MEC’21
Proceedings
Music Encoding Conference Proceedings
2021

19–22 July, 2021
University of Alicante (Spain): Onsite & Online

Edited by
Stefan Münnich and David Rizo
Foreword

Keynote I
Álvaro Torrente, Ana Llorens
The Musicology Lab: Teamwork and the Musicological Toolbox

Articles

Emilio Ros-Fábregas
Encoded Spanish Music Heritage through Verovio: The Online Platforms Fondo de Música Tradicional IMF–CSIC and Books of Hispanic Polyphony IMF–CSIC

Werner Goebl, David M. Weigl
Alleviating the Last Mile Of Encoding: The mei-friend Package for the Atom Text Editor

Klaus Rettinghaus, Alexander Pacha
Building a Comprehensive Sheet Music Library Application

David Lewis, Kevin R. Page, Laurence Dreyfus
Lohengrin TimeMachine: Musicological Multimedia Made with MELD

Axel Berndt
Music Performance Markup: Format and Software Tools Report

Chiara Martignano, Antonio Calvia, Michele Epifani
Tools and Perspectives for a Digital Critical Edition of Fourteenth-Century Polyphonic Music

Sara González Gutiérrez, Javier Merchán Sánchez Jara, María Navarro Cáceres
Encoding Traditional Spanish Music for Pedagogical Purposes through MEI: Challenges and Opportunities

Johannes Kepper, Susanne Cox
Encoding Genetic Processes II

Clemens Gubsch, Dennis Ried
METAdata and metaDATA

Kristina Richts-Matthaei, Irmlind Capelle
United, Linked, Connected – A Data Model for the Inventory of the Former Detmold Court Theatre (1825–1875), or: How Library Inventory History Can also Be Told

Kevin R. Page, Johannes Kepper, Christine Siegert, Andrew Hankinson, David Lewis
Beethoven in the House: Digital Studies of Domestic Music Arrangements
Posters

Maria João Albuquerque, H. Sofia Pinto, José Luís Borbinha, Luís Filipe Santos, Tiago Teixeira 125
The PROFMUS Application: Development, Status, and Future Progress

Mark Robert Haigh Gotham, Peter Jonas 131
The OpenScore Lieder Corpus

Kristina Richts-Matthaei, Martin Albrecht-Hohmaier, Daniel Röwenstrunk, Andreas Münzmay 137
MEI Meets NFDI4Culture

Johannes Hentschel, Fabian C. Moss, Andrew McLeod, Markus Neuwirth, Martin Rohrmeier 143
Towards a Unified Model of Chords in Western Harmony

Justin Lind 151
Visualizing Harmony Using Chordal Glyphs and Color Mapping

Anna Matuszewska, Christoph Seibert 159
Diagrammatic Analysis of J.S. Bach's The Well-Tempered Clavier Fugues, BWV 846–851

Benjamin W. Bohl, Johannes Kepper 167
ODD Structures and Where to Find Them

Nikolaos Siklafidis, Maria Alexandru 173
Font Design of Psaltic (Byzantine) Notation for Greek Musical Repertoires

Joshua Stutter 181
Annotation of Medieval Music Facsimiles Using ‘Good Enough’ OMR

Florian Zwißler, Philip Schwarz Bauer, Michael Oehler 189
Encoding Scores for Electronic Music

Panel

Karen Desmond, Laurent Pugin, Juliette Regimbal, David Rizo, Craig Sapp, Martha E. Thomae 197
Encoding Polyphony from Medieval Manuscripts Notated in Mensural Notation

Keynote II

Pip Willcox 221
Automatic for the People: Archives and the Future

Appendix

Conference schedule 223
Foreword

Stefan Mühnic
University of Basel, Switzerland
stefan.muennich@unibas.ch

David Rizo
University of Alicante, ISEA.CV, Spain
drizo@dlsi.ua.es

We are pleased to present the proceedings of the Music Encoding Conference 2021, which was held both online and on-site in Alicante, Spain, on July 19–22, 2021.

For some three thousand years, humankind has been trying to find the best way to encode music for its transmission, and at least seventy of those years have been dedicated to finding the best digital formats. This fact shows that the task of encoding music is actually very complex. The academic community, real people, need to share their thoughts, problems, and projects.

In 2020, the emergence of a pandemic the likes of which our world has not seen for about a century challenged such an exchange, but with a huge organizational effort, MEC2020 was moved entirely online at short notice. We all expected 2021 to be a better year in terms of vaccination rates, the number of infections, and the relaxation of travel restrictions, so a hybrid conference was planned for the third week of May as a link between the online-only mode in 2020 and a re-embracement of in-person events hopefully in the coming years. As the forecasts didn’t materialize, the conference was postponed to July. Many participants were interested in attending the conference in person. In the end, however, only a few people were able to travel to Alicante, and most participants registered to attend the conference online. For all attendees, whether online or on-site, we tried to organize the conference such that it would fulfill its mission of bringing people and ideas closer together. This is the reason why, for example, with more than 160 registered researchers from Brazil to Japan and Australia, from Norway to Israel, sessions started at only 3 p.m. local time to create a suitable schedule in which everyone could simultaneously participate.

Organizing a conference in such uncertain times was a real challenge. Most of the decisions remained open until just a few days before the event. The possibility of too many or too few people being present in person shaped the whole program. Given that fact, it was encouraging to see that five workshops could be held with about ninety participants, sixty of whom were enrolled in both sessions on the pre-conference day, and three of the convenors streamed their workshops from Alicante with attendees both in the same room and online. The music event that had initially been planned to be a live performance was rearranged as an online presentation of a play that has been annually celebrated near Alicante since the Middle Ages: the Misteri d’Elx, which was declared one of the Masterpieces of the Oral and Intangible Heritage of Humanity by UNESCO in 2001.

Although most of us are used to video calls and online meetings and conferences nowadays, hybrid events are not yet so common and familiar. To experience a partly physical event, two online platforms were used for the scientific part of the conference: ZOOM for the presentations and WONDER.ME for all parallel discussion sessions, breaks, and social events. Two SLACK channels were established for asynchronous communication, announcements, and urgent questions throughout the event. An internal conference website was made available before and during the conference to all registered participants to inform them in real time about all planned events, including about updated time slots and the platforms used, and it provided links to the presentations and PDFs of the contributions. These platforms were coordinated by the organizing chairs and volunteers from the so-called Control Room, a dedicated help desk for monitoring and supporting the conference.

In order to maintain consistency and to simplify visual orientation and the navigation of all the virtual event platforms, a complete visual design project was developed by students of the Alicante Art and Design School (EASDA). They created posters, badges, and all the printed materials for in-person participants, and the banners and videos for the online events.

But the conference was not only challenging for the organizers; it was also a challenge for the presenters. MEC2021 was the first Music Encoding Conference to ask for full paper submissions. Notwithstanding this fact and the still ongoing pandemic, eighteen long papers (oral presentations) and six short papers (posters) were submitted, of which the Program Committee ultimately accepted eleven as long papers and ten as short papers for presentation at the conference and for inclusion in this volume of the conference proceedings. The topics provide a cross-section of the prolific research activities in the field, ranging from Greek neumes to
electronic music, from canonical monuments such as Beethoven or Bach to previously forgotten pieces and traditions from the Iberian Peninsula, from color design to Linked Data or neural networks, from the latest developments in tools and applications to philological issues and cutting-edge approaches to digital music data. These paper contributions are framed by the keynotes held by Álvaro Torrente with Ana Llorens and by Pip Willcox. Additionally, this volume contains a written version of the panel discussion conducted by Karen Desmond and her colleagues at MEC2021. The spelling of these proceedings follows American English, except for contributions from British or UK-associated authors.

The paper “Alleviating the Last Mile of Encoding: The mei-friend Package for the Atom Text Editor” by Werner Goebl and David M. Weigl was awarded best paper, and “The OpenScore Lieder Corpus” by Mark Robert Haigh Gotham and Peter Jonas was awarded best poster. These awards were determined by online voting by attendees during the conference.

In our eyes, the acronym MEC not only stands for academic exchange about music encoding and digital music technologies at the annual conference; MEC stands even more for a huge, inclusive community event that features the annual gathering of the Music Encoding Community.

It has been a great honor and pleasure to be part of and contribute to this large, active community as chairs of the Organizing Committee and Program Committee, but a conference like this requires teamwork and collective effort. So we would like to thank the many helping hands in the spotlight and behind the scenes, without whose support and hard work MEC2021 would not have been possible: the members of the local organizing team (Luísa Micó, José Manuel Iñesta, Jorge Calvo-Zaragoza, José Javier Valero; the Control Room and help desk María Alfaro, Francisco José Castellanos, Antonio Ríos; the design team with Alicia Orts, Isabel Llorca, Aída Cabañero, Naomi Barrachina, and Luis González, supervised by Esther de las Heras; the administration staff Antonio Antón and Ana Isabel García); the members of the Program Committee (Daniel Bangert, Marie Destandau, Giuliano Di Bacco, Sophia Dörner, Julia Flanders, Jan Hajić, jr., Kristin Herold, Craig Sapp, Martha E. Thomae) for their tireless and extensive commitment; everyone involved in the virtual MEC2020, all of whom so readily and generously shared their experience and expertise (Anna Kijas and Richard Freedman and their teams for the local and program organization, and Elsa De Luca, Julia Flanders, and Irmlind Capelle for the MEC2020 proceedings); the MEI Board members – with special thanks to the then Administrative Chair Elsa De Luca and the Technical Co-Chair Johannes Kepper –, who have been a very important support and point of contact for us throughout; our invited keynote speakers (Álvaro Torrente with Ana Llorens and their team, and Pip Willcox) for their thought-provoking inputs to the conference; all the workshop convenors for sharing their expertise with the community and all the session chairs for guiding the scientific program so smoothly; and Rubén Pacheco for his marvelous introduction to the Misteri d'Elx. The fifty-seven authors of these proceedings cannot be thanked enough, since they not only took up the challenge of submitting full papers despite their busy schedules and despite a year full of uncertainties and physical and mental exhaustion, but also because they accompanied the editorial process of this volume with responsiveness, flexibility, and patience; and last but not least, a special thanks to our numerous private and institutional sponsors, including the University of Alicante and its Instituto de Investigación Informática, the Instituto Superior de Enseñanzas Artísticas de la Comunidad Valenciana, the Conselleria de Innovación, Universidades, Ciencia y Sociedad Digital de la Generalitat Valenciana, the Spanish Ministry HISPAMUS project TIN2017-86576-R, the MultiScore Project, I+D+i PID2020-118447RA-I00, funded by MCIN/AEI/10.13039/50110001103, for their support to the local organization, and the Social Sciences and Humanities Research Council (SSHRC) of the Government of Canada, whose generous support allowed us to fully reimburse all student bursary applications this year.

Sophia Dörner (Berlin) is to be thanked for her priceless help in preparing these proceedings.

April 2022

For the Program Committee
Stefan Münnich

For the Organizing Team
David Rizo
The Musicology Lab: Teamwork and the Musicological Toolbox

Álvaro Torrente
Instituto Complutense de Ciencias Musicales
atorrente@iccmu.es

Ana Llorens
Universidad Complutense de Madrid
allorens@ucm.es

Abstract

Musicology is a small discipline within the wide spectrum of human knowledge, yet it is already divided into various branches, each with its own societies, conferences, journals, jargons, degrees, prejudices, ..., and jobs. Although they share their object of investigation – “the art of music as a physical, psychological, aesthetic, and cultural phenomenon” [4, p. 153] –, these branches very often ignore one another. Research in musicology is mostly a solitary task, as investigations, papers, and publications are commonly signed by single authors, in contrast with STEM disciplines where teamwork is the rule. This is in part the result of tradition – the “Musicological Toolbox” [22] – but also the aftermath of the job market and financing programs.

Large funding schemes such as the European Research Council (ERC) grants are becoming a major disruptive factor in many disciplines in the humanities, including musicology. Scholars in all fields now have the opportunity to build research teams, and most of their members receive their salaries to exclusively work on the project. In other words, we are starting to build what could be called a Musicology Lab, learning along the way how teamwork is reshaping and transforming the Musicological Toolbox, the look and feel of our discipline, the way we work as well as the way we publish and disseminate our results.

This paper presents some of the key features of the ERC Didone project, one of its principal tasks being to create a digitally encoded corpus of some 3,000 arias in MusicXML format from about 180 musical settings of a small number of opera librettos by Pietro Metastasio. It focuses on some of the project’s research tasks, emphasizing how the skills of a team of eighteen scholars with very different expertise – historical musicology, music theory and analysis, cultural history, librettology, archival research, music performance, music engraving, MIR, computer science, and statistical modeling – combine to explore the potential answer(s) to the main research question of the project: How are emotions expressed through music?

Introduction

In a seminal article published in 1992, Don Randel presented the principles of what he ironically called the “Musicological Toolbox”, the research tools and conventions that “ensure familiarity and consistency” in our field and maintain the purity of the discipline to avoid any deviation from the academic canon. Like a computer’s interface, the primary function of the Musicological Toolbox is to “help reduce the time and effort required to produce the scholarly product” as well as to preserve the inherited rules, the look and feel, and the focus of our academic production [22, p. 10–13]. In the same volume, Robert Morgan observed that the history of Western music theory and analysis is deeply rooted in the conviction that “music constitutes a well-formed and coherent ‘language’ ... that there is a right way of understanding and realizing musical relationships, and correspondingly a wrong way as well”. Underlying music theory is “the belief in an unchanging eternal order surpassing time-bound stylistic transformations”, a conviction that is essentially at odds with any historical perspective of music [16, p. 44–45].

The rise of different approaches to the study of music, either ethnomusicology, critical musicology, or computational musicology, was more the result of a change in methodology than a change in the object of inquiry;

1  A recorded version of this keynote is available online: https://music-encoding.org/conference/2021/keynotes/. In this published version of the keynote presented on July 20 at MEC2021, we use the first person singular to keep the format of the conference, in which Álvaro Torrente presented this co-authored work.
in other words, it was a change of toolbox, a transformation of the look and feel. The maturity of any branch of knowledge is confirmed precisely by the establishment of its own toolbox, involving its own societies, conferences, journals, jargons, degrees, prejudices, ... and jobs. To a large extent, disciplines are a matter of power, a means to administrate the limited resources available in academia, as already observed by Bourdieu \[2\].

Despite this critical perspective, it is only fair to acknowledge that new methodologies normally result in the enrichment of knowledge, and this should be our main goal as academics. But it is also true that, when working in isolation within the scope of our restricted academic environment, the fruit of our investigation has obvious limitations. The rise of interdisciplinarity in the late twentieth century derived from the awareness of the constraints of isolated knowledge \[25\]. If separate disciplines must join forces to grow, this is even more true within the disciplines themselves, the branches of which, at least in theory, investigate the same object. A good and positive example in our field has been the interaction between historical musicology and ethnomusicology. The rise of the cultural history of music in the last few decades was a consequence of the adoption of the methods of ethnomusicology to study the music of the past. Quoting Randel again, “musicology and ethnomusicology begin to look a great deal more alike when we recognize that there is not such a thing as a work without a context” \[22, p. 11\]. Sometimes, the mimicry in historical musicology went so far as to study the context while entirely overlooking the musical work.

