formulated graphically on a 2-dimensional grid plane (cf. Figure 6.2). The plane is divided into the discrete grid $\mathbb{Z} \times [0:127]$, representing onset times and pitches. Within the plane, notes are represented by rectangles of a fixed height, variable horizontal and vertical positions, as well as widths encoding times, pitches, and durations, respectively. The rectangles (notes) can freely be created, deleted, changed in widths (durations), and moved around the grid (altered in onset times and pitches).

More precisely, a note is represented by a rectangle $[t, t+d) \times [p, p+1)$, where $t \in \mathbb{Z}$ is the note's onset time, $d \in \mathbb{Z}_{>0}$ the duration, and $p \in [0:127]$ the pitch. Each of the 128 pitch levels represents a semitone according to the MIDI standard, where the pitch number 69 corresponds to the standard concert pitch A4 (a', 440 Hz), a transition from p to $p+1$ corresponds to a semitone step upwards in the scale, and an octave consists of 12 semitone steps. Note that in comparison to the score view where individual pitches are implicitly given according to the currently used key signatures, the piano roll view directly renders all pitches explicitly as there are per se no key signatures. The pitch axis is represented by a virtual piano keyboard rotated counterclockwise by 90 degrees. Semitone steps and octaves are displayed by thinner and thicker horizontal lines, respectively. For representing onset times and durations the time axis resolution is quantized into 64ths of a measure which are displayed by thinner and thicker vertical lines. Accordingly, a note's duration of $d = 1$ corresponds to a 64th length. In particular, notes $[t, p, d]$ are elements of the note universe U defined in Chapter 5. Furthermore, the time axis is subdivided in quarter notes, where the vertical lines correspond to either 64th, 8th, or quarter notes, depending on the currently used configuration. The time signature can be changed via the appropriate menu entry.

Notes are created by using the mouse. By means of a click on a free cell of the grid, a note is created and inserted into the plane as a rectangle with pitch and onset time according to the mouse pointer position, and a duration of 1 (or, a 64th note length). By means of clicking, holding, and dragging, a note is created and placed on the plane as rectangle with pitch and onset time according to the mouse pointer position, and a duration according to the release position after dragging. Doing so, the newly created note is played back. Both pitch and onset

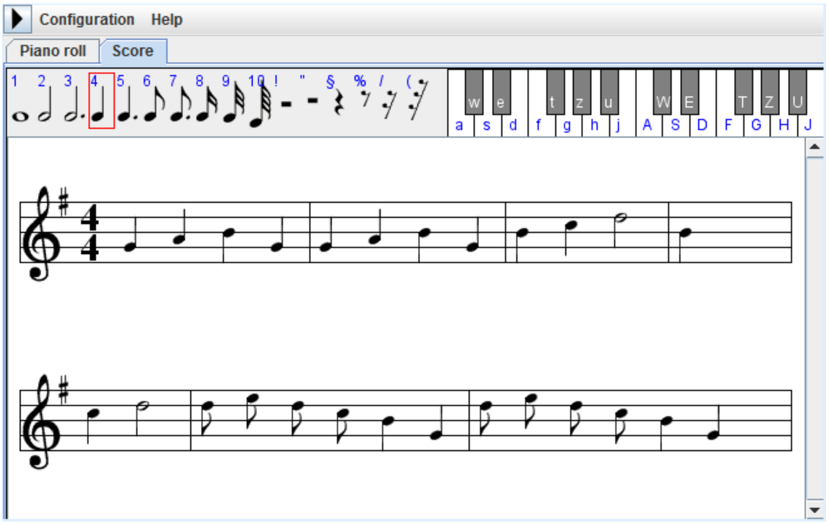


Figure 6.3 Score view of the score-based query formulation form.

time of a note can be corrected by freely moving it around the two dimensions. This happens by means of dragging the appropriate rectangle on the plane. The duration of a note can be adjusted by dragging the right edge of the appropriate rectangle. To remove a note from the score, the appropriate rectangle is to be right-clicked.

When playing back the input melody or score, the current playback position is visualized by a vertical yellow line moving from left to right. The plane is zoomable in both dimensions by means of the appropriate controls (magnifying glasses) in the corner of the piano roll view.

Score view In the score view, notes are represented and formulated graphically by means of notation symbols from the Western music standard notation system (cf. Figure 6.3). The score view's main part comprises one or more staff lines, each of which beginning leftmost with the treble clef along with the currently chosen key and time signature that both may be changed via appropriate menu entries. Each of the staff lines may contain zero (in case the query is empty), one, or more bars that are separated from each other by vertical bar lines and group together sets of notes, depending on the time signature. Melody or score arrangements can be created or revised by inserting or changing individual notes and chords on staff lines by using either the mouse or the keyboard. Depending on the currently chosen key signature, pitches of individual notes may be altered accordingly.

On top of the score view's main part, the toolbar is located. From there, different notes and pauses can be entered by using the mouse. The toolbar is divided into a left (note lengths and pauses) and a right (two octaves-ranging claviature from c' to b'') part, from where notes' lengths and pitches, respectively, can be chosen. A currently chosen note length lasts as long as another note length is chosen. By clicking on a claviature's key, the corresponding note is drawn in the score at the current position (indicated by a caret), depending on the chosen length, pitch, and key signature. The claviature can also be operated by means of the keyboard. Notes and pauses can also be entered by clicking directly on a certain position within the scores. In turn, a note or a pause is placed at the chosen position, dependent of both the time and the pitch position of the mouse relative to the staff where the click occurred. A pause may also be inserted by pressing the “p” key on the keyboard. By repeatedly clicking on an existing note, the sign can be circularly changed in order to transpose the note by semitones.

Chords are created as soon as another note is placed or inserted at a time position that is already occupied by another note of different pitch. Each note or chord is surrounded by a rectangle that becomes visible when hovering over the note or chord. Tied notes can be created manually by dragging one note onto a subsequent note on the right. Neighboring notes are automatically split into two tied notes if an inserted note overfills a measure. Clicking the "x" icon in the upper right corner of a surrounding rectangle removes the appropriate note or chord from the score. A note or chord can also be removed from the score by means of either the backspace or the delete key on the keyboard.

When playing back the input melody or score, the current playback position is marked by a transparent rectangle surrounding the appropriate note or chord.

6.1.4 Audio-based retrieval

The audio-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, more precisely one or more consecutive measures. After entering the query into the retrieval system, the system in turn retrieves all occurrences of the selected music excerpt within the indexed audio recordings. Note that the sheets of music are images obtained from scanned book pages and thus the actual musical content or semantics is expressed in the visual modality. Exploiting the previously described score–audio synchronization, instead of querying the selected score excerpt, the corresponding snippet of the associated (synchronized) audio recording that is currently used for acoustic playback 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 feature 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. Therefore, musically similar snippets can be found regardless of a particular performance [62, 6]. For a more detailed view, we refer to [80] and the references therein.

6.2 Multimodal query formulation and retrieval

Now we want to turn towards the multimodal retrieval. The query formulation area, shown in Figure 6.1 (top), consists of various query formulation forms for each 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 queries based on metadata, lyrics, melodies, score excerpts, and audio fragments by using the designated tab cards.

From within any tab card the user has the choice to either perform an immediate, unimodal search using the just formulated query (classical querying scenario) 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 in detail by clicking the plus-sign icon to the left of the query. To the right of each query there are icons for reformulating or manipulating the query and for removing it from the query bag as well. By clicking the pencil icon, the corresponding query formulation tab opens, ready 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.

6.2.1 Query processing and advanced ranking strategies

Once the query bag is submitted, the system disassembles it and delegates each contained single (individual) 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. The exact behavior of the multimodal retrieval strategy is described in detail in Subsection 5.3.5 from a technical point of view.

6.3 Integrated presentation of retrieval results

Typically, available search engines provide the user only with a flatly organized result list, where the list entries commonly consist of single documents. However, in case of the music domain, there are multiple document types (in our case audio recordings, sheets of music, and extracted lyrics) 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, this consideration is incorporated into the presentation of query results in the music DL system.

The bottom area of Figure 6.1 shows the result list (left) and the multimodal music player (right). While the result list shows matchings regarding the query on the level of whole pieces of music, the multimodal music player offers access to the entire indexed content belonging to the currently selected piece of music. The multimodal music player furthermore gives a detailed view on exact matching positions or regions within the piece's individual multimodal content (manifestations) on a fine-grained level and is also responsible for playing back and visualizing 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 matchings 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 or work surrogate. The latter shows the creator's name and the title of the corresponding piece of music, lyrics excerpts if available, as well as the matching data subsets, modalities, or manifestations along with their number (in brackets). Additionally, at the bottom there are links to show or retrieve all titles of the same creator and to save the result (see also Section 6.5). A more detailed view of the individual matching documents as well as the exact matching positions therein is also given in the multimodal music player.