One of the reasons for the relative isolation of the various branches of musicology is that, to a great extent, music research has been an individual enterprise. Most articles and monographs, and even most conference papers, are commonly signed by a single author, or two at the most. This is certainly a consequence of the discipline’s tradition – the Musicological Toolbox – but also derives from the methods for evaluating academic careers. At Anglo-American universities, career promotion is always dependent on the individual publication of journal articles and at least one monograph. In Spain, the evaluation of a publication signed by two or more authors is very often rated much lower than individual contributions; in other words, collaboration is penalized.

A collateral consequence of such individualized research is that each publication is normally confined to the specific branch of its author’s expertise. I am primarily a historical musicologist, and most of my publications essentially belong to that field. It is true that I frequently apply methodologies from other disciplines and branches in individual publications, such as philology, history, music analysis, or ethnomusicology, yet my own speciality usually prevails, and it is not unlikely that scholars from those other fields might find my use of their methods somehow naive, if not entirely flawed.\[3\]

As we all know, this is not the case in other academic disciplines, particularly in STEM (science, technology, engineering, and mathematics), where the rule is to acknowledge the contribution of several authors, usually not fewer than three and very often more than a dozen. Obviously, this is the consequence of different academic conventions and mostly derives from the predominance of teamwork in STEM research. For example, the “Recommendations for the Conduct, Reporting, Editing, and Publication of Scholarly Work in Medical Journals”\[4\] state that authorship of scientific papers should be based on four criteria:

1. substantial contributions to the conception or design of the work; or the acquisition, analysis, or interpretation of data for the work;
2. drafting the work or revising it critically for important intellectual content;
3. final approval of the version to be published; and
4. agreement to be accountable for all aspects of the work in ensuring that questions related to the accuracy or integrity of any part of the work are appropriately investigated and resolved.

If we were to apply these criteria to musicology, publications deriving from master’s and doctoral theses should always include the signature of the supervisor, as it happens in many other fields. However, no academic musicologist would dare to do so, under the risk of being accused of appropriating their student’s work. Also, if we

\[2\] A very good example that anyone will easily grasp is the spectacular increase in the use of LaTeX for writing academic papers, as it is becoming compulsory in many fields and areas of research: If you do not write your paper in LaTeX, your work will not look or feel adequate.

\[3\] For example, one of my recent publications focuses on the study and critical editions of two poems dedicated to the zarábanda, yet I do not discuss the zarábanda itself; the result is a combination of historical studies and philology, yet it was published in a musicological journal; see \[26\].

were to apply medical conventions, the contribution of music engravers, data collectors, database compilers,
and research assistants should also be recognized in some way. Yet, in our field, all these contributors are
usually accommodated in the acknowledgment page only, if at all.

Beyond traditions, conventions, and concepts of authorship, the prevalence of individual, isolated research
in certain disciplines is also an aftermath of funding schemes. In STEM fields, research structures and funding
resources favor collaboration, as scholars usually work in teams associated with laboratories where every
member is paid to contribute their knowledge to the common goal. This working structure is thus reflected
in their academic output. This is rarely the case in the Humanities, where collaboration, if any, is the result of indi-
vidual eagerness and enthusiasm rather than of the existence of any professional structure. To put it bluntly,
scholars in the Humanities collaborate in their spare time after having completed all their compulsory duties.

1 The Musicology Lab

Major funding schemes, such as the European Research Council (ERC) grants, are fostering a dramatic shift in
the research procedures and structures of many disciplines. Almost for the first time in history, musicologists
and other scholars are having the opportunity to build research teams in which most members receive their
salaries to work exclusively on the given project; they are not paid to teach, nor to earn a degree, or to pursue
their own research interests, but to best devote their time and energy to the main goal defined in the project.
In some ways, thanks to these large schemes, musicology is starting to move from the individual desk to the
lab, from the Lone Ranger to the squad, from the prima donna to the choir, from the ivory tower to the inter-
action of specialists of several branches and disciplines.

The possibility of building a musicology lab has one fundamental benefit: It allows us to think differently,
in new dimensions; it allows us to imagine how to apply methodologies and techniques that are beyond the
capacity of the individual scholar. In short, building a Musicology Lab ultimately represents a change of para-
digm, a new toolbox, a window of opportunity to expand our methodologies and, more important, to imple-
ment true interdisciplinarity. Yet, it also has consequences that can hardly be truly evaluated when writing
your application to the scheme. Teamwork requires defining procedures and protocols for both research and
publications, to ascribe tasks to every team member, to establish milestones, to coordinate and supervise
achievements, and to confront problems. Team members need to realize that, although they have been cho-


2 The Didone Project

Applying to a funding scheme of the magnitude of the ERC Advanced Grants is like writing a letter to Santa
Claus: You are invited to imagine, to request everything you have been longing for, to assemble a true Dream
Team to work with you. In academia, this starts with the declaration of the primary research question. In my
case, it all started with a question that had been in the back of my mind since I began to study music some
forty years ago: *How are emotions expressed through music?* Please note that this question does not refer to the
psychological dimension of music, the aesthetic level – how music might move emotions in us –, but to the
poietic and neutral levels. What I was looking for is to understand the resources used by composers to create
works aiming to be expressive of specific emotions. Whether these compositions would eventually arise those
emotions in the listener would be a completely different matter. In other words, I wanted (and still want) to
know what makes music so powerful to “affect the minds of all, to get inscribed in the soul of man, to be fixed
in our memory in such a way that we do not have to think, and we are driven like waking up from a dream™, in the words of the medieval theorist Glarean.

I am aware that this question may not sound very original, as emotions have become one of the hot topics of our times. From politics to entertainment, from advertising to music, from television to computer science, we have suddenly discovered the importance of emotions in our lives, how emotions shape and drive essential decisions, how the gurus pulling the strings of modern society influence our behavior to buy certain goods, vote for specific candidates, or plan our holidays by moving the emotions of our souls. The novelty of the Didone project lies not so much in its research question itself, but in the materials used and the methodology chosen to find answers to that question, as I shall explain below.

The connection between music and emotions has become an important academic topic in the last two decades, with hundreds of related publications. However, this relationship is as old as musical thought itself. Indeed, the belief in the capacity of music to move human emotions can be traced back to Ancient Greece. The same conviction, based on the research of the Italian erudite Girolamo Mei – whose name, by the way, does not seem to be related to the Music Encoding Initiative (MEI)™ –, was the trigger that moved the members of the Florentine cameratas to invent, in the late sixteenth century, the form of drama with music that we now call opera [18]. A few years later, the father of modern philosophy, René Descartes, opened his first treatise, written in 1618, with the following statement: “The aim [of music] is to delight and to move various emotions in us”™. He was neither the first, nor the last.

Today, there is a scholarly consensus that emotions play a central role in Western music, and particularly in Italian opera. As expressed by the musicologist Carl Dahlhaus,

> If we regard the affects [emphasis added], the emotions, and the emotional conflicts expressed musically onstage in the form of arias, duets, and ensembles as the “true” musical drama [emphasis added], dramaturgical analysis of an opera should not start with the way a narratable action is reflected in music. Rather, quite the reverse, it should try to show how an action constituted as a drama of affects, primarily by musical means, comes to be based on a story line in order to take shape on a stage [3, p. 73–74].

At this point the question arises: If emotions are one of the main features of the most distinctive musical unit in opera, the aria, if emotions are “the true musical drama”, shouldn’t we be able to study, analyze, categorize, and tabulate them in a systematic manner, as we can do with key, harmony, musical form, verse structure, and so many seemingly “objective” elements in musical structure? How can we discuss the meaning and function of the operatic aria, how can we discuss musical drama at all without taking into account its emotional semantics? Indeed, understanding the emotions expressed in arias should be a precondition to study opera.

## 3 Mapping Emotions in Eighteenth-Century Italian Opera

This is precisely the point of departure of the Didone project: to study how emotions are expressed in opera. Walking this path requires several steps:

1. to identify and select a corpus of music that is expressive of emotions;
2. to define a conceptual and taxonomical framework for human emotions;
3. to develop a methodology to classify musical works by the emotion expressed;
4. to establish a procedure to compile musical data, which involves:
   a. collation and pre-processing of sources,
   b. encoding,
   c. manual annotation of some features;
5. to build a database to connect source and historical data with analytical evidence;

---

5 “Qui omnium mentes afficiat, qui hominis animo insideat, qui denique ita haereat memoriae nostrae, vt saepe, ne cogitantibus quidem nobis subrepat, in quem perinde atque e somno experrecti prorumpamus” [7, lib. II, ch. Xxxxviii, p. 174].
6 For a good synthesis, I recommend the recent book by Patrick N. Juslin [12].
8 “Finis ut delectet, variosque in nobis moveat affectus” [5, p. 5].
6. to define a large-scale method of analysis;
7. to develop code to automatically extract data from the encoded musical corpus;
8. to interpret the results.

3.1 The Selection of a Musical Corpus Expressive of Emotions

One of the main characteristics of the Didone project is that we do not use existing audio recording or symbolic notation corpora. On the contrary, we have identified a historical body of music that we believe to be expressive of emotions, a collection that is only available in manuscripts disseminated in dozens of libraries and archives around the world: Metastasian *opera seria*.

Our point of departure is the well-established assumption that, in eighteenth-century Italian *opera seria*, “arias were thought capable of suggesting one or two emotions chosen from a wide range of subtly varied passions” [9, p. 385]. Opera structure and aria form were relatively stable during most of the eighteenth century: Operas consisted of an alternation of recitatives and arias, while choruses, duets, and ensembles were seldom used. The action took place during the recitatives, while arias were closed, self-contained, and independent musical units that suspend the dramatic time to express, in the form of a monologue, the emotions experienced by the character on the stage. Most aria poems have two stanzas, each one set to different music and the first part usually repeated after the second, either in *da capo* or *dal segno* form, or, as the century progressed, in other ternary schemata. In short, most arias have a tripartite form where the third section is a repetition or variant of the first one: ABA'. The consistency of form and function of operatic arias during this period makes it possible to compare them in order to identify coincidences and differences, and eventually to trace features that may be related to the expression of emotions.

The second important singularity that makes *opera seria* particularly suitable for our goal is the preeminence of the leading librettist of the period: Pietro Metastasio. His 26 librettos were set to music nearly one thousand times by more than three hundred composers over the course of more than a century. Metastasio’s influence was so overwhelming that, during the 1750s, settings of his librettos represented over 30% of all the operas performed in Europe. Many of his *drammi per musica* were set to music over 50 times, and one of them, *Artaserse*, nearly reached 100 different musical settings. In the Didone project, we have chosen his five most popular librettos, for which 30 or more versions survived: *Didone abbandonata* (1724), *Alessandro nell’Indie* (1730), *Artaserse* (1730), *Adriano in Siria* (1732), and *Demofonte* (1733). It is true that most librettos underwent changes in each production, changes that implied the suppression of some arias and the replacement of others. In most later versions, only a handful of arias from the original libretto survived, while there are dozens of substitute arias that would need a case-by-case investigation.9

Even considering these circumstances, our research confirmed that dozens of settings of a high number of arias from the same Metastasian librettos have survived in manuscript form. The advantage of this corpus is that we can easily compare the musical settings of different composers for the same aria poems, written for the same character at the same point in the plot, and, thus, presumably aiming to express the same emotion.

3.2 The Definition of a Conceptual and Taxonomical Framework for Human Emotions

One of the major challenges of the Didone project was to adopt a conceptual framework for human emotions. No academic consensus has been reached, even within the realms of psychology, physiology, or neuroscience, on the very concept of emotion, nor on its categories, dynamics, or signification [11]. For example, the psychologist Robert Plutchik writes that “almost everyone agrees that the study of emotion is one of the most confused (and still open) chapters in the history of psychology” [21, p. 344]. The obvious reason is that emotions are not clear and distinct objects that can be identified, measured, described, and compared. Emotions cannot even be observed; we know of their existence indirectly, through their reflection on the body – heartbeat, face color, breath, muscular tension, face gesture, or neural stimulation –, through individual behavior, or through verbal description. Any attempt to describe emotions is biased by their subjective character and by the absence of a shared and accepted scientific language.

---

9 The works of Giuseppe Sarti have been explored in detail regarding this issue: https://sarti-edition.de (accessed August 12, 2021); see also [20].
For this reason, and after a thorough revision of historical and modern theories of human emotions, we decided to apply a cognate theory of human emotions, this is to say, a theory "which shares its fundamental basis, assumptions, and vocabulary with the theory of the period ... but which answers to modern requirements for degrees of thoroughness, consistency, and precision that are generally not found in the theory of the period when viewed from a modern perspective" [10, p. 112]. In other words, we have adopted the taxonomy and conceptual framework of emotions familiar to eighteenth-century individuals involved in opera production, as defined by René Descartes in *Les Passions de l'âme* (1649) [27]. This treatise is the first systematic attempt to explain the physiological dynamics of human emotions from a scientific point of view, as well as the most comprehensive and organized classification of human passions in that period. More importantly, *Les Passions de l'âme* was the reference work on this matter during the eighteenth century, including contemporary music theorists such as Johann Mattheson, and the conceptualization was reportedly applied by Metastasio in the creation of his dramas.

Descartes' taxonomy identifies six basic and 43 further passions that are either derived from or combinations of the basic ones. Descartes also identifies further features for each emotion: 1) the basic passion from which it derives; 2) its positive, negative, or neutral valence; and 3) its temporary reference: past, present, or future.

### 3.3 A Methodology to Classify Musical Works by the Emotion Expressed

However, the Didone project is not about philosophy, but about music, and identifying the emotion expressed by a piece of music is an even more subjective task than interpreting Descartes' theory. Most experiments to classify music emotionally assess judgements expressed by groups of listeners [6]. This is a standard procedure in psychology that can attain notable results regarding how emotions are perceived by listeners, but has two major drawbacks for our project: First, as I explained earlier, the aim of the Didone project is to explore *not the emotion perceived but the emotion expressed* in a work of music; and second, the emotional perception of modern listeners, with or without musical training, can hardly be equated to that of an eighteenth-century audience because their personal and musical backgrounds and experiences are radically diverse.

For these reasons, we needed to pursue a different path. The advantage of operatic arias is that they are not abstract pieces of music – as it happens with instrumental works –, but do have a poetic text sung by well-defined characters whose personal circumstances are clearly explained in that they occur within specific points in the development of the plot. All the elements of the drama, from the *characters* and the *action* to the *words* they use, are like a magic torch that illuminates the emotional semantics of each aria.

In this context, our methodology to label the emotions of each aria basically consists in the close reading of the librettos under the light of Descartes' taxonomy, in order to identify one or several passions that are suggested by the poet at a specific point of the plot. To reach this goal, we needed to examine six main features:

1. the constellation of characters and their specific circumstances: age, gender, social status, and dramatic function;
2. the relationship between characters, particularly kinship and emotional ties;
3. the implications of the eventual concealment of identity of one or more characters;
4. the succession of actions in the plot that provoke changes in the emotional status of one or more characters;
5. the addressee of the aria;
6. the actual words of both the aria poem and the recitative preceding it.

The combined analysis of these six features for each of the arias in the librettos allows their systematic classification. In this manner, each aria receives several labels, namely for:

- the emotion or emotions expressed;
- the valence (positive, negative, or neutral);
- the basic passion associated;
- the temporary reference (past, present, or future).

---

10 "The doctrine of temperament and emotions, concerning which especially Descartes is to be read because he has done much in music, serves very well here, since it teaches one to distinguish well between the feelings of the listener and how the forces of sound affect them" [15, part I, ch. 3, art. 51, p. 130–131].
Limiting our scope to just the five librettos by Metastasio mentioned above makes the process much simpler compared to looking at a total of (about) 3000 arias from (some) 180 scores: This way, only 182 original aria texts had to be classified according to the procedure just explained. In those librettos, we have identified a total of 25 different emotions out of the 49 categories from Descartes’ taxonomy. Here I would like to illustrate the process with just two examples:

**Son regina e sono amante**

At the beginning of the opera *Didone abbandonata* (Act I, Scene 5), Iarba, King of the Moors, disguised as his own ambassador, visits Didone, the Queen of Cartago, in her own palace. There he claims that she must marry him, otherwise she would become his enemy and be destroyed. Didone’s reply in the aria “Son regina e sono amante” is a straightforward, proto-feminist statement, declaring herself to be a queen and a lover, and that no one will rule her throne or her bed. We could spend hours discussing the nuances of the emotions expressed in this aria, yet most people would agree that it is a statement of pride, as Didone expresses her female autonomy from male control, as well as of boldness, since she is bravely confronting her eventual (male) antagonist and slayer.

**Misero pargoletto**

In the opera *Demofoonte*, Timante, believed to be the son of the ruling king, is secretly married to Dircea, thought to be the daughter of the prime minister Matusio. Their offspring is the little boy Olinto. At the beginning of the third act, Timante finds out that Dircea is in truth Demofoonte’s daughter, implying that he would have unconsciously committed the most terrible of all sins, incest, as he married his own sister and be the father of his own nephew. After a moment of despair (Act III, Scene 5), Dircea, who is unaware of this information, brings the little boy in front of the father, thus inspiring the aria “Misero pargoletto” sung by Timante to his son. The confrontation with his abhorrent crime is portrayed in an aria expressive of both terror for the horrible consequences of his acts, as well as pity towards a boy that is the fruit of unconscious evil [28]. Shortly after this aria, they find out that Timante is in fact the son of Matusio and not of Demofoonte, and therefore has not committed incest, bringing the opportunity for the lieto fine.

As illustrated by these two examples, our emotional labeling process is built on a careful analysis of the libretto which considers all the factors involved in the drama.

### 3.4 The Compilation of the Data: Towards Encoding and Annotation

The step with the biggest workload in our project is the compilation of the socio-historical data and the creation of the corpus. Most team members participate in this task, and many devote all their time to it. Among them, there are four music engravers working full-time (Carlota González, Alberto Cubero, Iván González, and David Almazán), three part-time practical musicians involved in the correction of the copies (Javier Ulises Illán, José Antonio Montaño, and Teresa Casanova), one research assistant (Gorka Rubiales), and three researchers: the musicologist Tatiana Aráez, the librettologist Valentina Anzani, and the technical supervisor and co-author of this paper, Ana Llorens, who organizes and supervises the whole procedure. Most of these team members have both music and musicology degrees, including several instrumentalists and three conductors specialized in early music: Cubero, Illán, and Montaño. The process involves several phases:

#### a Sources

The first phase consists in compiling a complete inventory of the sources of all the musical settings for each of the five librettos under investigation. This is not the place to explain all the requirements to locate, identify, and retrieve the poetic and musical sources of some 180 operas. Despite the substantial information available in reference works such as *The New Grove* [23], Sartori [24], RISM, [12] or Corago, [13] members of the Didone team have found many other sources not identified there [13].

---

11. For a detailed discussion of the application of the system to each of the librettos studied, I invite you to watch the seminars delivered by José María Domínguez and myself available on the Didone project’s website: [https://didone.eu/index.php/seminars/](https://didone.eu/index.php/seminars/) (accessed August 12, 2021).


The next step is the provision of digital copies of the sources identified as most relevant to our project. This task was initially approached in partnership with several European libraries, such as the library of the Conservatoire royal de Bruxelles, the Österreichische Nationalbibliothek (Vienna), the Bibliothèque Nationale de France (Paris), the Conservatorio S. Pietro a Majella (Naples), or the Biblioteca da Ajuda (Lisbon), yet it was severely affected by the COVID-19 lockdown. To overcome this difficulty, two of our researchers (Anzani and Aráez) are currently visiting the libraries in person to make digital photographs of the musical manuscripts and poetic imprints.

b Encoding

The process of engraving and producing encoded versions of the arias faces all the challenges of making a music edition plus the additional demands of computational analysis.

The first issue in our case was to define the editorial criteria that would work for a repertoire covering some six decades and written by dozens of composers – further copied by several different hands – who used slightly different notational conventions. It soon became clear that with such a massive corpus it was not practical to follow the conventions of critical editing. Our aim is not to reconstruct the ideal version of each aria as conceived by its composer – if that were possible in this repertoire – but to obtain a clean digital copy from the most reliable source, usually the one most likely connected with the premiere of the opera. This required as a preliminary step the identification of the codex optimus for each opera, ideally the composer's autograph if preserved.

However, some adjustments were necessary to ensure consistency among our engraved arias, adjustments that deviate our copies from the conventions of critical, performative, or diplomatic editions. The purpose was to minimize false computational variants and to standardize the corpus. One important example is the notation of the voltas in da capo and dal segno arias, another the normalization of dynamics and articulations, and a third the interpretation of time signatures to convey a sensible hypermetric structure.

The Instituto Complutense de Ciencias Musicales (ICCMU), the research center at the Universidad Complutense de Madrid hosting the Didone project, has three decades of experience in publishing music editions of large scores, mostly opera and zarzuela, as well as symphonic and chamber music. Since all the ICCMU editions have been engraved using Finale music software since version 1.0, we took advantage of our team's expertise and decided to use that software from the beginning. Finale allows us to automatically export the corpus as MusicXML files, but it is also important that in the future our editions can be used for both digital and paper publication, as well as to extract parts for their eventual performance by orchestras and singers.

During the early phases of the project, we explored the potential advantages of other encoding formats, particularly MEI and **kern. We identified several differences and a few minor challenges in the transfer between the diverse formats, and in the end, we were able to confirm that MusicXML was the most satisfactory format for our needs [19]. This is probably due to the fact that our corpus completely responds to common notational practice and has very few unconventional requirements, as would be the case with medieval or Renaissance music.

Beyond the basic export from Finale, MusicXML has two additional advantages. On the one hand, MusicXML files can be opened with common-use music software such as Musescore, which we use for harmonic annotation. On the other hand, we can benefit from the power of the music21 library to extract meaningful information from our aria files in order to apply computational analysis.

c Annotation

At the start of the project, we were not aware of any effective solution that can automatically extract harmonic information from encoded symbolic music such as our corpus. We tried some applications, but they were not reliable enough, especially at junctures with double functionality or omitted roots in the chords. Since we are convinced that harmony is a major feature for music semantics, and specifically for the expression of emo-

---

17 Our former team member Paula Muñoz Lago even contributed an improvement in version 6.5 of music21 related to lyrics processing; see https://music21-mit.blogspot.com/2020/12/music21-v65-released.html (accessed August 12, 2021).
tions, we faced the necessity of finding or developing a system to annotate and analyze the harmonic structure of our arias.

On this matter, we contacted members of the Digital and Cognitive Musicology Lab (DCML), directed by Martin Rohrmeier, at the École Polytechnique Fédérale de Lausanne. They had already been working on a harmonic annotation standard syntax [17], together with Python tools – including the library ms3 [8] – for the extraction of such annotations. Since MusicXML files vary depending on the engraving software, the DCML uses MuseScore 3 to introduce and store the annotations to be extracted. Ana Llorens has been working with the DCML team for the last eighteen months to improve the standard, as well as to initiate some other shared projects. Apart from the obvious benefit to our analysis, the manual annotation of the harmony of several thousands of arias will result in a corpus of encoded music that could be used in the future for machine learning and the eventual development of more precise automatic systems of harmonic annotation.

3.5 The Database

To determine when and where the operas were premiered, or who performed them for the first time, our team of historical musicologists have had to design and build a large relational database to store all the potentially relevant data for our analytical enterprise and all the connections therein. While our guiding unit is the opera production – with its venue, date, occasion, composer, singers, librettist, etc. –, we enrich our data in a process of specification to the level of the aria, providing information on the singer’s voice type and compass, key, time signature, source, variants, dramatic position, passion, etc. This approach is different from that of other databases, such as Corago, which we took as an initial model. In this way, we can combine the musical and textual information extracted computationally from the MusicXML files with the contextual and historical data in which that information emerged.

3.6 A Large-Scale Method of Analysis

The ultimate purpose of the Diédone project is to identify musical features associated with the expression of emotions. To that aim, we aspire to design a model of analysis that can be applied to thousands of arias. The massive number of works to be analyzed necessarily excludes the inspection of individual details and puts the focus on parameters that can be compared on the large scale. Our music theorist Ana Llorens has defined eleven parameters by which data can be extracted from the encoded corpus, either automatically or from the manual annotations. Those are: 1) key, 2) scoring, 3) tempo, 4) melodic configuration, 5) harmony, 6) meter, 7) form, 8) rhythm, 9) texture/density (including syllabic density) and articulation, 10) dynamic structure, and 11) phrase structure.

For each of these parameters, specific metrics, or features – currently around 200 –, have been musically defined with a view to combine them in the analysis. Some of the features, such as those referring to key, harmony, or form, apply to the musical piece as a whole and thus can be compared without further standardization. Others, however, such as those related to melodic construction or texture/density, depend on the number of parts – or instruments – sounding in the aria. In fact, they even depend on the metric structure, the time signature of the piece, and the tempo if we consider the temporal dimension of music, this is to say, if we still want to treat music as annotated sound and not just as a symbolic entity.

That is the reason why we have defined various hierarchical levels from which to extract the data, namely the score level, the instrument level, and the part level. The score-level metrics can be extracted from all arias in the corpus, whereas the instrument- and part-level metrics vary depending on the scoring or instrumentation. Therefore, several metrics are extracted only from those parts/instruments that are present in the whole corpus, such as the strings and the vocal part, whereas we aim at reflecting the influence of other instruments in other metrics extracted at the score level.

19 By “instrument” we refer to as many equal instruments as there are in the piece. For instance, two oboes would form a single instrument.
3.7 The Code
At this stage, we needed to extract the relevant data automatically, not just for the arias one by one, but also for groupings thereof, in order to be able to discern whether, for instance, some historical or geographical factors played a role in the way our corpus actually sounded or the way we think it sounded. Therefore, we are developing two kinds of software in the Didone project.

First, we are designing a library, musiF, which serves to extract the desired metrics for each of the arias separately. Once the data are extracted, a large data frame with several thousand rows and columns containing the data for each of the arias is created, processed, and, ultimately, analyzed. The musical/poetic metrics are placed alongside the passion(s) expressed through the aria text, according to the method explained before, as well as other metadata that will help us to interpret the data in a more historically and aesthetically informed manner.

Second, we are creating code to generate data reports, in the form of Excel spreadsheets, with the selected metrics at the aria level, but also to group the arias by a variety of historical, geographical, dramatic, and even purely musical factors. This has two main advantages: First, these reports, along the accompanying visualizations, were designed by our music theorist and can easily be interpreted by her peers, thus fostering interdisciplinary collaboration; and second, thanks to human interpretation, they allow for discrimination of stylistic, historical, or other factors that might play a role in the ways composers expressed specific texts through music, beyond the influence of the emotion of the character on the stage.

3.8 The Interpretive Phase
We are continuously developing this phase, as all perspectives of the Didone project converge here. Currently, we are experimenting with various statistical and machine learning methods to analyze the data. As a first step, our computer scientists (Daniel Ibáñez and Martín Serrano) and our senior statistician (Eduardo García-Portugués) perform some exploratory analyses (regression models, correspondence methods, decision trees, etc.) on data from a small portion of our corpus, some 500 arias. They try to determine whether there is a specific musical metric or, most plausibly, a combination of metrics that can act as good predictors for the positive vs. negative valence of the emotion(s) expressed in the aria, yet this seems too general a classification. Therefore, they have already started to focus on the six basic passions as defined by Descartes and their musical expressions, with some promising preliminary results.

In such an analysis, the music is treated ahistorically, that is, our algorithms process the data regardless of their date or geographical provenance. Yet, thanks to the data reports I mentioned before, Valentina Anzani and Ana Llorens have identified some traits specific to composers’ vocal writing for the castrato Carestini in the 1740s [1], and Ana Llorens and myself have observed some metrical- and gender-related practices specific to the court Iberian productions of mid-eighteenth century [14].

Conclusion
I hope that this little promenade has illustrated the aims and procedures of the Didone project, and, more importantly, the need for interdisciplinary work in the Humanities. Choosing a complex cultural product as the main object of our investigation automatically implies going well beyond music. It requires an understanding of the poetic language, the dramaturgy, the social and political context, the contemporary codes and values, the production system, the technical resources, and the philosophical background of opera. In other words, we depart from a series of decisions grounded on historical evidence and influenced by cultural and philosophical considerations to search for answers through the application of music analysis, statistics, and algorithms to an encoded corpus of music.

The process has implied a transformation of our Musicological Toolbox. If the comprehensive study of opera necessarily calls for interdisciplinarity, massive analysis of operatic arias requires going one step further within the discipline of music itself, in an agreement between different branches of musical research. This could appear to be a simple task, but it has certainly required a substantial dose of patience, compromise, and empathy to understand and apply the procedures and requirements of colleagues from other branches. The best example is our system to engrave arias. We call it the Arias Factory, because it runs like an industrial
process involving a large part of the team. It is a process in which each member must perform specific tasks and register them in order to produce top-quality encoded versions.

We all have learned a great deal in the course: how to work together, how to fulfill our individual tasks, how the final results depend on the contributions of every member, how to leave aside our personal inclinations and needs in order to reach agreements and find solutions to each challenge. In short, we have started to define what we could call a Musicology Lab. I am using the plural because this procedure was not imposed top-down. Quite the opposite, it is the result of the input of all members, and their invaluable contributions have helped to define and polish measures and methods.

The transformation has taken place inside the group, and we have yet to expand its influence beyond our research environment. We need to demonstrate that three or more authors signing an article does not divide its value; quite the opposite, their contributions can multiply the quality of the work. We need to show that the work of a team, also in music research, can reach scientific summits that are unattainable to individual scholars. We also need to encourage financial support for new professional profiles in musicology beyond the traditional teacher-researcher. In short, we need to keep transforming the Musicological Toolbox.