Another key feature of the multimodal music player is the integrated display of matching segments along the timeline bar at the bottom (cf. the bottom area of the multimodal music player depicted in Figure 6.1). Besides the scrolling and the adjustment of the current playback position by using the slider knob, this display is used to show all matching positions or segments within all manifestations available for the currently selected piece of music at a glance. The matching positions are represented by small boxes along the timeline bar indicating their respective, individual temporal positions within the piece of music. The matching positions for the currently selected ("active") multimodal contents used for playback are depicted by colored boxes along the timeline bar, where the color and brightness or intensity of the boxes encode modality and ranking value, respectively. Additionally, matching positions within "inactive" documents, i. e., others than those ones used for playback, are displayed as gray boxes. Hovering over an individual match shows detail information about the exact position within the corresponding manifestation that contains the match along with its ranking value. Clicking on an individual match results in loading or "activating" the corresponding manifestation, followed by a jump to the beginning of the matching position. Subsequently, the synchronous, multimodal playback is started from the new position.

Besides the overview presentation of the matchings on the timeline bar, they are also displayed as semi-transparently colored regions within the appropriate manifestations or views. This should encourage the user's orientation even more.

6.4 Multimodal music presentation and interaction

In this subsection, we give a detailed view on the multimodal music player (see, e. g., Figure 6.4 depicting the score visualization mode), the central component for multimodal music presentation and navigation. The multimodal music player allows for the simultaneous, synchronized playback of musical content associated to a currently selected piece of music, including audio recordings, sheets of music, lyrics, and videos. More precisely, besides the playback of audio recordings, it provides four visualization modes, including a score, a spectrogram, a lyrics, and a video visualization. For example, while an audio recording is played back, available sheets of music or lyrics are displayed synchronously; i. e., the user can visually track the currently played measure or the currently sung words, respectively, within the audio recording (cf. Figures 6.4 and 6.6). Due to this style of experiencing music in a multimodal way, the multimodal music player may be thought of as being a multimedia player equipped with sophisticated options for user interaction such as navigation and query refinement, which are examined in Section 6.5.

The multimodal music player is divided into three areas, the top, the center and the bottom area, which are in the following explained in more detail. The top area consists of the metadata display and four buttons, each with associated text right next to it. The metadata display (left) shows the creator's name and the title of the currently selected piece of music. The two stacked buttons right next to the metadata display are used for exchanging score books and audio recordings. The buttons' individual text shows the currently selected documents (manifestations) used for presentation or interaction. In case that more than one document per modality is available, the user can freely exchange which documents are used for the audio-visual presentation of a piece of music. This can be achieved by clicking either the score book icon or the album cover art icon, respectively, whereafter a corresponding pop-up menu lists all available contents associated to the piece of music, from where the user can choose which audio or visual content, respectively, is used for playback. For example, if different audio recordings of a piece of music are available, the user has the choice of deciding which specific acoustic performance he wants to listen to. With this functionality, he is also allowed to switch between different performances while retaining the actual playback position in the musical sense. Thus, the user can additionally draw local comparisons between different

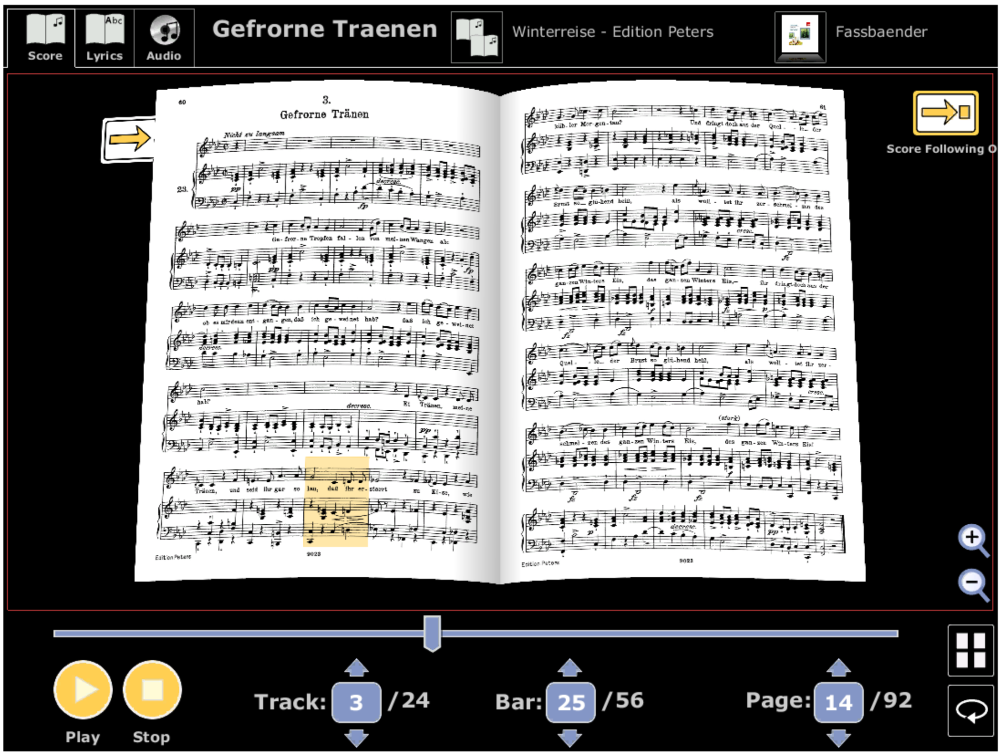


Figure 6.4 Multimodal music player in score visualization mode showing a scorebook in which the current measure is highlighted. The user can alternatively thumb through the pages. Selected measures can be used for querying. Adapted from [34].

interpretations of a piece of music. Sheet music books may also be exchanged if more than one edition is available. The rightmost two buttons are used for navigating in currently selected manifestations' track lists and A–B repeat playback.

In the center area, the various visualizations are displayed. Here, in conjunction with acoustic playback the user is presented either the score, the spectrogram, the lyrics, or the video view, exclusively. The view depends on the currently selected visualization mode, i. e., a specific visualization can be activated by clicking its appropriate view button. For each view or modality (except the auditory one), a designated view button exists to the left of the top area.

The bottom area shows a timeline bar that enables the user to adjust the playback position by moving the slider knob. 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 mode (cf. Figures 6.4, 6.5, 6.6, and 6.7).

For unification reasons, all views are based on the same underlying consistent query-by-selecting retrieval concept. Following the query-by-selecting or query-by-example paradigm, all the views can likewise be used to trigger a content-based search for content, and that in either the same or another modality than the modality the query is of. That is, from within every view the user is enabled to search content-based and cross-modal in any modality. In conjunction with the query bag concept, the user is also enabled to simultaneously search for content in multiple modalities, referred to as multimodal search.

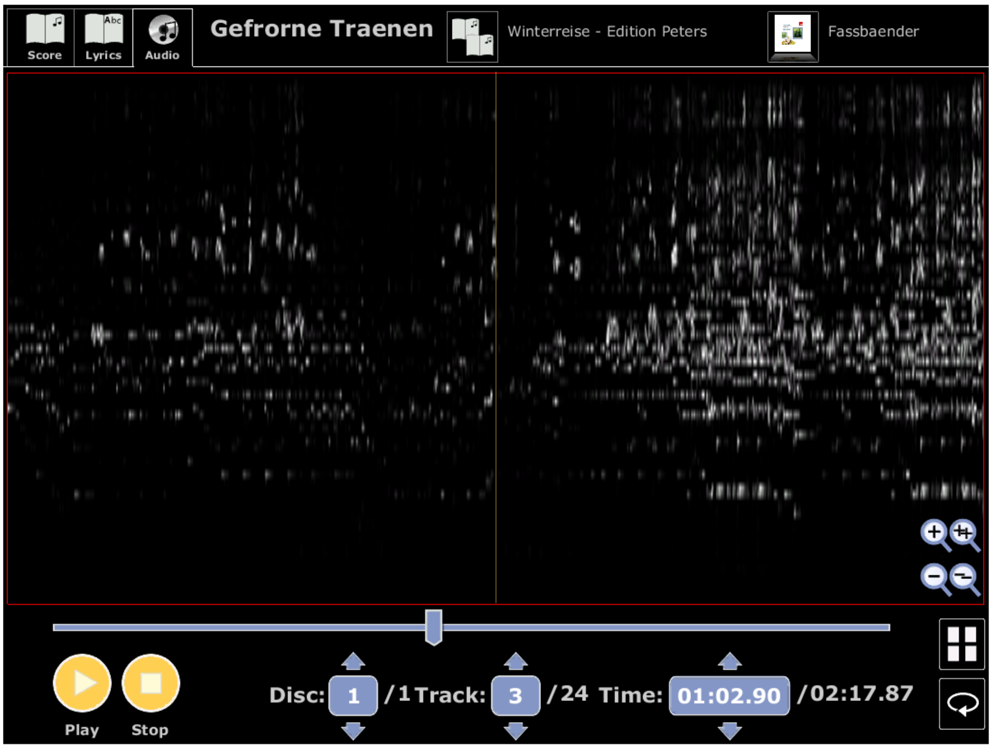
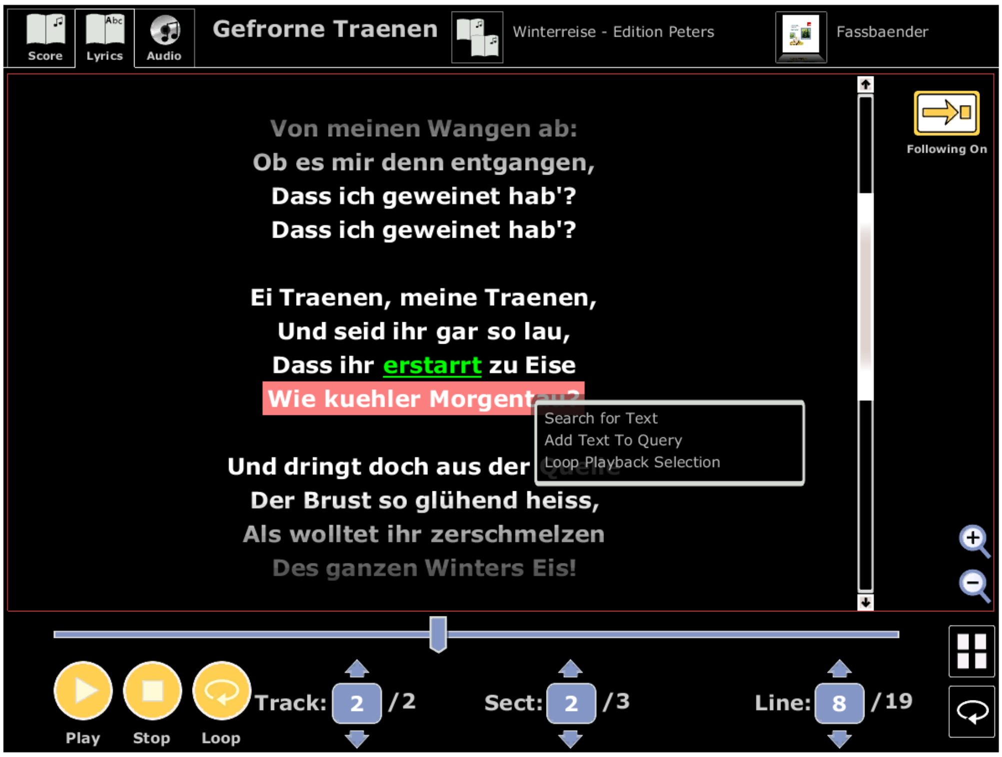