As explained above, the main outcome of the Didone project will be to identify certain resources used by eighteenth-century operatic composers to express emotions. Beyond this, we foresee several collateral outputs that will hopefully benefit a wide musical and musicological community. One significant result will be the development of algorithms for corpus analysis of symbolic music that will be applicable to other corpora. Furthermore, a major fruit will be to make new editions of thousands of neglected arias accessible in digital format, which will expand the range of music available to both individual performers and operatic venues. These arias will also be distributed in MusicXML and (annotated) Musescore formats to allow other scholars to double test our analytical processes as well as to apply other methodologies to our corpus. We have already set the date to grant open access to this massive corpus: It will take place on 1 February 2024, exactly three centuries after the first *dramma per musica* by Pietro Metastasio, *Didone abandonata*, was premiered in Naples with music by Domenico Natale Sarro. This will be our humble tribute to one of the most important figures in the history of opera.

Acknowledgments

Writing this paper has been a transformative process in itself. I am very grateful to Stefan Münnich and David Rizo for their invitation to give a keynote at the most important conference on music technology, not only for the honor of addressing its illustrious audience and the potential readers of this written version, but also, and perhaps more importantly, because this paper has prompted me to think in depth about the kind of work we are doing in the Didone project. In fact, if anyone deserves acknowledgment, this is the members of the Didone team, a team that constitutes a true Musicology Lab.

This paper is a result of the Didone project, which has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme, Grant agreement No. 788986.

Ana Llorens' work has been carried out under a “Juan de la Cierva–Formación” contract, funded by Spain's Agencia Estatal de Investigación, Ministerio de Ciencia e Innovación, ref. FJC2018-035534-I.

Works Cited

Encoded Spanish Music Heritage through Verovio: The Online Platforms *Fondo de Música Tradicional IMF–CSIC* and *Books of Hispanic Polyphony IMF–CSIC*

Emilio Ros-Fábregas  
Institución Milá y Fontanals of Research in the Humanities-CSIC, Barcelona, Spain  
emros@imf.csic.es

**Abstract**

This paper presents the recent implementation of encoded music notation examples in two open access platforms devoted to traditional music and polyphony, respectively: *Fondo de Música Tradicional IMF–CSIC* (FMT)\(^1\) and *Books of Hispanic Polyphony IMF–CSIC* (BHP)\(^2\). Even though, at first, both repertories seem unconnected, they have many textual/literary and musical points in common – such as the presence/survival of old 15\(^{th}\)–17\(^{th}\) century texts and/or melodies of polyphonic *romances* (ballads) in the 20\(^{th}\) century oral tradition –; future technological developments will facilitate further connections beyond the ones already found through search by text incipits and numeric melodic incipit connecting both platforms. The presentation will have two parts devoted to *FMT* (with more than 20,000 images of melodies in open access, it is the largest online archive of folklore in the Hispanic world; it is used also for OMR research) and *BHP* (a reference online catalogue for polyphonic choir books in Spain and books with Hispanic polyphony elsewhere). After a brief explanation about the origin and scope of the repertories covered in each platform, a few examples of encoded transcriptions of a melody, of an incipit in mensural notation, and of a polyphonic work (in MusicXML, **mens, and **kern; MEI can be used, too) rendered through Verovio will illustrate the potential development of *FMT* and *BHP*. Both websites constitute leading educational and musicological resources of the very rich Spanish music heritage, inviting national and international collaboration (crowdsourcing) to expand the contents of both platforms and to develop their digital technology.

**Introduction**

This paper presents the recent implementation of encoded music notation examples in two open access platforms devoted to Spanish traditional music and polyphony, respectively: *Fondo de Música Tradicional IMF–CSIC* (FMT)\(^1\) and *Books of Hispanic Polyphony IMF–CSIC* (BHP)\(^2\). The presentation will have two parts devoted to *FMT* and *BHP* to explain briefly their origins and scope of the repertories covered in each platform; musical examples of a melody, of an incipit in mensural notation, and of a polyphonic work will illustrate their rendering through Verovio.

\(^1\) [https://musicatradicional.eu](https://musicatradicional.eu) (accessed January 12, 2022).


© Published in 2022 with Humanities Commons under a CC BY-NC-ND 4.0 ([https://creativecommons.org/licenses/by-nc-nd/4.0/](https://creativecommons.org/licenses/by-nc-nd/4.0/)) license.
 Origins of the Fondo de Música Tradicional IMF-CSIC (FMT) and Encoded Transcription of Spanish Folk Tunes

In 1943, the Spanish musicologist Higini Anglès (1888–1969) founded the former “Instituto Español de Musicología” of the CSIC (Consejo Superior de Investigaciones Científicas / Spanish National Research Council) in Barcelona, one of the main projects of which was to collect pieces of oral tradition throughout Spain. For these so-called “misiónes folclóricas” (1944–1960), Anglès engaged the most prestigious folklorists of the country and invited Marius Schneider (at that time head of the Phonogramm-Archiv in Berlin) to direct the Folklore department. As a result of this project, more than 25,000 melodies were collected on paper from close to 3,000 locations, as well as cards with basic data about the informants who sang/played these pieces. Although between 1951 and 1987 the CSIC published five volumes with transcriptions of pieces collected through these “misiónes” (and competitions, “Concursos”, to collect repertory), most of the materials remained unpublished and basically forgotten for several decades at my institution, the Institución Milá y Fontanals of Research in Humanities (IMF-CSIC), heir to the former Instituto Español de Musicología.

In 2010, I started to organize and catalogue all these materials (with the help of my colleague at the IMF-CSIC, María Gembero-Ustárroz, and former PhD student Ascensión Mazuela-Anguita, now teaching at the University of Granada); in September 2012, Jan Koláček helped us to create the online platform Fondo de Música Tradicional IMF-CSIC, presenting it to the public on 4 February 2013 with the first thousand items. It should be pointed out that those materials had no name, and thus its current name Fondo de Música Tradicional is new, referring both to an old ‘physical archive’ at the IMF-CSIC in Barcelona and to an ‘online platform’ to present the materials. At that time, and with no financing, the priority was to digitize thousands of cards of the melodies and of the informants’ data to present the information online in a structured database (MySQL in DRUPAL) which could be useful both to researchers and the general public. From the very beginning, we wanted to incorporate music incipits, and we adopted a useful numeric incipit for practical reasons, since a complete music transcription and encoding of the melodies was out of the question at that time due to our limited financial and technological resources. Since then, the platform has incorporated digitized IMF-CSIC audios from cylinders, magnetic wire, magnetic tape from the “misiónes” as well as other related materials from other sources. By 2016, an international review stated: FMT “[…] presents in digital form the single most important archive of Spanish musical folklore” [3, p. 869].

Now that FMT has included more than 46,000 pieces (from FMT and other sources) and is getting closer to presenting most of the melodies held at the IMF-CSIC, we can look forward to incorporating the full text of the songs, complete transcriptions of the melodies, and their encodings. In recent years, we have experimented with exporting transcriptions from Sibelius or Musescore to MusicXML, which is rendered in FMT through Verovio, with SVG being the front-end notation graphics seen on the website (see Figure 1). This result is the fruit of collaborative work among members of our project: Antonio Pardo Cayuela (Universidad de Murcia) and webmaster Jan Koláček, based on the initial modern transcription by Esperanza Clares (Universidad de Murcia). We are engaging undergraduate and graduate students to transcribe melodies of the FMT, export them to MusicXML (MEI or **kern), and incorporate them into the website as part of academic internship opportunities at different universities; we hope that this kind of crowdsourcing will be very productive in many respects. FMT currently has over 800 modern transcriptions that can be searched under the menu “Pieces”.

---

3 For a full discussion of FMT and BHP, see [4].

4 The numeric music incipit we created for FMT represents the intervals between the first few notes of a piece, indicating whether the interval is ascending (+), descending (-), or represents a repeated note (=0); the number corresponds to the number of semitones (C-D is +2). For instance, the music incipit F-A-C-D-F-D can be searched through its equivalent numeric incipit: +4+3+2+3-3. This simple system helps to identify identical melodies transposed to different tonalities and/or with different texts, as well as to distinguish between different musical versions of pieces with identical text; for instance, this incipit +4+3+2+3-3 helped identify two instances of the same melody, one without text starting on F and the other with text starting on D.

Figure 1: FMT presentation of the lullaby “Mi nenica es mi nenica”, collected in 1947 in Lorca (Murcia), with images of the original cards of the transcription of the music and text; it also includes the modern transcription (a MusicXML file rendered through Verovio). See Fondo de Música Tradicional IMF-CSIC: https://musicatradicional.eu/piece/12280 (accessed January 12, 2022).
We are also collaborating with the R+D project “Handwritten Spanish music heritage preservation by automatic transcription (TIN2017-86576-r)” carried out by Jorge Calvo-Zaragoza, David Rizo Valero, and José Manuel Iñesta Quereda in order to apply the Optical Music Recognition (OMR) application MuRET, developed by them at the Universidad de Alicante, Spain, to FMT; this research will certainly speed up the transcription/encoding process [1]. We are planning to incorporate user-friendly one-click melodic analysis and other music searching tools into FMT.

FMT is still a modest project from the music encoding point of view – especially when compared, for instance, to such formidable repositories as the Dutch Song Database6 –, but it constitutes a very important and well-documented collection of Spanish music heritage with ramifications in Latin America. Moreover, the future incorporation of the FMT as a dataset for Music Information Retrieval (MIR) research could open new vistas. As pointed out by Peter van Kranenburg and Berit Janssen, the same dataset of 20,000 songs, the Essen Associative Code and Folksong Database (EsAC)7 is used over and over (de-contextualized) “to test segmentation algorithms, melodic similarity measures, pattern discovery algorithms, and the like”, and perhaps FMT could provide another alternative.

2 Origins of *Books of Hispanic Polyphony IMF-CSIC (BHP)* and Encoded Transcription of Spanish Polyphony

Soon after the creation of the *Fondo de Música Tradicional IMF-CSIC* – designed for uncatalogued holdings in my institution (literally in the office next door) –, I realized that a similar platform would be very useful for my own research on Spanish polyphony and of interest to the community of scholars. Thus, I started *Books of Hispanic Polyphony IMF-CSIC (BHP).* Why was it necessary? At that time, in 2013, and still today, we could say (although we have now a better idea thanks to BHP):

- We don’t know how many books of polyphony there are in Spain.
- We don’t know how many books outside Spain contain Hispanic polyphony.
- Catalogues and other existing reference works are very useful, but
  - they may refer to a single institution only,
  - they may be cataloguing only manuscripts,
  - they may be cataloguing only printed books,
  - they may cover only a restricted chronological period (15th–16th century),
  - they may cover only a particular genre.
- *RISM (Répertoire International des Sources Musicales)*, with over 1.3 million records, is “the largest and only global organization that documents written musical sources”8. RISM attempts to catalogue all music to 1900 (including all Spanish/Hispanic sources), but that’s quite a lot to cover, and the Hispanic world is not well represented. *BHP* can play a collaborative role.
- All of these limitations directly affect our view of Hispanic polyphony, and in particular hinder a broader, *longue durée* perspective beyond the 16th century.
- The digital age offers new research tools.

After four years of preparatory work, our team (see *BHP* homepage) made the website available to the public in open access coinciding with our presentation during the Medieval and Renaissance Music Conference in Prague on 7 July 2017.

---

8 “An illustration of this is the way in which the EsAC collection is used in MIR research. Virtually all papers in the proceedings of the yearly conference on Music Information Retrieval (ISMIR), in which this set of melodies are used, do not show an interest in folk music as such. Instead, the melodies are taken as just a collection of labeled musical data to test segmentation algorithms, melodic similarity measures, pattern discovery algorithms, and the like. The meta-data that comes with the collections (e.g., region of origin, tune family membership, segment boundaries), are used as ground-truth data for corresponding MIR tasks” [5, p. 118].
The objective of *Books of Hispanic Polyphony* (*BHP*) is to serve as a comprehensive research tool concerning manuscript and printed polyphonic books in Spain as well as books with Hispanic polyphony elsewhere. *BHP* has no chronological limitations, and we have started to cover the period from the 15th through the 20th century. We do not provide images of the sources, unless the institutions that hold them do so through repositories or permission. In such cases, we incorporate the appropriate links; for instance, we have found the following repositories particularly useful: (a) *Biblioteca Digital Hispánica*[^10] for books at the Biblioteca Nacional de España in Madrid; (b) *Digital Vatican Library*[^11] for choir books at the Biblioteca Apostolica Vaticana; and (c) the digital collections at the Bayerische Staatsbibliothek in Munich[^12], among others. We also incorporate links to manuscripts described in *Digital Image Archive of Medieval Music (DIAMM)*[^13] to printed books in *Printed Sacred Music Database 1500–1800*,[^14] to *Portuguese Early Music Database (PEM)*[^15], and to other appropriate websites that could offer relevant information. Our menu with “Sources”, “Locations”, “Institutions”, “People”, “Genres”, “Works”, “Movements”, “Documents”, and “Bibliography” should give you a sense of our objective, and we are still building searching tools and different ways to connect information. We would like to serve as a reference tool and as a platform to present original research and foster international collaboration. This digital platform was initiated as part of the four-year (2013–2016) R+D Project “Libros de polifonía hispana (1450–1650): catálogo sistemático y contexto histórico-cultural” (HAR2012-33604) of the Spanish Ministry of Economy and Competitiveness. *BHP* continued to be part of the objectives – together with *FMT* – of a new R+D project “Hispanic Polyphony and Music of Oral Tradition in the Age of Digital Humanities” (HAR2016-75371-P, Spanish Ministry of Science and Innovation, 2016–2020).

Currently, *BHP* contains information about 2478 polyphonic sources, 545 institutions, 1821 people (musicians and non-musicians related to books of polyphony), 5929 works, 6029 movements of works, 104 documents (such as old inventories), and 1770 bibliographic items; it has become an international reference work in the field. As we did with *FMT*, we included a search option by numeric music incipit as a helping aid to locate possible concordances among pieces for the identification of anonymous works; the only difference between *FMT* and *BHP* in this respect is the addition of =P to indicate the presence of a pause/rest of any length in the music incipit of a polyphonic voice in *BHP*[^16].

An important aspect of the project is the incorporation of music encodings using two Humdrum formats: **mens** for incipits in original, mensural notation, and **kern** for full transcriptions of polyphonic works in modern notation, both rendered through Verovio. This has been accomplished thanks to the collaboration with David Rizo Valero (Universidad de Alicante), Antonio Pardo Cayuela (Universidad de Murcia), and our webmaster Jan Koláček; in the near future, these and other formats will allow us to incorporate sophisticated analytical tools. See in Figure 2 the incipit in mensural notation of “Qui pius prudens” from the hymn “Iste confessor” by Francisco Guerrero (1528–1599), and in Figure 3 the beginning of the full transcription of this work in the webpage of *BHP*[^17].