Figure 6.5 Multimodal music player in audio visualization mode showing a frequency-binned spectrogram. Selected segments can be used for querying. Adapted from [34].

6.4.1 Score visualization mode

In the score visualization mode, the multimodal music player presents sheet music and associated audio recordings to the user, as illustrated in Figure 6.4. Here, a virtual score book showing two pages of our running example is shown. When starting the playback of an audio recording, corresponding measures within the sheet music are synchronously highlighted by a semi-transparently colored rectangle, referred to as highlighting box. This feature utilizes the linking structure generated by the score–audio synchronization described in Subsection 5.2.3. In Figure 6.4, a region at the bottom of the left page, corresponding to the 26th measure of our running example, is overlayed with the highlighting box, depicting the current playback position. When reaching the end of an odd-numbered page during playback, the page is turned automatically if score-following is enabled (cf. the “Score Following On/Off” button displayed in the upper right). The multimodal music player, particularly its score visualization, manages entire scanned sheet music books and hence also allows to navigate through those books. For navigation purposes, the user may thumb easily through a sheet music book by either pointing and clicking on the edge of an individual page or moving the slider knob of the timeline bar located below the score book. When scrolling through the score book during playback, the score-following is disabled—otherwise the multimodal music player would scroll back to the page containing the currently played measure. A convenient way of changing the playback position of the audio recording is to select a specific measure within the sheet music. By clicking on a particular measure, the playback position is changed accordingly to where the measure begins to play. Furthermore, when scrolling through the book with the help of the timeline bar, the playback position is also changed accordingly to where the slider knob is



Von meinen Wangen ab:
Ob es mir denn entgangen,
Dass ich geweinet hab'?
Dass ich geweinet hab'?

Ei Traenen, meine Traenen,
Und seid ihr gar so lau,
Dass ihr erstarrt zu Eise
Wie kuehler Morgentau?

Und dringt doch aus der Quelle
Der Brust so glühend heiss,
Als wolltet ihr zerschmelzen
Des ganzen Winters Eis!

Figure 6.6 Multimodal music player in lyrics visualization mode displaying sung text, where the current word is highlighted. Selected text fragments may be used for querying. Adapted from [34].

being released. Additionally, the score visualization supports smooth zooming, allowing the user to magnify the sheets of music. Besides those presentation and navigation features, the score of an underlying piece of music, particularly consecutive measures, can also be used for content-based retrieval purposes. This characteristic is discussed further in Section 6.5.

6.4.2 Audio visualization mode

In the audio visualization mode, the multimodal music player acts similar to an audio player showing a short-time spectral analysis of the audio signal, as illustrated in Figure 6.5. Here, the spectrogram, i. e., the spectral contents over short time windows, is displayed on a time-frequency plane, where the frequency axis is logarithmically scaled and divided into 88 frequency bands. The division in 88 frequency bands, which is also referred to as frequency binning, corresponds to the 88 semitones of the well-tempered piano. The vertical dash in the middle of the plane indicates the current playback position. By means of this view, the user is enabled to see which frequencies in what intensity at what times are involved in the underlying audio recording. In addition, the view offers navigation and retrieval capabilities. For navigation purposes, the user can utilize the time-frequency progression to orient himself via the visual modality using melodic contours. For example, melodic contours can be used as clues in order to identify and find specific parts of the currently loaded piece of music. When starting the playback of an audio recording, the vertical dash moves right through the spectrogram, with new parts running from right into the view A convenient way of changing

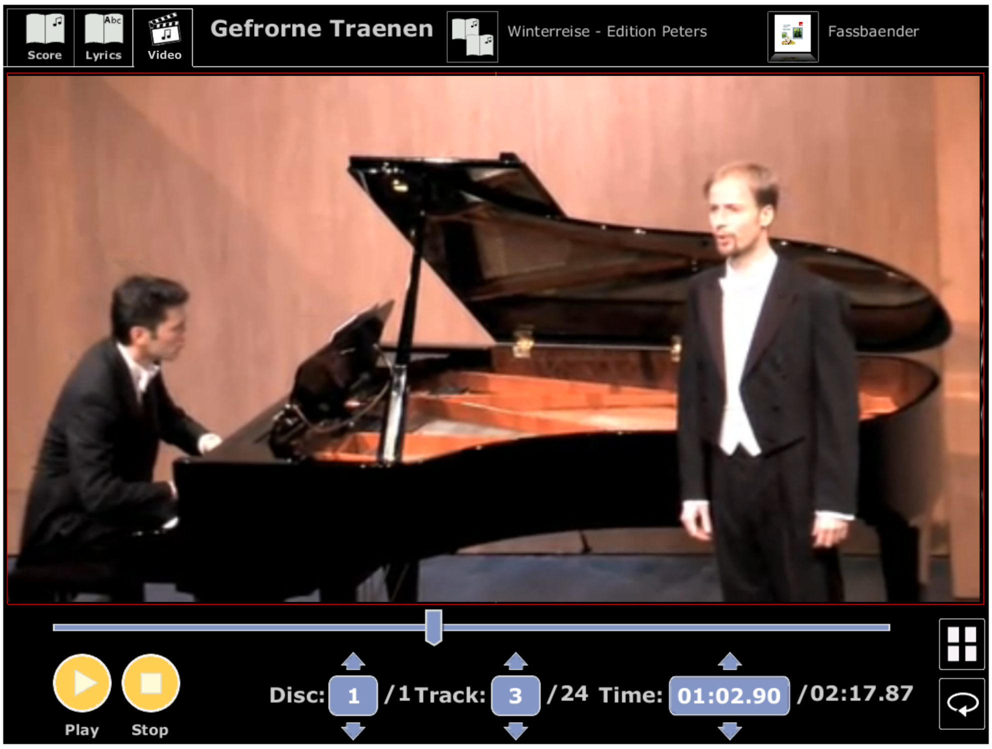


Figure 6.7 Multimodal music player in video playback mode. A video recording of a performance of the current piece of music is shown. Adapted from [34].

the playback position of the audio recording is to select a specific part within the spectral view. By clicking on a particular location in the spectrogram, the playback position is changed accordingly. For scrolling purposes, the view can be panned along the horizontal axis by means of dragging with the mouse. When panning the view, the playback resumes at the position where the user has released the mouse button. Additionally, the time axis can be zoomed. At the limits, the zooming enables the user to get either an overall view on the spectral contents of the underlying audio recording or a detailed view on the frequency distribution over short time segments of the underlying audio recoding. Besides those presentation and navigation features, the spectral contents of an underlying piece of music, particularly small selections of the spectrogram (corresponding to short audio snippets), can also be used for audio-based retrieval purposes. This characteristic is considered further in Section 6.5.

6.4.3 Lyrics visualization mode

In the lyrics visualization mode, the multimodal music player presents lyrics—in the context of classical music also referred to as libretti—and associated audio recordings to the user, as illustrated in Figure 6.6. Here, a textual excerpt of the lyrics of our running example is displayed in a teleprompter-like manner. When starting the playback of an audio recording containing singing voice, corresponding words within the lyrics text are synchronously highlighted by displaying them underlined and in a special color, similar to the score visualization. Again, similar to the score visualization, the lyrics text scrolls appropriately during playback, if text-following is enabled (cf. the “Following On/Off” button displayed in the upper right). For

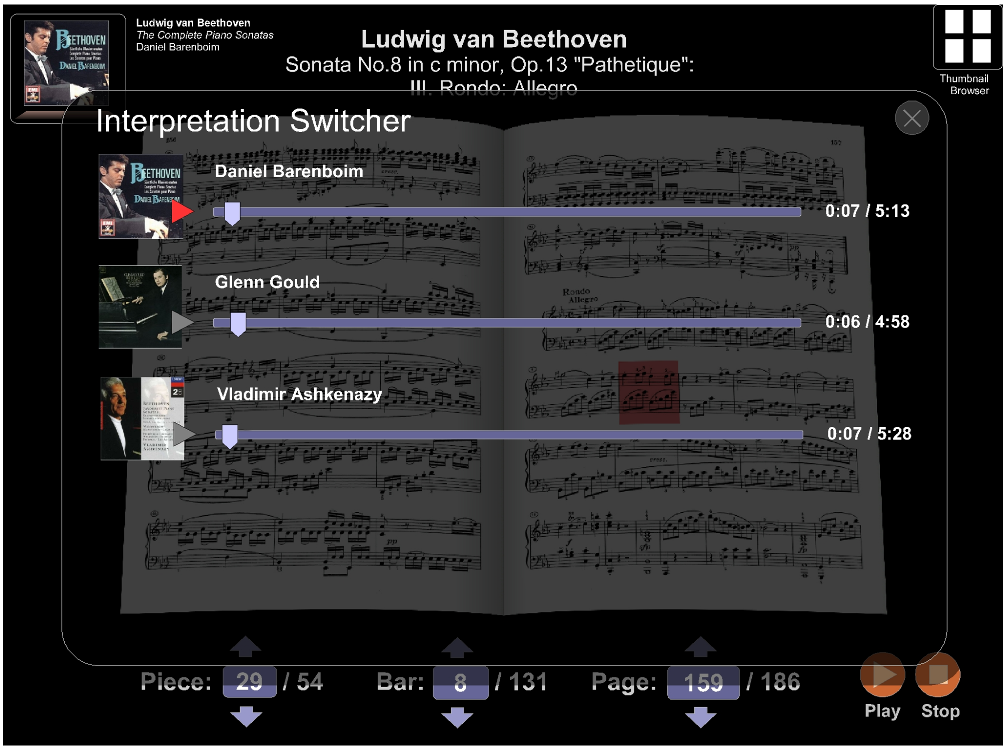


Figure 6.8 Multimodal music player in interpretation switching mode. The three timeline bars correspond to three interpretations of the same work. Adapted from [76].

navigation purposes, the user may scroll the text by using the vertically aligned scrollbar to the right of the text. When scrolling the text, text-following is disabled. Furthermore, a convenient way of changing the playback position of the audio recording is to select a specific word within the lyrics text. By clicking on a word, the playback position is changed accordingly to where the word is sung. Additionally, the text size can be adjusted. Besides those presentation and navigation features, the lyrics of an underlying piece of music, particularly consecutive words, can also be used for text retrieval purposes. This characteristic is considered further in Section 6.5.

6.4.4 Video visualization mode

In the video visualization mode, the multimodal music player acts like a video player, as illustrated in Figure 6.7. Here, the moving images of a video performance of our running example are played back. For navigation purposes, the user can orient himself towards the moving images in order to find specific parts of the currently loaded piece of music. The video visualization currently does not support specific retrieval capabilities. Further characteristics are considered in [123].

6.4.5 Interpretation switching mode

In the interpretation switching mode, the the user may jump at any time from one interpretation to another interpretation of the same work while preserving the musical playback position,

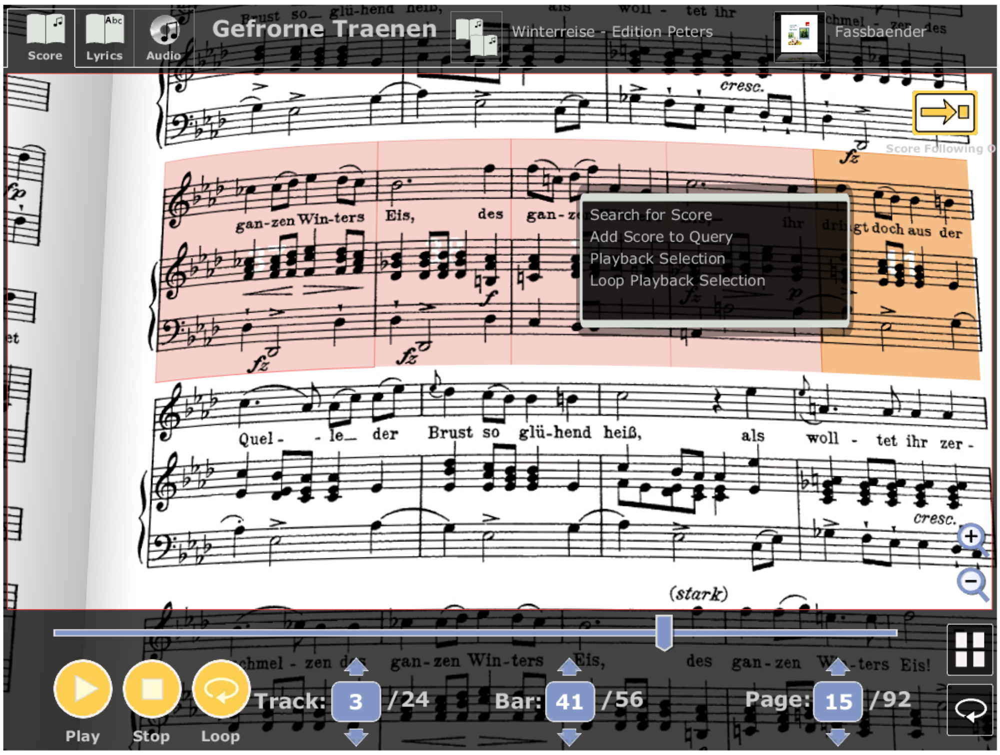


Figure 6.9 Multimodal music player in score visualization mode. Underlying content of selected measures can be queried in several modalities. Adapted from [34].

as illustrated in Figure 6.8. Here, the moving slider knobs of the timeline bars depict time positions in the respective interpretations of Beethoven's Piano Sonata Op. 13 ("Pathethique"). With the help of interpretation switching, the user is enabled to draw local comparisons between particluar interpetations.

6.5 Query refinement and cross-modal 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 a metadata query consisting of only the artist's name and a subsequent new search is performed, finally resulting in an updated list that displays all pieces of music by this artist contained in the database.

Moreover, a user-friendly retrieval functionality based on the query-by-selecting paradigm has been integrated into the multimodal music player. It enables the user to utilize content-based searching capabilities from within visual content following the query-by-selecting paradigm. More precisely, the user is enabled to select specific regions within manifestations using the mouse. When the user selects a region of either sheet music, the spectrogram of an audio recording, or the text of the lyrics, he can use this excerpt for a new query, see Figures 6.4 and 6.9. By right-clicking on the selected portion, a pop-up menu opens where the user has multiple options. He has the option to start either a complete new search based exclusively on the selected portion, or to add the query as an additional partial query to the query bag. Independently from the current view, the pop-up menu lists all currently available search modes. For example, in the case of sheet music, a query may consist of two modalities, score and text (cf. Figure 6.4). Here, the user can choose whether he queries both modalities together or separately from each other. But he has also the option to trigger a cross-modal audio-based search by means of the currently selected portion of the sheet music. In general, selecting a musical portion from within any view, the user is enabled to formulate cross- or unimodal score-, audio-, or lyrics-based queries. Again, this reflects the consistently implemented concept of the holistic consideration of multimodality. As an example, consider Figure 6.9 where some measures from our running example are selected as a query for an audio-based retrieval. Exploiting the linking structure generated by the score–audio synchronization described in Subsection 5.2.3, the selected sheet music region corresponds to a certain time interval of the audio recording currently used for playback. In our example, the measures of the piece of music correspond to the seconds 51 to 58 of a performance by Fassbänder. A sequence of chroma features is then extracted from the audio recording. Subsequently, audio matching as described in Subsection 5.3.4 is used to query the feature sequence in the audio index. Note that as the query features are extracted from an audio recording, they are not affected by possible OMR errors.

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

Chapter 7

Conclusions and future work

In this thesis, we presented a framework for a digital music repository that has been developed for use in real-world library scenarios. For content-based document analysis, search, browsing, and navigation, several state-of-the-art methods from the field of music IR are put into practice, bridging the gap between basic research, on the one hand, and real-world applications, on the other hand. As its main components, the framework comprises

- a data model for managing bibliographic data, metadata and the available heterogeneous document types,
- an overall workflow for automated document processing, particularly addressing the steps of content analysis, annotation and indexing,
- retrieval techniques with adequate UI components that allow for content-based searching, browsing and navigating the underlying document collection, as well as
- a modular repository architecture.

As a first major contribution and underlying principle of the proposed system, cross-modal document processing is an integral part of all the stages of the document processing chain, from ingesting new music documents into the system up to the search, retrieval, and delivery of linked musical content. As a second major contribution, to facilitate cross- and multimodal retrieval, we propose an enhanced retrieval strategy offering composite queries.

We illustrated how the proposed framework is currently set up at the BSB Munich as part of the PROBADO digital library initiative. Note that the developed software system and workflows are not restricted to work only within the BSB—rather they are realized for generic application in real-life libraries. Both the developed software system and workflows have been designed as *generic* components and may hence be equally used in a wide range of real-world music library scenarios. In fact, the underlying DL framework comprises a generic DL architecture for generalized document types and is currently installed for additional document collections such as 3D models from the field of architectural data [14].