[^16]: See the alphabetical list of piece titles with the corresponding numeric music incipit in the menu “Movements”: https://hispanicpolyphony.eu/movements (accessed January 12, 2022).
[^17]: For information and multiple resources about Humdrum, see CCARH Humdrum Portal (http://humdrum.ccarh.org/; accessed January 12, 2022).
**Figure 2:** Hymn "Iste confessor" by Francisco Guerrero (1528–1599). BHP webpage of the polyphonic verse "Qui pius, prudens" with the music incipit of the superius voice in the original, mensural notation rendered by Verovio from encoding in **mens. See Books of Hispanic Polyphony IMF-CSIC: https://hispanicpolyphony.eu/movement/26874 (accessed January 12, 2022).
<table>
<thead>
<tr>
<th>Source navigation:</th>
<th>Piece navigation:</th>
</tr>
</thead>
<tbody>
<tr>
<td>E-ORic 06</td>
<td>05. Iste Confessor</td>
</tr>
</tbody>
</table>

**E-ORic 06 (18v-23r)**

**Order No.:** 05  
**Source:** E-ORic 06  
**Title:** Iste Confessor  
**Text incipit:** Qui pius, prudens  
**Text underlay:** \[\texttt{Qui pius, prudens}\]  
**Ascription:** Amen.  
**Attribution:** Guerrero, Francisco  
**No. of voices:** 4  
**Scribes:** Sart, Josè  
**Language:** Latin  
**Genre:** Sacred vocal + Hymn  
**Date:** 18/4 (1777)  
**Location/Origins:** Comunidad Valenciana + Alicante + Orihuela  
**Comments:** Alternating and abbreviated version of Iste confessor included in Liber Vesperarum Francisco Guerrera Hispaniae Ecclesiae Magistro auctore (Roma: Domenico Bassa, 1584) (RSM G487). This setting uses the verses of Iste confessor Domini coelentes instead of Iste confessor Domini sacratus, used in Liber vesperarum. The music applied to verse 4 (Unde nunc noster chorus in honorem, canon in diapason) in Liber Vesperarum's version has been suppressed.

**Text:**

[***Iste confessor Domini coelentes variant***].

1. Iste confessor Domini coelentes/ Quem ple laudant populii per orbem/ Hac die letus meruit beatas/ Scandere sedes.  
   *(If it is not the day of his death, the last line is changed to)* Hac die letus meruit suprerns/ Laudis honores.  
2. Qui pius, prudens, humilis, pudicus/ Sediis duet sine labe vitam/ Donec humanos animavit aure/ Spiritus artus.  
4. Noster hinc illi, Chorus obsequentem/ Concit laudem, celebresque palmares/ Ut pils ejus precibus juventur/ Omne per avum.  
5. Sit salus illi decus atque virtus/ Qui super cali solio coruscans/ Toltis mundi seriem gubernat/ Trinius, & unus. Amen.

![Sheet music of Iste confessor](image)

**Figure 3:** Webpage of BHP with the hymn “Iste confessor” by Francisco Guerrero (1528–1599) in the choir book *Libro de Partitura de Abril* 6 at the Cathedral of Orihuela (Alicante); image includes the beginning of the complete four-voice polyphonic transcription, starting with “Qui pius, prudens” rendered by Verovio from **kern. See Books of Hispanic Polyphony IMF-CSIC: https://hispanicpolyphony.eu/piece/16370 (accessed January 12, 2022).
Since our webmaster, Jan Koláček, is also responsible for other related websites, we are exploring the simultaneous search in several databases. In the menus “Works” and “Movements” in BHP, a search by “Title” offers additional searches in two websites: 1) Portuguese Early Music Database (PEM); and 2) Fondo de Música Tradicional IMF-CSIC (FMT). For instance, a search by “Title” in BHP of the word “Domine” finds 278 pieces containing that word; at the end of the list, the option “Click here to search in external resources (PEM)” provides 84 additional findings in that database, and 4 other works after clicking on “Click here to search in Fondo de Música Tradicional IMF-CSIC”. The connection between BHP and FMT is particularly valuable to research possible relationships between melodies and texts from old romances and villancicos in Spanish (15th–17th century) and the repertory of oral tradition collected in the 20th century [2]. See Figure 4, where a search for “Triste de mi” finds the villancico “Triste qué será de mi” in the Cancionero Musical de Palacio (E-Mp II-1335; 1495–1520) and six pieces in FMT, probably not related in this case but containing the words “triste de mi”. We hope to apply simultaneous search of encoded music notation soon.

BHP has also adopted the list of liturgical feasts (“List of feasts”) from Cantus: A Database for Ecclesiastical Chant – Inventories of Chant Sources; we thank Debra Lacoste (University of Waterloo, Canada) for her permission to do so. This way, in the near future, a search for a particular feast in BHP will provide liturgically related polyphonic works and, with an additional click, plainchant melodies for the same celebration listed in Cantus.

![Figure 4](image-url)

**Figure 4:** A search in BHP by title “Triste de mi” finds one piece in the Cancionero Musical de Palacio (E-Mp-1335; 1495–1520) and, after a simultaneous search in FMT, six other pieces of oral tradition collected in the 20th century containing those words.

---

Conclusion

Altogether, both platforms, *Fondo de Musica Tradicional IMF-CSIC* and *Books of Hispanic Polyphony IMF-CSIC*, constitute leading educational and musicological research resources for the very rich Spanish music heritage, inviting national and international collaboration (crowdsourcing) to expand the contents of both platforms and to develop their digital technology related to encoded music notation and analytical tools.

Works Cited


Alleviating the Last Mile of Encoding: The mei-friend Package for the Atom Text Editor

MEC 2021 BEST PAPER AWARD

Werner Goebl       David M. Weigl
Department of Music Acoustics –     Department of Music Acoustics –
Wiener Klangstil (IWK), mdw –      Wiener Klangstil (IWK), mdw –
University of Music and Performing Arts     University of Music and Performing Arts
Vienna, Austria        Vienna, Austria
goebl@mdw.ac.at      weigl@mdw.ac.at

Abstract

Though MEI is widely used in music informatics and digital musicology research, the relative lack of authoring software and the specialised nature of its community have limited the availability of high-quality MEI encodings. Translating to MEI from other encoding formats, or generating MEI via optical music recognition processes, is thus a typical component of many MEI-project workflows. However, automated translations rarely achieve results of sufficient quality, a problem well-known in the community and documented in the literature. Final correction and validation by hand is therefore a common requirement. In this paper, we present mei-friend, an extension to the Atom text editor, which aims to relieve the degree of manual labour required in this process. The tool facilitates most common MEI editing tasks including the insertion and manipulation of MEI elements, makes the encoded score visible and interactively accessible to the user, and provides quality-of-life conveniences including keyboard shortcuts for editing functions as well as intelligent navigation of the MEI hierarchy. We detail the tool's implementation, describe its functionalities, and evaluate its responsiveness during the editing process, even when editing very large MEI files.

Introduction

The manual encoding of non-trivial music scores using MEI currently requires extensive specialist knowledge, both of XML syntax, processes, and workflows more broadly, and of the MEI schema in particular. Even with such expertise in place, the encoding activity is laborious and is usually frequently interrupted to re-render and inspect the notation corresponding to the encoding, typically using the Verovio\(^1\) MEI engraving toolkit.

There are alternatives to manual encoding from scratch, though none afford the same level of control over the quality of the final encoding outcome. Pre-existing encodings in other formats, e.g., MusicXML, or the Humdrum **kern data format, may be identified and translated to MEI, though this is liable to introduce conversion issues; or, optical music recognition (OMR) tools may be used to generate encodings from score images, though these tend to be error-prone and usually do not produce MEI natively; or, scores may be typeset using the graphical user interfaces of commercial notation software, albeit with similar limitations on native MEI-support (though in the case of Sibelius, the Sibmei\(^2\) extension may be used to accomplish this).

In each of these approaches, once the rough process of generating or bootstrapping digital score encodings and converting them to MEI is accomplished, a manual editing phase is required in which the MEI is checked for errors and inconsistencies, polished, and finalised if high-quality encodings are to be achieved.

\(^1\) https://www.verovio.org (accessed January 12, 2022).

© Published in 2022 with Humanities Commons under a CC BY-NC-ND 4.0 (https://creativecommons.org/licenses/by-nc-nd/4.0/) license.
Background

Imperfect Nature of Encoding Format Conversion

Challenges of translating between encoding formats are well known in the community and have been the subject of research in recent years:

- [7] surveys the classes of problems arising in conversion between music encodings in **kern, Lilypond, MEI, and MusicXML formats;
- [6] demonstrates how apparently equivalent encodings in different encoding formats can lead to discrepancies in the outcomes of music analyses;
- [8] considers how methodological choices undertaken during the generation and processing of music encodings may impact on conversion success in unintuitive ways.

The problems outlined in these studies can in part be addressed by improving the conversion processes themselves, particularly where whole classes of problems arise from flawed conversion of frequently used encoding constructs. We have worked in this direction by improving Verovio’s MusicXML to MEI conversion process and have promoted community involvement by participating in the organisation of a “Developing Verovio” workshop at the 2020 Music Encoding Conference. But even when the conversion process is improved as far as possible, there will be certain aspects benefiting from final manual ‘polishing’, due to semantic inconsistencies between encoding formats. Conducting such manual MEI finalisation in a text editor is labour intensive. Schema-aware but music-naive XML editors, such as Oxygen, partially alleviate the process by automating validation tasks. Like plain text editors, however, they (somewhat unintuitively) situate all interactions with music notation in the text (XML) domain, placing the cognitive load of navigating and modifying the MEI encodings in a musically-informed way squarely on the user. Specialised MEI editors provide further assistance by incorporating interactive, dynamically-rendered scores as additional means of navigation and interaction and by providing shortcuts to accomplish the most frequent notation-editing operations.

Editing MEI

Verovio’s affordances for graphical interaction with encoded music notation [9] have resulted in the creation of several score editors incorporating its JavaScript toolkit to render and manipulate encoded scores. Neon is a web-based square-notation editor aimed at correcting encodings of neume notation generated via OMR [10]. Alongside support for essential elements of square-notation music, including neumes (punctums, virgas, custos), basic neume groupings (e.g., pes, clivis, torculus), and C and F clefs and custos, there is also built-in support for displaying and editing text (lyrics) and interactive integration of source image facsimiles using the International Image Interoperability Framework (IIIF) [11]. Another editor for MEI encodings of neume notation, monodi+ [1], focuses on incorporating editorial variants and capturing hierarchical structures in medieval monophonic music, and it employs a custom rendering engine in place of Verovio. Both editors provide powerful tools for the transcription, validation, and finalisation of MEI encodings, but they are limited to neume notation and consequently do not support the comparatively more complex notation context of CMN.

Verovio-Humdrum-Viewer [12] is an elaborated editor for CMN and mensural notation developed for **kern. It supports a variety of basic editing operations through a graphical user interface, alongside an integrated text editor for more complex changes.3 It has been used to generate a substantial corpus of encodings, a notable ongoing endeavor being the encoding of the entire Chopin first editions.4 Beside its native support for **kern, the tool is able to import encodings in MEI and MusicXML and to export encodings to MEI and (subsequently) render them as SVG using Verovio. However, it does not support editing of MEI directly; all editing operations are performed in **kern prior to MEI conversion. This limits the editor’s applicability for finalising MEI encodings, as the restriction of the editing functionality to the **kern format means that it cannot be used to address post-MEI-conversion issues or inconsistencies. Further, this division into ‘read-write’ **kern and ‘read-only’ MEI introduces addressability issues into the generated MEI encodings. In converting from **kern, the tool mints

---

new @xml:id (local XML identifier) attributes for the created MEI elements according to their corresponding **kern-native line and column numbers. As a consequence, any (**kern) editing operations that affect the number of lines (e.g., adding a new note) will re-assign the identifiers in the corresponding MEI to potentially different elements. While this is not a problem in terms of local addressability within the updated MEI file as the new identifiers retain their internal consistency, it does violate the persistence of identifiers for use in a global context, as required, e.g., in Linked Data applications, where MEI elements may be externally addressed using fragment URIs [16].

Verovio has been integrated into the Atom⁵ text editor desktop application using an extension called mei-tools-atom [5]. This tool embeds a dynamically updating Verovio score SVG within an editor pane displayed alongside the MEI (XML) text buffer. It offers basic navigation aides, such as clicking a rendered score element to jump to the corresponding line of MEI or selecting a line of MEI to highlight the corresponding rendered score element on the current page. However, its functionality is otherwise limited, the integrated version of Verovio is outdated, the code has not been updated for two years, and the GitHub code repository of the project has recently been archived (set to read-only) by its owner.

Finally, a prototype editor has been built into the Verovio toolkit. This editor has the advantage of being a native part of Verovio, requiring no other dependencies, and supporting the development of Web applications that interact directly with MEI encodings through the Verovio toolkit⁶ via vrvTk.edit({editCommand}) function calls. However, only a small set of basic edit commands (such as inserting and deleting notes, and modifying or setting attributes of existing elements) is currently supported, with many functionalities important in the MEI finalisation process yet to be implemented.

The mei-friend Package for the Atom Text Editor

Use Case: Encoding Beethoven’s Piano Works

TROMPA (Towards Richer Online Music Public-domain Archives) is a project building on the semantic affordances of MEI encodings to address the music information needs of wider audiences, with distinct project strands focusing on music performers and enthusiasts, as well as on scholars [15]. The facilitation of the MEI encoding process has been one focus of research in this project, as the relatively small number of available, publicly licensed encodings (relative to TROMPA’s target repertoire of European classical music) necessarily limits the applicability of the developed user-facing prototypes.

One avenue of research has been into the crowd-generation of MEI by parcelling up the process of validating and fixing OMR-generated encodings into smaller (e.g., measure-level) units and farming these out to larger communities. This approach is relevant where a larger group of music experts share a time-sensitive interest in the generation of a new encoding and thus was developed with a focus on orchestra use cases [3]. TROMPA’s development of this approach is partly informed by the OpenScore Initiative [2] which has applied comparable crowd-based processes in the generation of MSCX (MuseScore) format score encodings.

However, such crowdsourced encodings still benefit from a finalisation step, and other use-case contexts do not necessarily include ready access to a motivated crowd and so stand to benefit all-the-more from a facilitation of the encoding process by individuals using an MEI editor.

To illustrate our requirements, we briefly describe the generation of solo-piano encodings for TROMPA’s instrumental players use-case [13, 14]. The recent 250th birthday of Ludwig van Beethoven and the available encodings of his 32 sonatas for piano in the **kern format⁷ initially motivated the attempt to encode the remaining piano solo works by Beethoven into MEI format. These pieces comprise several variation cycles (among them the Eroica Variations, Op. 35, the Diabelli Variations, Op. 120, or the C-minor Variations, WoO 80), Rondos (Op. 51), Bagatelles (Opp. 33, 119, 126), the Polonaise (Op. 89), Sonatinas (WoO Anh. 5), Phantasie (Op. 77), and pieces, such as Rondo a Capriccio, Op. 129, and Clavierstück “Für Elise”, WoO 59. As the target edition to encode, we chose Breitkopf and Härtel’s first complete edition from 1862–1868, fully available as scanned,

---

publicly licensed images at IMSLP. The aim of the encoding was to follow the target edition as closely as possible using the current version 4.0.1 of MEI as engraved by Verovio (currently version 3.3). The corpus encoded now comprises 18 works by Beethoven spanning over 220 pages, all published under open license on GitHub.