The music DL system described in this thesis is an integral part of a superordinate DL system in which the music DL is embedded as subsystem. This superordinate system, evolved from the German PROBADO digital library initiative, provides DL services of all of its embedded subsystems as well as a Dublin Core (DC)-based metadata search system that aggregates harvested DC-compliant subsets of metadata of its individual subsystems. The PROBADO digital library initiative is a research effort to develop next generation DL support for non-textual documents. It aims to contribute to all parts of the 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 PROBADO Music containing music. The PROBADO Music service is based on an implementation of the music DL system considered in this thesis, which is set up at three different sites including the BSB Munich.

During the last year, the PROBADO Music service based on the music DL framework presented in this thesis has gradually been integrated into the librarian archiving and retrieval system of the BSB. The digitalization efforts of the BSB Munich so far produced a total of about 100 000 digital copies of music documents, including 95 000 scanned pages of sheet music and 4000 CD-audio recordings. The data set consists mainly of classical romantic pieces of music, including piano sonatas, string quartets, and orchestral works. While the indexing process of piano sonatas works quite well so far, a particular challenge pose complex orchestral works because of the high number of instruments involved. Especially the occurrence of transposing instruments (relative detunings of individual involved instruments of concrete realizations) has a negative influence on the construction of chroma features, the basis for the score search index and alignments [124]. Therefore, and since in general it will be hardly possible to avoid all possible types of errors, the DMS offers an interactive dialog system in case of the failure of document preprocessing.

Possible applications of the music DL system are diverse. There are a number of practical application scenarios, where such a system can be used effectively. One aspect concerns the user interaction with the system. For example, the system can be used for a variety of studies in the field of musicology. Here, the system can support musicologists in typical tasks like the rapid location of aurally or melodically similar pieces of music in large collections of music documents, or the direct and local comparison between various acoustic performances of the same piece of music. The system can also be used to facilitate the study and rehearsal of pieces of music—already available acoustic interpretations of a piece of music may give clues on how to realize particular passages in the scores of that piece. Besides these tasks, a variety of other application possibilities are conceivable. Another aspect concerns the distribution of music. For example, music publishers could use the system to distribute their digital musical media over the Internet in an economic and appealing fashion. Yet another aspect concerns the automatic processing of Web-scale music data sets comprising millions of documents. For example, search engine providers usually have indexed vast amounts of music documents, where the latter are often unstructured. Using the methods presented in this thesis may help to structure these data sets, and offer them to search engine users. Note that, as previously discussed, a full automation can currently only be achieved by compromising the processing quality.

Required future work is manifold and ranges from several open basic research tasks to be addressed, over the adaptation of existing research results to become feasible solutions for everyday use in a library, to the improvement of the proposed workflow and its adaptation to further relevant processing modes and document types, as well as the detailed evaluation of preprocessing times, search times, and search quality. As some important examples, we mention the robust alignment of scanned sheet music to CD-audio recordings considering structural differences, variabilities or errors in the different documents, or the improvement of OMR/OCR results by using additionally available side information. Furthermore, automatic detection and processing of only partially available documents or inter-document inconsistencies is a challenging task for future work. Additionally, the review of the synchronization results is an important part of quality assurance and improvement. This specific task can neither be achieved automatically nor is a systematic execution by the library staff reasonable for very large data sets. Thus, a system for user feedback should be established to report incorrect synchronizations. Subsequently, the erroneous synchronizations can be revised by the library staff.

While the music DL framework presented in this thesis, for the first time, employs various singular music IR mechanisms such as music synchronization, matching, and indexing, the systematic use of such techniques to achieve a fully automated content-based indexing of music collections comprising solving the latter tasks, remains a largely open challenge.

List of Figures

Acronyms

AJAX Asynchronous JavaScript and XML

BPM beats per minute

BSB Bavarian State Library ("Bayerische Staatsbibliothek")

CD compact disc

CLI command line interface

CSS Cascading Style Sheet

DC Dublin Core

DL digital library

DMS document management system

DOM Document Object Model

DPI dots per inch

DTW dynamic time warping

ER entity–relationship

FFT fast Fourier transform

FRBR Functional Requirements for Bibliographic Records

GUI graphical user interface

HTML Hypertext Markup Language

HTTP Hypertext Transfer Protocol

IFLA International Federation of Library Associations and Institutions

IP Internet Protocol

IR information retrieval

IS information system

ISBD International Standard Bibliographic Description

JS JavaScript

MAB Maschinelles Austauschformat für Bibliotheken

MACAO music management tool for content-based analysis and organization

MARC Machine-Readable Cataloging

MDS multi-dimensional scaling

MIDI Musical Instrument Digital Interface

OCR optical character recognition

OMR optical music recognition

OPAC Online Public Access Catalog

PCM pulse code modulation

QBE query-by-example

QBH query-by-humming

QBS query-by-selecting

QBT query-by-tapping

RMI Java Remote Method Invocation

RPC remote procedure call

SOA service-oriented architecture

SOAP Simple Object Access Protocol

SQL Structured Query Language

TCP Transmission Communication Protocol

UI user interface

WSDL Web Service Definition Language

XML Extensible Markup Language

XSS cross-site scripting

Bibliography

[1] E. Allamanche, J. Herre, B. Fröba, and M. Cremer. AudioID: Towards content-based identification of audio material. In *Proceedings of the 110th Audio Engineering Society (AES) Convention*, Amsterdam, Netherlands, 2001.

[2] V. Arifi, M. Clausen, F. Kurth, and M. Müller. Synchronization of music data in score-, MIDI- and PCM-format. *Computing in Musicology*, 13:9–33, 2004.

[3] R. A. Baeza-Yates and B. A. Ribeiro-Neto. *Modern Information Retrieval*. ACM Press, Addison-Wesley, 1999. `http://citeseer.nj.nec.com/433337.html`.

[4] D. Baggi, A. Barate, G. Haus, and L. A. Ludovico. NINA—Navigating and interacting with notation and audio. In *Proceedings of the 2nd International Workshop on Semantic Media Adaptation and Personalization (SMAP)*, pages 134–139, Washington, DC, USA, 2007. IEEE Computer Society. `doi:10.1109/SMAP.2007.28`.

[5] D. Bainbridge, J. Thompson, and I. H. Witten. Assembling and enriching digital library collections. In *Proceedings of the 3rd ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL)*, pages 323–334, Washington, DC, USA, 2003. IEEE Computer Society.

[6] M. A. Bartsch and G. H. Wakefield. Audio thumbnailing of popular music using chroma-based representations. *IEEE Transactions on Multimedia*, 7(1):96–104, Feb. 2005.

[7] J. P. Bello, G. Monti, and M. Sandler. Techniques for automatic music transcription. In *International Symposium on Music Information Retrieval*, pages 23–25, 2000.

[8] D. Bergen. *Issues of Access in the New Information Age*. ERIC Clearinghouse, Washington, DC, USA, 1984.

[9] R. Berndt, I. Blümel, M. Clausen, D. Damm, J. Diet, D. W. Fellner, C. Fremerey, R. Klein, F. Krahl, M. Scherer, I. Sens, V. Thomas, and R. Wessel. Aufbau einer verteilten digitalen Bibliothek für nichttextuelle Dokumente – Ansatz und Erfahrungen des PROBADO Projekts. In *Proceedings of the 5. Konferenz der Zentralbibliothek im Forschungszentrum Jülich WissKom 2010 (poster paper)*, Sept. 2010.

[10] R. Berndt, I. Blümel, M. Clausen, D. Damm, J. Diet, D. W. Fellner, C. Fremerey, R. Klein, F. Krahl, M. Scherer, I. Sens, V. Thomas, and R. Wessel. The PROBADO project—approach and lessons learned in building a digital library system for heterogeneous non-textual documents. In *Proceedings of the 14th European Conference on Digital Libraries (ECDL)*, pages 376–383, Glasgow, UK, Sept. 2010. `doi:10.1007/978-3-642-15464-5_37`.

[11] A. d. Bimbo, S. Gradmann, and Y. Ioannidis. Future research directions. 3rd DELOS brainstorming workshop report. Report, July 2004. `http://www.delos.info/files/pdf/events/2004_Jul_8_10/D8.pdf`.

[12] W. P. Birmingham, K. O'Malley, J. W. Dunn, and R. Scherle. V2V: A second variation on query-by-humming. In *Proceedings of the 3rd ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL)*, pages 380–380, Washington, DC, USA, 2003. IEEE Computer Society.

[13] W. P. Birmingham, B. Pardo, C. Meek, and J. Shifrin. The MusArt music-retrieval system: An overview. *D-Lib Magazine*, 8(2), 2002. `http://www.dlib.org/dlib/february02/birmingham/02birmingham.html`, `doi:10.1045/february2002-birmingham`.

[14] I. Blümel, H. Krottmaier, and R. Wessel. The PROBADO framework: A repository for architectural 3D-models. In *Proceedings of the International Conference on Online Repositories in Architecture*, Venice, FL, USA, Sept. 2008. Fraunhofer irb Verlag.