In encoding this corpus, we began by scanning a modern commercial edition of each score using a Konica Minolta C308 printer at 300dpi grayscale at A3 landscape with the saturation set 2 steps darker; this provided considerably better OMR results than when we started with the publicly licensed PDFs of the target edition directly. The derived PDFs were rotated, and black margins cropped, using Adobe Acrobat Pro (Version 9). The proprietary Neuraton PhotoScore application (Version 2018 8.8.7) was used to import the rotated and cropped PDFs at highest resolution and to perform OMR. Evident OMR errors, such as incorrect key signatures, meter indications, triplets, or notes, were fixed in-application, and the outcomes were exported to uncompressed MusicXML format. The resulting files were imported into MuseScore (Version 3.1.22425) to clean up further OMR errors and to adjust the encoding to the target edition.

Subsequently, the encodings were re-exported as MusicXML and converted to MEI using Verovio. During this step, we developed fixes for several classes of conversion error, including: support for ending elements, improvements in handling slurs, ties, hairpins, clef changes, cross-staff notes, pedals, arpeggios, turns, and backup elements. These systematic errors were addressed at source through contributions to Verovio's open-source C++ codebase. The MEI encodings were then processed algorithmically to remove duplicated @accid and @accid.ges attributes, insert missing @xml:id, and to renumber the measures, taking into account measures that do not conform to the meter signature (@metcon="false") and measures inside multiple endings.

Having arrived at this stage, the MEI encodings were largely complete, correct, and faithful to the target edition; but further laborious steps were required to undertake final editing of details, such as inserting slurs and/or precisely placing their beginnings and endings, and adjusting their direction of curvature, inserting or fixing dynamics markings, directives, ornaments, pedal markings, etc. To illustrate: The insertion of a slur requires the @xml:id attributes of the target notes at two distinct positions in an encoding to be identified and a line of code to be added to the end of the measure containing the first note or chord. Manually, this operation is quite time consuming and repetitive, as slurs are often misplaced or completely missed during the OMR workflow described above.

**Implementation of mei-friend**

Specialised MEI editing software was required to ease this finalisation process, but none of the currently available editors (see Editing MEI) fulfilled the requirements of our use-case for CMN notation. We gratefully made use of the existing open-source (MIT-licensed) code for mei-tools-atom, forking it with a new name (to reflect the new project direction) and housing it in a new GitHub repository [4] under the same license.

mei-friend is written in JavaScript using the Atom package framework and automatically loads the latest release version of Verovio as a dependency using the Node package manager (npm). Through the provided text-Editor object, mei-friend interacts with Verovio's JavaScript toolkit to ingest the text buffer (MEI) content and to render the corresponding SVG, displaying it in a repositionable panel within the Atom application window, alongside the panel displaying the text buffer. User interactions with both panels are supported, allowing, e.g., for the selection of a line of text to highlight the corresponding rendered score element(s), and vice versa. Atom also provides tight git integration, supporting easy and fast interaction with MEI encodings hosted on GitHub.

---


Figure 1: Annotated screenshot of the mei-friend package for the Atom text editor, showing an excerpt of the encoding of Beethoven’s Diabeili Variations (top), its automatic rendering with Verovio (middle) and the same system in the target edition by Breitkopf & Härtel (bottom, another window outside the Atom editor).

Figure 1 illustrates the actions for display, navigation, and editing available to the user through the main menu bar (from left to right):

- viewing controls to scale the score in the notation panel and to invert the notation colours (black on white, or white on black);
- controlling page turning (jump to first, previous, next and last page) and flipping the score page to correspond to the current cursor position in the text buffer;
- setting Verovio’s --breaks option, and toggling between normal and speed mode;
- setting the updating (score re-rendering) behaviour: automatic, after each edit to the text buffer; or, manually, upon clicking the ‘update’ button;
- selecting SMuFL font for notation;
- navigating through the notation by modifying the currently-selected note or rest, via button clicks or key-bindings (note-wise back-/forwards, measure-wise back-/forwards, page-wise back-/forwards, layer-/staff-wise up/down);
- re-processing (loading and serialising) the MEI XML text-buffer contents through Verovio to standardise the order of encoded elements and insert missing @xml:id’s (or remove unnecessary @xml:id’s);
- displaying Verovio’s current version, and the key bindings help panel (see Figure 2).
The tool’s editing capabilities currently comprise inserting control elements (slur, tie, hairpin, dynamics, directive, tempo indication, fermata, trill, mordent, turn, arpeggio, glissando, pedal, octave) or layer elements (beam) on selected elements and manipulating some of their properties (curve direction, placement, and stem direction). This is achieved using specialised keyboard shortcuts as listed in the help panel (Figure 2). The underlying logic is to select one or two notes and to use a keyboard shortcut to insert new control elements into the MEI code. For example, pressing the S key with two selected notes inserts a slur at the end of the measure containing the first selected note, the slur spanning from the first to the second selected note. Subsequent key-presses of X toggle the placement (@curvedir) of the slur from automatic (attribute not set) to “above” to “below”.

Combining the keyboard shortcut with the CTRL key inserts the control element with the @curvedir attribute pre-set to “below”. There are specialised key-bindings for other elements: e.g., crescendo versus diminuendo hairpins, or starting versus ending pedal markings. Some insertion operations are more complex: for example, when inserting an octave shift element spanning two selected notes in the same staff, the @oct.ges attributes of all notes falling between the two selected will be modified accordingly to ensure valid MEI (and will be reset when a selected octave element is deleted). The complete list of key bindings is given in Figure 2 showing the help panel of mei-friend.

Further functionalities include:

- toggling of articulation elements (staccato, marcato, ...) on selected notes, chords, or higher-level units containing those, such as beams. Following the logic mentioned above, the placement of articulations can be inverted by pressing the X key;
- deleting inserted elements with the DELETE or BACKSPACE key. Currently supported are all control elements, beam, clef change, accidentals, and articulation, moving elements up and down in pitch changing the @pname/@oct for notes or the @ploc/@oloc attributes for rests (including mRest and multiRest), and moving elements up and down staffwise to realise cross-staff notation of piano scores;
- support for specific encoding manipulations, such as re-assigning measure numbers, which are liable to be misattributed during OMR and encoding conversion processes where incomplete measures may be missed, or removing doubled @accid.ges attributes when an @accid attribute is present.
Beside supporting control element insertion, *mei-friend* improves upon *mei-tools-atom* by supporting: better dependencies management (now loading the latest release version of Verovio via the Node package manager, and showing the Verovio version number in the user interface to help diagnose conversion issues), user interface improvements (introduction of tooltips, a help panel and documentation, keyboard shortcuts, grouping of icons, and navigation inside the notation panel through buttons or key-bindings), and facilitated interaction with Verovio (setting breaks options; selecting SMuFL font; re-processing the MEI XML through Verovio to provide consistent formatting, fix ordering, and generate any missing @xml:id). 

**Performance Considerations**

Large MEI encodings (of about 25,000 lines and more) cause performance issues with Verovio, affecting both *mei-tools-atom* and *mei-friend* (in its default “normal mode”). This is because the entire MEI encoding must be parsed and laid out by Verovio on import before the rendered SVG for a given requested page can be returned for display in the user interface (see Figure 3a). As the current implementations of both Atom extensions are single threaded, the corresponding time-intensive loading and re-rendering operations make the user interface increasingly unresponsive with larger files. When editing the MEI directly in the text buffer, automatic updating of the rendered notation involves repeating this intensive process after each edit. To circumvent this issue, automatic updates may be disabled in *mei-friend*’s interface, instead allowing the user to request re-rendering of the notation manually on button click.

![Figure 3: Processing flow chart from encoding to notation engraving. a) In “normal mode”, the entire content of the encoding is passed to the Verovio toolkit. After the notation layout for the entire encoding is processed, a selected page is rendered as SVG. b) In “speed mode”, the MEI encoding is loaded into the Document Object Model (DOM). To render the requested page, its corresponding MEI elements are extracted from the DOM and used to generate an artificial MEI encoding with dummy pages placed around the requested page that incorporate single-note ‘anchors’ for spanning elements. Verovio is then used to render the requested (middle) page.](image)

To further ease the loading and reprocessing requirements posed by large files, we have implemented a “speed mode” (see Figure 3b), which currently operates only with encodings containing break elements (page and system breaks) and requires a correspondingly set --breaks option in the interface. Instead of requiring the entire MEI encoding to be processed, speed mode only transfers a single-page excerpt encompassing the most recently selected element in the text buffer to Verovio. Two additional ‘dummy’ MEI pages are generated
to surround the MEI excerpt to be displayed. The preceding page contains a score definition that is updated to
the latest clef, meter, and key signature change preceding the MEI excerpt. Both dummy pages each contain a
measure with staff and layer elements, and a single note from and to which time spanning elements (such as
slurs and hairpins) are connected, so that the middle page (corresponding to the excerpt) is able to show these
elements correctly. These excerpting and inserting operations are performed on an XML DOM (Document Object
Model) representation of the MEI encoding, allowing them to be processed very efficiently. In this mode, only a
constantly small portion of the MEI encoding (the excerpt) is transferred to Verovio, limiting its processing load
and thus keeping interactions swift and smooth.

We evaluated the performance using Atom's internal Chromium profiler by measuring the difference in re-pro-
cessing time (i.e., user interface unresponsiveness) to quantify the increase in performance of our “speed mode”
implementation, using an MEI encoding of Beethoven's WoO 57 (Andante favori, 11.8k lines of MEI) and two arti-
ficial encodings (50k and 100k lines of MEI),\(^\text{10}\) which we generated by adding the mdiv elements of several pieces
together. Table 1 displays the median amount of time (in ms) taken to i. open the file, ii. flip to the next page, iii.
flip to the last page, and iv. insert a slur, in normal and speed mode, calculated across three repetitions on a 2017
iMac with a 4.2 GHz Intel Core i7 running OS X 10.14.6, Atom 1.54.0, and mei-friend 0.3.3. Results demonstrate a
significantly improved performance and user experience in speed mode working with large MEI encodings when
performing initial loading of the file and re-loading after edits. Page flipping was slightly slower, performance
deteriorating with later page numbers, as more DOM processing is required to track score definitions.

Known issues and limitations of speed mode remaining to be addressed include potential inconsistencies with
normal mode when rendering automatic curvature directions for slurs spanning across the excerpted measure
and not rendering time spanning elements using time stamps rather than start-/endids. Speed mode can be
enabled and disabled using a checkbox.

<table>
<thead>
<tr>
<th></th>
<th>WoO 57 (11.8k lines)</th>
<th>Beethoven (50k lines)</th>
<th>Beethoven (100k lines)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normal mode</td>
<td>666 / 37 / 40 / 625 ms</td>
<td>2290 / 30 / 35 / 2301 ms</td>
<td>6760 / 31 / 36 / 6870 ms</td>
</tr>
<tr>
<td>Speed mode</td>
<td>143 / 60 / 93 / 71 ms</td>
<td>204 / 38 / 142 / 45 ms</td>
<td>322 / 42 / 239 / 48 ms</td>
</tr>
</tbody>
</table>

Table 1: Evaluation of mei-friend performance in normal mode and speed mode for i. opening file / ii. flipping to the next page / iii. flipping to the last page, and / iv. inserting a slur in different MEI-encoding-size contexts.

Planned Functionalities and Future Development

Several improvements, partly requested by users through GitHub issues, have been implemented since the
initial submission of this article. Further potential developments include support for speed mode in all break op-
tions, and porting this package to the Visual Studio Code editor,\(^\text{11}\) which tends to perform better than Atom and
includes a similar JavaScript extension mechanism. Ultimately, we envision an integration of the functionalities
presented here with browser-based crowdsourcing mechanisms along the lines of those developed within the
TROMPA project.

Conclusion

Although MEI is increasingly adopted for use in music edition and scholarship, the lack of native support by com-
mercial notation software has limited the extent of the available MEI-encoded repertoire. While the mei-friend
package we have presented here is unlikely to suddenly open up MEI encoding to the masses (it still requires
extensive technical expertise from its users, both in XML more broadly and in the MEI schema specifically), it
does significantly alleviate the burdens of the final stages of MEI encoding. As most typical processes used for
generating MEI encodings employ stages in their workflows liable to introduce errors or deviations that require
manual intervention, we hope that mei-friend will contribute toward making validation and fixing of the corre-
spending process outcomes less painful and more enjoyable.


We must acknowledge limitations in the work presented here. The project context that gave rise to mei-friend has been characterised by a predominant focus on European classical piano music, particularly Beethoven’s works (among others). It is likely that functionalities may have been identified as useful in other encoding contexts, but are currently missing because they simply didn’t arise in our use. We hope to collectively work with the MEI community on the continued development of mei-friend to further increase its usefulness in the future.

Acknowledgements

This project has received funding from the European Union’s Horizon 2020 research and innovation programme “Study European heritage, memory, identity, integration and cultural interaction and translation, including its representations in cultural and scientific collections, archives and museums, to better inform and understand the present by richer interpretations of the past” (H2020-EU.3.6.3.1.) under grant agreement No. 770376. We thank Sienna M. Wood and the nCoda team for contributing the mei-tools-atom package, and Laurent Pugin for his valued contributions and advice in the development of mei-friend’s integration of Verovio.

Works Cited

Building a Comprehensive Sheet Music Library Application

Klaus Rettinghaus    Alexander Pacha
Enote GmbH, Berlin, Germany   Enote GmbH, Berlin, Germany
klaus.rettinghaus@enote.com   alexander.pacha@enote.com

Abstract

Digital symbolic music scores offer many benefits compared to paper-based scores, such as a flexible dynamic layout that allows adjustments of size and style, intelligent navigation features, automatic page-turning, on-the-fly modifications of the score including transposition into a different key, and rule-based annotations that can save hours of manual work by automatically highlighting relevant aspects in the score. However, most musicians still rely on paper because they don't have access to a digital version of their sheet music, or their digital solution does not provide a satisfying experience. To bring digital scores to millions of musicians, we at Enote are building a mobile application that offers a comprehensive digital library of sheet music. These scores are obtained by a large-scale Optical Music Recognition process, combined with metadata collection and curation. Our material is stored in the MEI format and we rely on Verovio as a central component of our app to present scores and parts dynamically on mobile devices. This combination of the expressiveness of MEI with the beautiful engraving of Verovio allows us to create a flexible, mobile solution that we believe to be a powerful and true alternative to paper scores with practical features like smart annotations or instant transpositions. We also invest heavily into the open-source development of Verovio to make it the gold standard for rendering beautiful digital sheet music.

Introduction

Many parts of our lives have become digital in the last decades, from navigating on the street to buying groceries. This digitization allowed us to do things we were not able to do several years ago, like cars that are driving autonomously. In other areas of our life, it helped to make existing processes more efficient or more convenient.

The digitization of music scores is one of these areas. For over 50 years researchers and companies have been working on typesetting music on a computer [18], digitizing scores [14], or collecting and publishing a digital library of public domain scores [13]. These projects have led to many products that are widely used today (e.g., MuseScore, Finale, Sibelius, and Dorico are used by millions of musicians to typeset music scores). However, when practicing and performing, many musicians still rely on printed music scores for understandable reasons: they are universally available, easy to handle, and do not suffer from technical problems like an empty battery. Even more so, existing digital alternatives are often not a full replacement for paper or fail to unlock the full potential of digital scores, and therefore get rejected quickly. For example, taking a photo of a paper score and opening the image on a tablet is a digital solution that reduces the amount of paper that needs to be carried around but might suffer from a loss of quality. Instead, a machine-readable, symbolic version of the music score can be obtained through Optical Music Recognition (OMR) [2]. Recent research suggests that many challenges of OMR can be solved quite robustly with machine-learning approaches [1, 20, 21]. However, these are never guaranteed to be completely error-free, and even if a flawless digital score can be obtained in the end, e.g., by manually typesetting it in a music score editor, a satisfying solution for musicians must include a way to easily view and interact with that score.

To build such a solution, the start-up Enote GmbH was founded in 2018 by Boian Videnoff, Josef Tufan, and Evgeny Mitichkin to revolutionize digital sheet music. Enote is based in Berlin and has around 30 employees working on a mobile application for iOS that musicians can use to display and interact with our digital music scores. In this paper, we want to give some insights into our solution as well as why and how we use MEI and Verovio for our product.
Digital Music Scores Challenges

There are several major challenges when creating a music score application like Enote, including questions on how to obtain the material that the users will see, how to ensure a high quality, how to store the digital music scores, and how to create a satisfying digital experience on a mobile device? Figure 1 gives an overview of our digitization efforts that are discussed in the following chapters.

![Diagram of digitization process]

Figure 1: Our digitization efforts in a nutshell: Scores and Metadata are obtained from several public sources. Paper-based scores are converted into digital symbolic scores by an Optical Music Recognition system and manually corrected and enriched with corresponding metadata to build a verified Digital Music Score Library that can be displayed on a mobile device with Verovio.

Building a Digital Music Score Library

Thousands of music scores have been written in the past. They exist in many forms and can be obtained from several sources including books or existing digital collections like IMSLP, MuseScore.com, or CPDL [10]. Given the legal protection of music (scores), an agreement with the copyright holders must be obtained, or one has to restrict oneself to copyright-free material. For the beginning, Enote settled with the second option, focusing on public-domain music that can be used and distributed freely.

Although many scores can be obtained or purchased, the quality of the material varies significantly, reaching from poor scans of printed music to digitally encoded scores of very high quality [6]. Unfortunately, free, high-quality, digitally encoded music is only sparsely available. Therefore, to ensure the quality of our material, we are developing a highly-accurate Optical Music Recognition (OMR) system that uses state-of-the-art machine-learning approaches and can create a digitally encoded version of the available music scores. Several machine-learning approaches have been shown to work quite well for visually recognizing objects in typeset and handwritten music scores [11, 19]. These approaches have in common that they make heavy use of deep convolutional neural networks. However, even if the visual recognition works well, a complete OMR system also requires a semantical reconstruction stage [16] that builds a suitable musical representation [8, 12] out of these detections, which can be exported into a common digital music format like MusicXML or MEI. The difficulty of building this stage is frequently underestimated and has not received a lot of attention from the research community. Most existing OMR systems attempt to solve this challenge with an extensive set of hand-crafted rules.

Rule-based systems can give quick results but become harder to maintain and expand over time. One reason is that music notation is often used as freely as the music it encodes. Many rules and conventions of Common Western Music Notation [7] have evolved over time and composers as well as editors frequently bent the rules to capture a musical idea with the tools they had at that time (see Figures 2 and 5). This leads to the problem that regardless of how well an OMR system works and how many rules you implement, there are always exceptions that require special treatment. And the more material you collect, the more questionable editorial decisions you will discover (even in many modern editions from esteemed publishing houses). In [3], the authors proposed an interesting idea on how to deal with this problem, by allowing the operator of an interactive system to dynamically enable and disable rules and constraints during reconstruction, allowing the system to handle exceptional cases gracefully while working robustly on most scores with a reduced set of constraints.
Given the current state of the art, any OMR system that aims at producing flawless, digital scores requires manual verification, more often some form of manual correction. Our musicology department is in charge of this process, along with the selection of pieces that are digitized, the consolidation of multiple sources, and advising the software engineers in music-related questions. Scores that have minor errors are manually corrected by our musicologists in a slightly adapted version of the Humdrum Verovio Viewer [17], which is also publicly available.

Collecting Metadata

The most complete music library would still be pretty useless if you weren’t able to find what you’re looking for. To make the library usable and searchable, additional metadata is required. Therefore, we are complementing all of our pieces with extensive metadata that we’ve collected, such as information about the composer or the instrumentation. Although it is possible nowadays to fetch a lot of metadata directly from sources like IMSLP or Wikidata – that alone is not enough. The collected data can be inconsistent, incomplete, or generally of poor quality. And even if you start an intense search, some information can be genuinely hard to obtain, e.g., information about the duration of a piece, even though several recordings might exist.

After collecting the metadata, you have to make sure that users can actually find what they are looking for. Although this can seem like an easy task, it has become very tricky in a diverse and open field such as music. Think of terms like Charakterstück or Motet, which can mean something quite specific or almost anything depending on your point of view. What would you expect to find if you search for Charakterstück? Works by Couperin, only works of the 19th century, or even just pieces that contain that specific phrase in their title? Not to mention the many possible spellings of non-western composers, like Aram Khachaturian, Sergej Rachmaninoff, or Tōru Takemitsu, which is a well-known problem in information science, but cannot yet be handled sufficiently by relying on authority controlled data alone. Even public authority records like those from the Library of Congress have substantial gaps in this regard (see Figure 3).

You may clean up and sort some of the metadata automatically, but a rule-based system quickly reaches its limits here. Our musicology department invests extensive brainpower in compiling, completing, and consolidating the metadata to help musicians find what they are looking for in our music library. We do so by comparing our information to the latest available scientific catalogs of works provided by scholars all over the world.

Storing the Musical Data

For storing our digital sheet music we decided to use MEI [9], as it brings some major advantages over several other non-binary, open music encoding formats we evaluated, such as ABC, Guido, Humdrum, or MusicXML. While rumors have it that MusicXML [5] will more or less reach its end-of-life with the latest version 4.0 release, its designated successor MNX is still only a draft specification. MEI on the other hand is under active development and has seen multiple iterations over the last couple of years. MEI is also quite well documented – at least compared to MusicXML. This is one of the reasons why different implementations of MusicXML diverge substantially from one another and, based on our experience and testing, not a single one seems to be feature-complete. The status of ABC is quite unclear, as the next version to be released is in the making since early 2013, and its documentation website keeps disappearing from time to time.
One of the strongest arguments for MEI is the ability to extend the format to fit personal needs in a tailor-made customization, while also allowing us to participate in its active development as part of the community. A huge advantage over similar formats is MEI's 'semantic approach' in encoding musical information. Take the encoding of hairpins as an example: MusicXML connects their start- and endpoints to noteheads. While this sounds like a sensible choice, it can become a problem in certain situations, like a hairpin ending on a barline. In MusicXML, you have to manipulate the hairpin graphically by specifying a relative offset to achieve the desired visual result. This makes encoding of a *messa di voce* messy and will probably result in a broken layout if you compress or stretch the score. In contrast, MEI allows you to set the beginning and ending of hairpins within the flow of the music and thus allows for a truly dynamic layout.

MEI is also well-suited for OMR applications, as it allows to explicitly model the graphical information of the underlying source, which cannot be done in MusicXML. By using native features (taken from TEI), it is possible to link each semantic musical object exactly to the respective graphical object in the source, which, e.g., allows features like a visual playback overlaid on a scan of the original printed score.

---

**Figure 3:** Excerpt of the Library of Congress's authority record for Aram Khachaturian showing several variant spellings – his original Armenian spelling, however, is missing here.
Presenting Scores

To display the sheet music in our mobile application, the Verovio music notation engraving library is used [15]. We decided to use Verovio for score rendition not only because it is the natural choice to engrave MEI files but because it is a fast and reliable cross-platform application. Furthermore, it is free open-source software, which allows us to enhance it with features we need.

Figure 4: Comparison of a small excerpt from Carl Phillip Emanuel Bach’s “Prussian Sonatas”, rendered in our four different notation styles, using different font faces. From top to bottom: classic, traditional, legacy, and modern.
There are already other mobile apps available using Verovio, like Cantico, NomadPlay, and Trala, but we make heavy use of the possibilities to use different SMuFL compliant fonts and styling features. For example, we offer four different styles for displaying the music, which we named classic, traditional, legacy, and modern (see Figure 4).

Our legacy style, for example, uses a custom-created font named Legato [4], which is based on Steinberg’s open-licensed Bravura font. We also extended Verovio’s own Leipzig font with additional glyphs to cover a broader range of the Standard Music Font Layout.

We strive for a consistent look and feel throughout our scores. That is why apart from simply correcting reconstruction errors, our musicologists and engraving specialists also enforce a consistent style across our library.

In order to improve Verovio with new features that we need, we invest heavily in its development. Basically, all our changes and improvements are sent directly back upstream, so we get a review from the maintainer, and the community immediately benefits from our work. Also, all our development efforts on Verovio are visible in our public fork on Github. So even if changes are not merged back into Verovio’s main repository, you can still get those features from our fork.

Figure 5: Example for visual improvements in Verovio. Before (left) the tempo marking was positioned poorly, while now (right) it includes preceding accidentals as proposed in Behind Bars [7].

Some of the visual improvements we’ve made so far include

- automatic cross staff rest positioning,
- rendering of ties on dotted notes,
- alignment of dots in different voices,
- horizontal layout with dots and flags,
- alignment of tempo markings (see Figure 5),
- rendering of octave bracket endings.

We introduced several new options and recently we added support for additive meter signatures (see Figure 6).

Figure 6: The first two measures in bulgarian rhythm from the very last piece from Béla Bartók’s Mikroközmosz showing an additive time signature in Verovio’s rendering.
One major problem remains: Verovio tries to follow the rules and conventions of music notation, as recently collected and presented by Elaine Gould [7] and others before her. In printed scores, however, you find these rules very often extremely bent or even broken (see Figure 7). But even if you want to address all possible edge cases of notation, Verovio can only produce layouts based on a determined (and finite) set of rules. And trying to reproduce bad or nonsensical layout would contradict our goal of a unified user experience mentioned above.

**Figure 7:** In this excerpt of Chopin's Ballade No. 1 in G minor, Op. 23, the editor decided to squeeze the quarter rests in the upper system between the noteheads by making it smaller than other rests, although there would be other (more consistent) ways to notate this. Also, it is not totally clear why rests are shifted vertically in the lower staff.

It is yet unclear how to deal with such cases, but you certainly have to deal with them. Interestingly, editors tend to stick with erroneous layouts when they prepare new editions of scores, even if there would be solutions that better comply with the standard rules of notation and are more pleasant to the musician’s eye. Unfortunately, to clear up problematic situations, appropriately handle dense scores, or automatically do a complete re-layout, a computer would need to have a deep understanding of the music – maybe even deeper than their original editors.

**Conclusion**

Digital symbolic sheet music offers several advantages compared to paper-based scores. However, to offer musicians a true alternative, more than just a collection of PDF documents is needed. Our efforts are directed towards building a comprehensive sheet music library application, made (not only) for musicians. We are collecting and digitizing thousands of music scores, enriching them with vital metadata, and offer our users a digital solution that unlocks the full potential of digital sheet music. In building a comprehensive sheet music library application, we see a viable business model, but equally importantly a step towards the preservation of our cultural heritage. We also see an opportunity to break with the habit of constantly copying poor or faulty notation. Our artificial intelligence plays a key role in our efforts by automatically reporting situations that require additional manual work so we can bring the music of our ancestors to musicians worldwide.
Works Cited


Lohengrin TimeMachine: Musicological Multimedia Made with MELD

David Lewis
University of Oxford e-Research Centre, UK
david.lewis@oerc.ox.ac.uk

Kevin R. Page
University of Oxford e-Research Centre, UK
kevin.page@oerc.ox.ac.uk

Laurence Dreyfus
Faculty of Music, University of Oxford, UK
laurence.dreyfus@magd.ox.ac.uk

Abstract

Music and the scholarship around it can be challenging to present in the forms associated with books and articles – primarily linear and with an emphasis on the static and visual over the sonic and interactive. We introduce the Lohengrin TimeMachine, a multiple-path multimedia app, optimised for a touch-screen tablet. The app offers two essays about motifs in the opera – one in textual form, and one a 30-minute video. These linear narratives are supported by audio examples, along with dynamic links that take the reader into a fully-interactive exploration of the occurrence of motifs across the opera. The reader's journey is supported with recorded music and scores, as well as novel visualisations of the orchestration and the timeline of the opera. The app is designed to operate over standards-based web documents published online, with extension and reuse in mind. In this paper, we describe the app, its underlying technology, and the journeys it supports.

Introduction

Musicology is often presented in linear, textual form, even though its materials are complex and multimodal. Scholarship must necessarily simplify and dramatically reduce complexity in order to make an ordered prose argument that others can follow. More detailed interaction with the scholar’s observations occurs almost entirely through other resources.

We present the Lohengrin TimeMachine, a web application in which linear musicological scholarship (in the form of an essay and a 30-minute video) is augmented by user-driven interactive, multiple-path exploration of related material that can be explored with or independently from the narrative. The material, which includes musical notation, audio, analytical content, and textual quotation is accessed through a touchscreen-optimised web application for guided and self-directed exploration. Underlying resources are published on the web using standard file formats (including MEI and TEI) and associated with machine-readable Linked Data, supporting the creation of additional interfaces or analyses.

The application is intended to be accessible to enthusiastic amateurs as well as scholars, and concerns Wagner's use of motifs in his early opera, Lohengrin, advocating a more sophisticated understanding than 'Leitmotif' guides often communicate. It is optimised for use with a tablet, without excluding other means of browsing. The musicological content was conceived, written and, for the video, presented by co-author Laurence Dreyfus, a Wagner specialist, with the companion realised through a multi-disciplinary collaboration.

The Lohengrin TimeMachine allows users to explore the compositional devices Wagner uses to vary his motifs, browsing the whole opera for motif occurrences and their musical and textual contexts. Visualisations and recordings support the analysis, making it accessible to an audience that might otherwise struggle with a Wagnerian orchestral score. Exploration of this material can follow or be triggered by the musicological article and video, but can also be entirely reader-driven, with free browsing of the curated musical landscape.

© Published in 2022 with Humanities Commons under a CC BY-NC-ND 4.0 (https://creativecommons.org/licenses/by-nc-nd/4.0/) license.
The Application and Its Data

The application dynamically constructs views using data published openly using standard file formats and Linked Data frameworks [3]. Materials are retrieved live from a public web server. Music notation is published as MEI, and text, including the main article, as TEI. Relationships and more abstract information are made available using the Resource Description Framework (RDF),¹ serialised as JSON-LD² and N-Quads³: annotations use the Web Annotation model⁴ [1], while much of the musical information is represented using the Music Ontology.⁵

The sum of all these forms an independent, repurposable, and open Research Object [5]. Interactive user views are generated directly and dynamically in the browser from this knowledge graph using novel visualisations, which enable the user to navigate all possible paths through the evidential multimodal materials.