[15] D. Booth, H. Haas, F. Mccabe, E. Newcomer, M. Champion, C. Ferris, and D. Orchard. Web Services Architecture. Technical report, World Wide Web Consortium, Feb. 2004.

[16] C. L. Borgman. What are digital libraries? Competing visions. *Information Processing and Management*, 35(3):227–243, 1999.

[17] T. Bray, J. Paoli, C. M. Sperberg-Mcqueen, and E. Maler. Extensible Markup Language (XML) 1.0 (Second Edition). W3C Recommendation, Oct. 2000.

[18] D. Byrd and M. Schindele. Prospects for improving OMR with multiple recognizers. In *Proceedings of the 7th International Conference on Music Information Retrieval (ISMIR)*, pages 41–46, Victoria, Canada, 2006.

[19] L. Candela, D. Castelli, P. Pagano, C. Thanos, I. Pisa, Y. Ioannidis, G. Koutrika, G. Athens, S. Ross, H. J. Schek, and Others. Setting the foundations of digital libraries: The DELOS manifesto. *D-Lib Magazine*, 13(3-4).

[20] P. Cano, E. Battle, T. Kalker, and J. Haitsma. A review of audio fingerprinting. In *Proceedings of the 5th IEEE Workshop on Multimedia Signal Processing (MMSP)*, St. Thomas, Virgin Islands, USA, 2002.

[21] J. Cardoso, A. P. Barros, N. May, and U. Kylau. Towards a unified service description language for the internet of services: Requirements and first developments. In *IEEE SCC*, pages 602–609, Miami, FL, USA, July 2010. IEEE Computer Society.

[22] P. P.-s. Chen. The entity-relationship model—toward a unified view of data. *ACM Transactions on Database Systems*, 1(1):9–36, 1976. `doi:10.1145/320434.320440`.

[23] G. S. Choudhury, T. DiLauro, M. Droettboom, I. Fujinaga, B. Harrington, and K. MacMillan. Optical music recognition system within a large-scale digitization project. In *Proceedings of the 1st International Conference on Music Information Retrieval (ISMIR)*, Plymouth, MA, USA, 2000.

[24] M. Clausen. Lecture notes for „Grundlagen des Multimediaretrievals“. Department of Computer Science III, University of Bonn, Germany, 2010.

[25] M. Clausen and F. Kurth. A unified approach to content-based and fault tolerant music identification. *IEEE Transactions on Multimedia*, 6(5):717–731, Oct. 2004.

[26] M. Clausen and M. Müller. Lecture notes for „Inhaltsbasiertes Multimediaretrieval“. Department of Computer Science III, University of Bonn, Germany, 2007.

[27] G. Cleveland. Digital libraries: Definitions, issues and challenges. *IFLANET*, Mar. 1998. `http://www.ifla.org/udt/op/udtop8/udtop8.htm`.

[28] T. F. Cox and M. A. A. Cox. *Multidimensional Scaling*, volume 59 of *Monographs on Statistics and Applied Probability*. Chapman & Hall, London, UK, 1994.

[29] A. D'Aguanno and G. Vercellesi. Automatic music synchronization using partial score representation based on IEEE 1599. *Journal of Multimedia*, 4(1):19–24, 2009.

[30] D. Damm. Textbasierte Musiksuche im Rahmen des SyncPlayer-Frameworks. Master's thesis, Department of Computer Science III, University of Bonn, Germany, 2007.

[31] D. Damm, C. Fremerey, F. Kurth, M. Müller, and M. Clausen. Multimodal presentation and browsing of music. In *Proceedings of the 10th International Conference on Multimodal Interfaces (ICMI 2008)*, Chania, Crete, Greece, 2008.

[32] D. Damm, C. Fremerey, F. Kurth, M. Müller, and M. Clausen. SyncPlayer – Multimodale Wiedergabe, Navigation und Suche in heterogenen digitalen Musikkollektionen. In *Proceedings of the Workshop on Lernen, Wissensentdeckung und Adaptivität (LWA)*, pages 13–20, Würzburg, Germany, 2008.

[33] D. Damm, C. Fremerey, F. Kurth, M. Müller, and M. Clausen. SyncPlayer – Multimodale Wiedergabe, Suche und Navigation in digitalen Musikkollektionen. In *Proceedings of the 34. Deutsche Jahrestagung für Akustik (DAGA)*, 2008.

[34] D. Damm, C. Fremerey, V. Thomas, M. Clausen, F. Kurth, and M. Müller. A digital library framework for heterogeneous music collections: from document acquisition to cross-modal interaction. *International Journal on Digital Libraries*, 12(2-3):53–71, Aug. 2012. `doi:10.1007/s00799-012-0087-y`.

[35] D. Damm, F. Kurth, C. Fremerey, and M. Clausen. A concept for using combined multimodal queries in digital music libraries. In *Proceedings of the 13th European Conference on Research and Advanced Technology for Digital Libraries (ECDL)*, pages 261–272, Kanoni, Corfu, Greece, 2009. `doi:10.1007/978-3-642-04346-8_26`.

[36] I. Damnjanovic, J. Reiss, and D. Barry. Enabling access to sound archives through integration, enrichment , and retrieval. In *Proceedings of the 2008 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1597–1598, Hannover, Germany, 2008. `doi:10.1109/ICME.2008.4607756`.

[37] R. B. Dannenberg and C. Raphael. Music score alignment and computer accompaniment. In Pardo [101], pages 38–43. `doi:10.1145/1145287.1145311`.

[38] J. Diet and F. Kurth. The PROBADO music repository at the Bavarian State Library. In *Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR)*, pages 501–504, Vienna, Austria, Sept. 2007.

[39] S. Dixon and G. Widmer. MATCH: A music alignment tool chest. In *Proceedings of the 6th International Conference on Music Information Retrieval (ISMIR)*, London, UK, 2005.

[40] J. W. Dunn, D. Byrd, M. Notess, J. Riley, and R. Scherle. Variations2: Retrieving and using music in an academic setting. In Pardo [101], pages 53–58. `doi:10.1145/1145287.1145314`.

[41] European Union. EUROPEANA, 2007. `http://www.europeana.eu/portal/index.html`.

[42] C. Faloutsos and K.-I. Lin. FastMap: A fast algorithm for indexing, data-mining and visualization of traditional and multimedia datasets. In M. J. Carey and D. A. Schneider, editors, *Proceedings of the 1995 ACM SIGMOD International Conference on Management of Data*, pages 163–174, San Jose, CA, USA, 1995. ACM. `doi:10.1145/223784.223812`.

[43] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, and T. B. Lee. Hypertext transfer protocol – HTTP/1.1. Technical report, June 1999. `http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.50.4528`.

[44] E. A. Fox, R. M. Akscyn, R. K. Furuta, and J. J. Leggett. Digital libraries. *Communications of the ACM*, 38(4):22–28, Apr. 1995. `http://doi.acm.org/10.1145/205323.205325`, `doi:http://doi.acm.org/10.1145/205323.205325`.

[45] C. Fremerey. SyncPlayer—a framework for content-based music navigation. Master's thesis, Department of Computer Science III, University of Bonn, Germany, 2006.

[46] C. Fremerey. *Automatic Organization of Digital Music Documents – Sheet Music and Audio*. PhD thesis, Universität Bonn, Institut für Informatik III, 2010.

[47] C. Fremerey, M. Clausen, S. Ewert, and M. Müller. Sheet music-audio identification. In *Proceedings of the 10th International Conference on Music Information Retrieval (ISMIR)*, pages 645–650, Kobe, Japan, Oct. 2009.

[48] C. Fremerey, D. Damm, M. Müller, F. Kurth, and M. Clausen. Handling scanned sheet music and audio recordings in digital music libraries. In *Proceedings of the International Conference on Acoustics NAG/DAGA*, Rotterdam, Netherlands, 2009.

[49] C. Fremerey, M. Müller, and M. Clausen. Towards bridging the gap between sheet music and audio. In E. Selfridge-Field, F. Wiering, and G. A. Wiggins, editors, *Knowledge representation for intelligent music processing*, number 09051 in Dagstuhl Seminar Proceedings, Dagstuhl, Germany, Jan. 2009. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, Germany. `http://drops.dagstuhl.de/opus/volltexte/2009/1965`.

[50] C. Fremerey, M. Müller, and M. Clausen. Handling repeats and jumps in score-performance synchronization. In *Proceedings of the 11th International Conference on Music Information Retrieval (ISMIR)*, Utrecht, Netherlands, 2010.

[51] C. Fremerey, M. Müller, F. Kurth, and M. Clausen. Automatic mapping of scanned sheet music to audio recordings. In *Proceedings of the 9th International Conference on Music Information Retrieval (ISMIR)*, pages 413–418, Philadelphia, USA, Sept. 2008.

[52] M. Good. MusicXML: An internet-friendly format for sheet music. In *Proceedings XML Conference and Exposition*, Orlando, FL, USA, 2001. `http://www.idealliance.org/papers/xml2001/papers/html/03-04-05.html`.

[53] Google Inc. Google Book Search, 2007. `http://books.google.com`.

[54] M. Gorman, P. W. Winkler, and A. L. Association. *Anglo-American cataloguing rules*. American Library Association, 1978. `http://books.google.it/books?id=uLhAAAAAMAAJ`.

[55] M. Goto. A chorus-section detecting method for musical audio signals. In *Proceedings of the IEEE Internatinal Conference on Acoustics, Speech, and Signal Processing ICASSP*, pages 437–440, Hong Kong, China, 2003.

[56] Gracenote. Music Search, 2008. `http://www.gracenote.com/`.

[57] C. Grande and A. Belkin. The development of the notation interchange file format. *Computer Music Journal*, 20(4):33–43, 1996.

[58] W. Grosso. *Java RMI*. Number October. O'Reilly Media, Inc., 2001.

[59] M. Gudgin, M. Hadley, N. Mendelsohn, J.-J. Moreau, and H. F. Nielsen. Soap version 1.2 part 1: Messaging framework. W3C Recommendation, June 2003. `http://www.w3.org/TR/soap12-part1/`.

[60] A. Hankinson, L. Pugin, and I. Fujinaga. Interfaces for document representation in digital music libraries. In *Proceedings of the 10th International Conference on Music Information Retrieval (ISMIR)*, pages 39–44, Kobe, Japan, 2009. `http://ismir2009.ismir.net/proceedings/OS1-3.pdf`.

[61] M. D. Hansen. *SOA Using Java Web Services*. Prentice Hall, May 2007. `http://www.worldcat.org/isbn/0130449687`.

[62] N. Hu, R. Dannenberg, and G. Tzanetakis. Polyphonic audio matching and alignment for music retrieval. In *Proceedings of the 4th IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, New Paltz, NY, USA, Oct. 2003.

[63] D. M. Huber. *The MIDI Manual*. Focal Press, 1999.

[64] IFLA Study Group on the Functional Requirements for Bibliographic Records. Functional requirements for bibliographic records: Final report. *UBCIM Publications, New Series*, 19, 1998. `http://www.ifla.org/VII/s13/frbr/frbr.htm`.

[65] Indiana University Digital Library Program. Variations3: An integrated digital library and learning system for the music community. Website, 2005. Available online at `http://www.dlib.indiana.edu/projects/variations3/docs/Indiana_University_IMLS_2005-02-01.pdf`; visited on May 19th 2011. `http://www.dlib.indiana.edu/projects/variations3/docs/Indiana_University_IMLS_2005-02-01.pdf`.

[66] International Federation of Library Associations and Institutions. ISBD(G): General international standard bibliographic description. 2004 revision. Recommended by the ISBD Review Group. Approved by the Standing Committee of the IFLA Section on Cataloguing. 2004. Available online at http://www.ifla.org/files/assets/cataloguing/isbd/isbd-g_2004.pdf; visited on June 6th 2011. http://www.ifla.org/files/assets/cataloguing/isbd/isbd-g_2004.pdf.

[67] Y. E. Ioannidis, D. Maier, S. Abiteboul, P. Buneman, S. B. Davidson, E. A. Fox, A. Y. Halevy, C. A. Knoblock, F. Rabitti, H.-J. Schek, and G. Weikum. Digital library information-technology infrastructures. *International Journal on Digital Libraries*, 5(4):266–274, 2005. 10.1007/s00799-004-0094-8. http://dx.doi.org/10.1007/s00799-004-0094-8.

[68] G. Jones. SharpEye music reader, 2008. http://www.visiv.co.uk/.

[69] B. Kahle. Internet Archive, 1996. http://www.archive.org/index.php.

[70] E. Keogh. Exact indexing of dynamic time warping. In *Proceedings of the 28th International Conference on Very Large Data Bases (VLDB)*, pages 406–417, Hong Kong, China, 2002.

[71] A. Klapuri and M. Davy, editors. *Signal Processing Methods for Music Transcription*. Springer, New York, 2006.

[72] E. Krajewski. DE-PARCON Softwaretechnologie, 2008. http://www.de-parcon.de/.

[73] H. Krottmaier, F. Kurth, T. Steenweg, H.-J. Appelrath, and D. Fellner. PROBADO—a generic repository integration framework. In *Proceedings of the 11th European Conference on Research and Advanced Technology for Digital Libraries (ECDL)*, Budapest, Hungary, Sept. 2007.

[74] J. B. Kruskal. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, 29(1):1–27, Mar. 1964. doi:10.1007/BF02289565.

[75] J. B. Kruskal and M. Wish. Multidimensional scaling. *Quantitative Applications in the Social Sciences*, 11(7), 1978.

[76] F. Kurth, D. Damm, C. Fremerey, M. Müller, and M. Clausen. A framework for managing multimodal digitized music collections. In *Proceedings of the 12th European Conference on Research and Advanced Technology for Digital Libraries (ECDL)*, pages 334–345, Aarhus, Denmark, 2008. doi:10.1007/978-3-540-87599-4_35.

[77] F. Kurth and M. Müller. Efficient index-based audio matching. *IEEE Transactions on Audio, Speech, and Language Processing (TASLP)*, 16(2):382–395, Feb. 2008.

[78] F. Kurth, M. Müller, D. Damm, C. Fremerey, A. Ribbrock, and M. Clausen. SyncPlayer—an advanced system for content-based audio access. In *Proceedings of the 7th International Conference on Music Information Retrieval (ISMIR)*, London, UK, 2005.

[79] F. Kurth, M. Müller, and C. Fremerey. Audio Matching für symbolische Musikdaten. In *Fortschritte der Akustik, Tagungsband der DAGA*, Mar. 2007. http://www.cs.uni-bonn.de/~meinard/publications/07_KuMuFr_DAGA_SymbAudioMatch.pdf.

[80] F. Kurth, M. Müller, C. Fremerey, Y. Chang, and M. Clausen. Automated synchronization of scanned sheet music with audio recordings. In *Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR)*, pages 261–266, Vienna, Austria, Sept. 2007.

[81] Y. Lafon. Web Services. `http://www.w3.org/2002/ws/`.

[82] C. Landone, J. Harrop, and J. Reiss. Enabling access to sound archives through integration, enrichment and retrieval: the EASAIER project. In *Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR)*, pages 159–160, Vienna, Austria, Sept. 2007.

[83] V. I. Levenshtein. Binary codes capable of correcting deletions, insertions, and reversals. Technical Report 8, 1966.

[84] L. A. Ludovico. IEEE 1599: a multi-layer approach to music description. *Journal of Multimedia*, 4(1):9–14, 2009. `http://www.academypublisher.com/jmm/vol04/no01/jmm04010914.pdf`.

[85] N. C. Maddage, C. Xu, M. S. Kankanhalli, and X. Shao. Content-based music structure analysis with applications to music semantics understanding. In *Proceedings of the 12th annual ACM International Conference on Multimedia*, pages 112–119, New York, NY, USA, 2004. `doi:http://doi.acm.org/10.1145/1027527.1027549`.

[86] G. Marchionini. *Information seeking in electronic environments*. Cambridge University Press, New York, NY, USA, 1995.

[87] MIDI Manufacturers Association. The complete MIDI 1.0 detailed specification. 1996. `http://www.midi.org`.

[88] M. Müller. *Information Retrieval for Music and Motion*. Springer, 2007.

[89] M. Müller and D. Appelt. Path-constrained partial music synchronization. In *Proceedings of the 34th International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume 1, pages 65–68, Las Vegas, Nevada, USA, Apr. 2008.

[90] M. Müller and M. Clausen. Transposition-invariant self-similarity matrices. In *Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR)*, pages 47–50, Vienna, Austria, Sept. 2007.

[91] M. Müller and F. Kurth. Towards structural analysis of audio recordings in the presence of musical variations. *EURASIP Journal on Applied Signal Processing*, 2007(89686):18, Jan. 2007.

[92] M. Müller, F. Kurth, D. Damm, C. Fremerey, and M. Clausen. Lyrics-based audio retrieval and multimodal navigation in music collections. In *Proceedings of the 11th European Conference on Research and Advanced Technology for Digital Libraries (ECDL)*, pages 112–123, Budapest, Hungary, Sept. 2007. `doi:10.1007/978-3-540-74851-9_10`.

[93] M. Müller, F. Kurth, and T. Röder. Towards an efficient algorithm for automatic score-to-audio synchronization. In *Proceedings of the 5th International Conference on Music Information Retrieval (ISMIR)*, pages 365–372, Barcelona, Spain, Oct. 2004.

[94] M. Müller, H. Mattes, and F. Kurth. An efficient multiscale approach to audio synchronization. In *Proceedings of the 7th International Conference on Music Information Retrieval (ISMIR)*, pages 192–197, Victoria, Canada, 2006.

[95] E. Newcomer and G. Lomow. *Understanding SOA with Web Services (Independent Technology Guides)*. Addison-Wesley Professional, 2004.

[96] OASIS. Reference model for service oriented architecture 1.0. *Architecture*, 2000(October):1–31, 2006. `http://docs.oasis-open.org/soa-rm/v1.0/`.

[97] O. C. Oberhauser. Card-image public access catalogues (CIPACs): An international survey. Program: Electronic library and information systems 37(2). In *Libri: International Journal of Libraries and Information Services*, 2003.

[98] N. Orio. Alignment of performances with scores aimed at content-based music access and retrieval. In *Proceedings of the 6th European Conference on Research and Advanced Technology for Digital Libraries (ECDL)*, pages 479–492, Rome, Italy, 2002.

[99] N. Orio, S. Lemouton, and D. Schwarz. Score following: State of the art and new developments. In *Proceedings of the Conference of New Interfaces for Musical Expression (NIME)*, pages 36–41, Montreal, CA, 2003.

[100] B. Pardo. Introduction. [101], pages 28–31. `doi:10.1145/1145287.1145309`.

[101] B. Pardo, editor. *Special Issue: Music Information Retrieval*, volume 49. ACM, New York, NY, USA, Aug. 2006. `doi:10.1145/1145287`.

[102] G. Peeters, A. L. Burthe, and X. Rodet. Toward automatic music audio summary generation from signal analysis. In *Proceedings of the 3th International Conference on Music Information Retrieval (ISMIR)*, Paris, France, 2002.

[103] J. Pickens, J. P. Bello, G. Monti, T. Crawford, M. Dovey, and M. Sandler. Polyphonic score retrieval using polyphonic audio queries: A harmonic modeling approach. In *Proceedings of the 3rd International Conference on Music Information Retrieval (ISMIR)*, pages 140–149, Paris, France, 2002.

[104] A. Pinto. Multi-model music content description and retrieval using IEEE 1599 XML standard. *Journal of Multimedia*, 4(1):30–39, 2009. `http://www.academypublisher.com/jmm/vol04/no01/jmm04013039.pdf`.

[105] J. C. Platt. Fast embedding of sparse music similarity graphs. *Advances in Neural Information Processing Systems*, 16:571–578, 2004.

[106] J. C. Platt. FastMap, MetricMap, and Landmark MDS are all nyström algorithms. In *Proceedings of the 10th International Workshop on Artificial Intelligence and Statistics*, pages 261–268, 2005.

[107] J. Postel. RFC 791: Internet Protocol. 1981.

[108] J. Postel. RFC 793: Transmission Control Protocol. 1981.

[109] L. Prechelt and R. Typke. An interface for melody input. *ACM Transactions on Computer-Human Interaction*, 8:133–149, 1998.

[110] L. Prechelt and R. Typke. An interface for melody input. *ACM Transactions on Computer-Human Interaction*, 8(2):133–149, 2001.

[111] M. Quist and G. Yona. Distributional scaling: An algorithm for structure-preserving embedding of metric and nonmetric spaces. *Journal of Machine Learning Research*, 5:399–420, Apr. 2004.

[112] C. Raphael. A hybrid graphical model for aligning polyphonic audio with musical scores. In *Proceedings of the 5th International Conference on Music Information Retrieval (ISMIR)*, Barcelona, Spain, 2004.

[113] C. Raphael. A hybrid graphical model for aligning polyphonic audio with musical scores. In *Proceedings of the 5th International Conference on Music Information Retrieval (ISMIR)*, Barcelona, Spain, 2004.

[114] A. Rauber and M. Frühwirth. Automatically analyzing and organizing music archives. In *Proceedings of the 5th European Conference on Digital Libraries (ECDL)*, Springer Lecture Notes in Computer Science, Darmstadt, Germany, Sept. 2001. Springer. `http://www.ifs.tuwien.ac.at/ifs/research/publications.html`.

[115] B. Schölkopf. The kernel trick for distances. In *Proceedings of the 14th Annual Conference on Neural Information Processing Systems (NIPS)*, pages 301–307, Denver, CO, USA, 2000.

[116] E. Selfridge-Field, editor. *Beyond MIDI: The Handbook of Musical Codes*. MIT Press, Cambridge, MA, USA, 1997.

[117] R. K. Sharma and K. R. Vishwanathan. Digital libraries: development and challenges. *Library Review*, 50(1):10–16, 2001. `http://www.emeraldinsight.com/10.1108/00242530110363190`.

[118] R. N. Shepard. Multidimensional-scaling, tree-fitting, and clustering. *Science*, 210(4468):390–398, 1980. `doi:10.1126/science.210.4468.390`.

[119] D. Soergel. A framework for digital library research: Broadening the vision. *D-Lib Magazine*, 8(12), 2002.

[120] F. Soulez, X. Rodet, and D. Schwarz. Improving polyphonic and poly-instrumental music to score alignment. In *Proceedings of the 4th International Conference on Music Information Retrieval (ISMIR)*, Baltimore, MD, USA, 2003.

[121] I. S. H. Suyoto, A. L. Uitdenbogerd, and F. Scholer. Searching musical audio using symbolic queries. *IEEE Transactions on Audio, Speech, and Language Processing (TASLP)*, 16(2):372–381, 2008. `doi:10.1109/TASL.2007.911644`.

[122] V. Thomas, D. Damm, C. Fremerey, M. Clausen, F. Kurth, and M. Müller. Probado music: a multimodal online music library. In *Proceedings of the International Computer Music Conference (ICMC)*, Ljubljana, Slovenia, 2012.

[123] V. Thomas, C. Fremerey, D. Damm, and M. Clausen. SLAVE: A score-lyrics-audio-video-explorer. In *Proceedings of the 10th International Conference on Music Information Retrieval (ISMIR)*, Kobe, Japan, Oct. 2009.

[124] V. Thomas, C. Wagner, and M. Clausen. OCR-based post-processing of OMR for the recovery of transposing instruments in complex orchestral scores. In *Proceedings of the International Society for Music Information Retrieval Conference(ISMIR)*, pages 411–416, Miami, USA, 2011.

[125] W. S. Torgerson. Multidimensional scaling: I. Theory and method. *Psychometrika*, 17(4):401–419, Dec. 1952. doi:10.1007/BF02288916.

[126] R. J. Turetsky and D. P. Ellis. Force-aligning MIDI syntheses for polyphonic music transcription generation. In *Proceedings of the 4th International Conference on Music Information Retrieval (ISMIR)*, Baltimore, MD, USA, 2003.

[127] R. J. Turetsky and D. P. W. Ellis. Ground-truth transcriptions of real music from force-aligned MIDI syntheses. In *Proceedings of the 4th International Conference on Music Information Retrieval (ISMIR)*, Baltimore, MD, USA, 2003. http://ismir2003.ismir.net/papers/Turetsky.PDF.

[128] R. Typke, F. Wiering, and R. C. Veltkamp. A survey of music information retrieval systems. In *Proceedings of the 6th International Conference on Music Information Retrieval (ISMIR)*, pages 153–160, 2005. http://ismir2005.ismir.net/proceedings/1020.pdf.

[129] Union der deutschen Akademien der Wissenschaften. Neue Mozart Ausgabe, 2007. http://www.nma.at/.

[130] University of Chicago Library. Chopin Early Edition, 2004. http://chopin.lib.uchicago.edu/.

[131] University of Rochester Libraries. UR Research—Sibley Music Library, 2009. https://urresearch.rochester.edu/home.action.

[132] A. L.-C. Wang. An industrial-strength audio search algorithm. In *Proceedings of the 4th International Conference on Music Information Retrieval (ISMIR)*, Baltimore, MD, USA, 2003. http://www.ee.columbia.edu/~dpwe/papers/Wang03-shazam.pdf.

[133] Y. Wang, M.-Y. Kan, T. L. Nwe, A. Shenoy, and J. Yin. LyricAlly: automatic synchronization of acoustic musical signals and textual lyrics. In *Proceedings of the 12th annual ACM International Conference on Multimedia*, pages 212–219, New York, NY, USA, 2004. ACM Press. http://doi.acm.org/10.1145/1027527.1027576.

[134] D. J. Waters. What are digital libraries? CLIR Issues 4, Council on Library and Information Resources, 1998. http://www.clir.org/pubs/issues/issues04.html.

[135] Web Services Description Working Group. Web Services Description Language (WSDL) Version 2.0 Part 1: Core Language. Technical report, 2007. http://www.w3.org/TR/wsdl20/.

[136] H. Weber and W. Mattauch. Forschungsarbeiten im THESEUS-Umfeld in Japan, 2009. https://www.theseus.joint-research.org/assets/publikationsreihe/THESEUSJapan-v03.pdf.

[137] Wiener Wissenschafts-, Forschungs- und Technologiefonds. Schubert-Autographe. `http://www.schubert-online.at/`.

[138] I. H. Witten, R. J. McNab, S. J. Boddie, and D. Bainbridge. Greenstone: A comprehensive open-source digital library software system. In *Proceedings of the 5th International Conference on Digital Libraries (ECDL)*, Darmstadt, Germany, 2000. `http://citeseer.ist.psu.edu/witten99greenstone.html`.

[139] I. H. Witten, A. Moffat, and T. C. Bell. *Managing Gigabytes*. Van Nostrand Reinhold, 2nd edition, 1999.

[140] Working Group on the General International Standard Bibliographic Description. *ISBD(G): General International Standard Bibliographic Description. Annotated Text. Prepared by the Working Group on the General International Standard Bibliographic Description set up by the IFLA Committee on Cataloguing*. IFLA International Office for UBC, London, UK, 1977.

[141] M. M. Yee. FRBRization: a method for turning online public finding lists into online public catalogs. *Information Technology and Libraries*, 24(3), 2005. `http://repositories.cdlib.org/postprints/715/`.

Erklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbständig verfasst, noch nicht anderweitig für andere Prüfungszwecke vorgelegt, keine anderen als die angegebenen Quellen und Hilfsmittel benutzt, sowie wörtliche und sinngemäße Zitate als solche gekennzeichnet habe.

Bonn, ______________________________ ______________________________

(Datum) (Unterschrift)