Technical Implementation

The TimeMachine is constructed as a MELD application – that is, it uses the MELD 2.0 (Music Encoding and Linked Data) framework⁶ [7]. MELD traverses Linked Data graphs, ‘following its nose’ by successively requesting web-accessible Linked Data – with each set of statements read potentially giving links to new URIs to explore. From this data, MELD builds up a local knowledge graph within the browser, allowing apps to discover, select and filter relevant information. The MELD framework provides reusable components for creating and retriev-

3  https://www.w3.org/TR/n-quads/ (accessed January 12, 2022).
Yet this iteration of "Fragenverbot" doesn’t repeat Lohengrin’s prohibition so much as issue a threat about the ban. To notice this shift is to pay attention to a subtle change in the X-segment. Instead of the "Fragenverbot" leading off with a heavy downbeat, Wagner has added a little anticipatory upbeat before the first note, which immediately evokes the opening signal of a brass fanfare (da-dâ). This additional upbeat even prevents the recall of Lohengrin’s opening words – "Nie sollst du mich befragen" – which began with a strong accent on the word "Nie" [Never].

The text of the "Fragenverbot" therefore recedes into the background, and the theme transforms itself instead as an orchestral fanfare of doom. It is as if a mighty instrumental choir taunts Elsa with a spiteful recollection of the promise she made. Note also the "Fragenverbot" is no longer heard in Lohengrin’s region of A flat and A minor, summoning forth the holy Grail, but is transposed downwards into Otfrid’s realm located a third lower of F (and F sharp) minor. As one can see from the TimeMachine, Wagner maintains this threatening upbeat into the next two iterations found in Act III 1 – "Fragenverbot" 10 and 11 – even when the tonal realm shifts back to...
The TimeMachine view is a central navigation point, both for selecting single iterations of the motif and for choosing a second with which to compare. A local knowledge graph is built within a certain number of traversal steps from that. The application then dynamically selects the components of the interface based on the information and relationships to media that have been discovered. Much of the interface is constructed of generic MELD components, which are then styled for the specific look of the app using CSS.

Using the Application

A user visiting the app website will initially encounter a landing page with links to several entry points for exploring the app (a diagram of paths the app offers between views is given as Figure 3). The essay and video themselves are integrated in the app, providing important narrative grounding and intuitive starting points. Readers or viewers are presented with relevant links into various other application views as they progress through the narrative (Figures 1 and 2).

Most links point to a form of the Inspector view (Figure 4). This gives the user a variety of information and visualisations about one iteration of the motif being discussed. In the terms of this app, an iteration of a musical idea is one instantiation of it at a particular point in the music (thus, the Frageverbot motif has 17 iterations in the opera itself, and one that occurs in a cut scene). Two iterations can be selected for side-by-side juxtaposition, using a separate Comparison view (Figure 5).

Selecting motif iterations from an overview of a complete opera can be daunting, and in this case is achieved through the TimeMachine view (Figure 6). Users can also flick left or right through motif occurrences, visualised in a carousel-like rapid overview.

Many types of information and visualisation recur within different page-level views. Audio and textual information, including libretto (or ‘poem’) content, are made available – this is both a crucial reflection of the integrated nature of Wagner’s artistic conception and a practical help to those for whom the score is not an easy way of engaging with the material.

Throughout the app, we provide two different visualisations for music notational content. Vocal scores are rendered from MEI using Verovio, with structural analysis (musical sections) dynamically overlaid (Figure 4, left pane). Activating these annotations triggers audio playback from the beginning of that section.

For full orchestral scores, a new, more abstract, notational visualisation of MEI is used, simplifying the complexity of a Wagnerian orchestral score. In our Orchestration pane, each instrument playing at a particular time is shown as a coloured ribbon, with the instrument’s section of the orchestra providing the colour (Figure 4, right pane). This makes clear differences in orchestration that may be made invisible by the reduction to vocal score. The visualisation is generated live in the browser based on a full score in MEI, and is rendered as an SVG image. It is implemented as a reusable React component within the meld-clients-core module, and so can be

![Diagram showing the main pages of the application and the flows between them. The TimeMachine view is a central navigation point, both for selecting single iterations of the motif and for choosing a second with which to compare.](image-url)
easily incorporated into other applications. The list of instruments or singers, and the divisions of the orchestra or choir can be customised for any specific application.

For an opera thousands of bars long, overviews are crucial. In addition to the TimeMachine view itself, every view has a timeline at the bottom, which shows all occurrences of a motif, providing a visual summary and a base for navigation. The view summarises the sequence within the opera, supporting quick comparisons, and also acting as an index to detail views.

Discussion

The application is possible because of a combination of factors. With a musicological study that was conceived by its author for interactive exploration, and collaborating on design with that author, we can ensure that the visualisations we provide are relevant and add to the scholarly argument and to a user’s understanding of the material.

On a technological level, the combination of web standards and formats – especially MEI – that are easily addressed using URLs supports building rich, specialised applications without the need for bespoke or closed data formats. Furthermore, open-source rendering libraries such as Verovio and CETEicean, which preserve to a large extent the structure of their originating documents and propagate IDs through from the XML to the rendered result, provide essential support for semantics on the data level to be transformed into interactive exploration on the application level in a principled and re-usable way.

The app itself is quick and responsive once loaded, and, although it takes 10–20 seconds on initial load to process and reframe the graph for use, we believe this to be within reasonable expectations for a modern
Figure 5: Comparison view, showing the Vocal Score panes for two iterations of the Frageverbot motif. A structural analysis (x and y segments) is indicated on the score. Both iterations are labelled clearly in the timeline, so their place in the opera is clear. Colour coding in the timeline of motifs in the ‘Magical’ and the ‘Grail’ realm (as magenta and turquoise lines, respectively) draws attention to and gives context for some of the differences in timbre and setting.

web application. This processing time itself depends substantially on the Javascript JSON-LD library,¹² and has improved by a factor of 10 as the library has matured.

By relying on walking Linked Data graphs to discover information and resources, MELD has no requirement for its materials to be collocated on the same server or published by the same institution. Although individual MELD components, such as the score component, have certain expectations for supplied parameters, the framework is not strongly opinionated or exclusive about the ontologies of its knowledge graph, and applications frequently transform graph data before it goes into these components. This makes the framework highly flexible for external data. There is no reason in principle why information and media could not be drawn from multiple external sources, such as Wikidata, Europeana, or YouTube, nor anything to prevent others building MELD apps using the Lohengrin TimeMachine’s resources.

The MELD framework has already been used for a variety of musically-rich interfaces. We believe that the Lohengrin TimeMachine can serve as a model for many more.

**Acknowledgements**

The Lohengrin TimeMachine was developed by the Arts and Humanities Research Council-funded Unlocking Musicology project (AH/R004803/1) as a companion to the essay “Asking a Forbidden Question: A new view of musical themes in Wagner’s Lohengrin” by Prof. Laurence Dreyfus (University of Oxford Faculty of Music). Unlocking Musicology builds on research undertaken during the Transforming Musicology project (AH/L006820/1) with important contributions from Carolin Rindfleisch. The MELD framework was created, with David Weigl as lead developer, as part of the Fusing Audio and Semantic Technologies for Intelligent Music Production and

We thank Will Elsom for his work on the Lohengrin TimeMachine design specification, Steve Pierce and Oxford Filmshed Productions for the video tutorial, and Ralph Woodward for his music engravings.

Consumption project, funded by the UK Engineering and Physical Sciences Research Council (EP/L019981/1).

Works Cited


Music Performance Markup: Format and Software Tools Report

Axel Berndt
Center of Music and Film Informatics
University of Applied Sciences and Arts OWL, Germany
axel.berndt@th-owl.de

Abstract
With Music Performance Markup (MPM) we introduce a new XML format for describing musical performances in a systematic way. The format builds upon a series of mathematical models that capture the characteristics of performance features such as continuous tempo and dynamics transitions, articulations, and metrical accentuations. Bundled with MPM comes an infrastructure of documentations and software tools. This paper aims to provide an overview of and introduction to MPM, its infrastructure and ongoing development activities.

1 Introduction
Music culture of the past century has been strongly influenced by audio media. Music has been and continues to be published on all kinds of audio media to an overwhelming extent. The study and analysis of such audio documents poses special challenges to musicology. Typical questions are: How can the connection between an audio document and a musical score be drawn? How can the musical realization of a performer be described and put in relation to printed performance instructions and to the performances of other musicians? The tools available to musicology for this purpose still leave much to be desired. Music Performance Markup (MPM) is a contribution to enrich the musicological toolbox with a novel description format.

While music notation can be described in a wide variety of symbolic music formats (MusicXML, MEI, Humdrum, ABC), its performance is a complex transformation process that only partly derives from the notation. MPM offers a systematic and comprehensive approach to the phenomena of musical performance (such as timing, dynamics, and articulation). It enables the publication and reuse of research results from musicological performance research and edition for a variety of application and research contexts, such as musicology, timbre research, music psychology, Music Information Retrieval, and music production. In doing so, MPM and its tools also inspire new research designs such as corpus analyses and analysis-by-synthesis methods. The ‘simulation’ or digital ‘reconstruction’ of musical performances allows, for instance, the experimental testing of hearing impressions and hypotheses about how exactly a performance has been made and with which expressive means a certain effect has been achieved.

However, for MPM to become an accepted and practically usable standard, a variety of further efforts are required beyond the mere definition of the format itself. An infrastructure must be created, consisting of software tools (editor, converter, MIDI and audio renderer), guidelines, sample encodings and tutorials as well as development guidelines, which can promote a community-driven further development of the format. In the following, we report on this ongoing effort.
2 Conception

The expressive performance of a piece of music given in symbolic representation – e.g., in common western music notation – is, formally speaking, the entirety of all transformations that are necessary to make the piece sound or to convert it into an equivalent audio signal. This includes the chronological progression of sound events as well as their specific execution. In the course of several years of fundamental research, a series of mathematical models has been developed by which these transformations can be plausibly emulated [1, 4]. These models currently capture the following performance features:

- timing: tempo (incl. discrete and continuous tempo changes such as ritardando and accelerando), rubato, asynchrony, random/unsystematic deviations from exact timing;
- dynamics: macro dynamics (incl. discrete and continuous dynamics changes such as crescendo and decrescendo), metrical accentuations, random/unsystematic deviations from exact dynamics;
- articulation: with absolute and relative effects on tone duration, loudness, tuning, and timing (e.g., agogic accents) as well as random/unsystematic fluctuations of duration and tuning.

MPM is an XML-based data format and can be regarded as a scripting language for describing musical performances by means of these models. Simply put, it specifies when to apply a performance feature to a musical part. In this respect, MPM is distinct from the generative models that dominate the literature on this subject; for an overview, see [6]. The models in generative systems map a musical input to a specific performance. MPM’s feature models, on the other hand, are purely descriptive. Their input parameters serve the purpose of a user-friendly customization of such features (e.g., the shape of a tempo curve, the effect of an articulation, the stochastic character of dynamics fluctuations). Generative systems may use MPM as a construction kit to render their generated performance plans, though.

Given the inherent structure of MPM’s descriptions and the fact that one and the same piece of music can be performed in many different ways, we opted against expanding an already existing, structurally incompatible data standard (like MEI) but to conceive MPM as a format to be used in tandem with other symbolic music formats (incl. MEI) [2, 3]. Thus, those symbolic music formats constitute the notes to be played while MPM describes how they are played.

3 Schema Definition and Documentation

At the beginning of the process, we executed an analysis of the application scenario, i.e., the musicological work with markup formats for music encoding and description. MPM is an XML-based data format. Working with XML editors, most notably the Oxygen XML Editor, is common practice. For special tasks, more comfortable editors with optimized input masks or graphical user interfaces can also be found, such as the metadata editor MerMEId [9, 10] and the Vertaktoid tool for measure annotation [11]. Nevertheless, and especially for advanced users, the XML editor remains the primary tool. For productive work, two editor functions are of essential importance: automatic code completion and (live) validation. Both require a fully comprehensive format definition in a suitable schema language.

The MPM schema definition is therefore of essential importance for productive work in the XML editor. The schema was created by means of the TEI ODD meta language, which can be compiled into a number of different other schema languages, such as RNG, XSD and DTD, so that maximum compatibility with the commonly used XML parsers is provided. The MPM schema is published in a GitHub repository1 under the BSD-2 and CC-BY-4.0 open source licenses. The precompiled release assets include an RNG and XSD compilation.

In its current version 2.1.0, the MPM schema is fully specified. Beyond the purely syntactic definition, it also implements several Schematron rules for content-based tests. These are also included in the validation and detect content-related errors, such as non-existent references, violations of value ranges or invalid value combinations. Everything is communicated to the user through appropriate warning and error messages.

The complete textual documentation was also realized directly in ODD and can be exported to various formats, including HTML and PDF. The documentation includes all elements and attributes, always with compact

---


© Published in 2022 with Humanities Commons under a CC BY-NC-ND 4.0 (https://creativecommons.org/licenses/by-nc-nd/4.0/) license.
code examples, furthermore an extensive guidelines text, which serves as a practical handout and introduction to MPM. The latter introduces the underlying concepts, mathematical models and their representation in XML, explains the structure of MPM documents and supplements this with more detailed code examples.

![Figure 1: A screenshot of the MPM web page in the documentation area. All elements, model and attribute classes are listed and documented here. This also includes short exemplary code snippets. A filter function makes it easier to find entries quickly.](image)

In addition to the transformation from ODD to HTML provided by the TEI OxGarage and XSLT, a custom transformer (doc_generator) was developed. The web page\(^2\) created with it includes both, documentation and guidelines, and marks the best entry point for anyone willing to work with MPM (see Figure 1). The doc_generator is published in the MPM repository along with the MPM schema and is part of the continuous integration pipeline. Thus, in the future it will suffice to upload schema updates in the ODD source code; the generation of the web page and all release assets is fully automated.

Furthermore, the repository contains sample encodings currently for three pieces of music:

- monophonic piece: Johann Sebastian Bach, Minuet No. 2 from Cello-Suite, BWV 1007 (one performance);
- homophonic piece: Melchior Vulpius, four-part chorale harmonization of „Die helle Sonn’ leucht’ jetzt herfür“ (three contrasting performances);
- polyphonic piece: Georg Philipp Telemann, Grave from TWV 51-D7 (three contrasting performances).

Each encoding includes the raw MIDI data (resp. symbolic music without performance), the MPM file with its performances, a PDF file of the score with performance annotations, and an audio rendering of each performance.

4 Software Infrastructure

Working with MPM should not be limited to the comparatively inefficient and unintuitive editing of XML code in XML or text editors. More convenient software tools are developed. Also, these tools should not have to implement the basic functionalities for parsing, creating, editing, and storing MPM data each time anew. It is the task of the Application Programming Interface (API) to provide such a reusable interface for application development.

4.1 Application Programming Interface and Performance Rendering Engine

In parallel to the schema definition, an API for the MPM format was developed. The MPM API was implemented as part of the Java library meico\(^3\). meico is already a well-developed conversion tool for several music formats (including MEI) and provides some functionality that usefully interacts with the MPM API. meico’s MEI-to-MSM (Musical Sequence Markup) export has been extended to export MPM data from MEI encoded performance instructions as well.

A fully featured performance rendering engine has been integrated into the MPM API. This allows the performances described in MPM to be applied to score material in MSM format to create expressive MIDI sequences. MSM is designed as a partner format to MPM with a similar structure and syntax. Originally, it served as an intermediate format in meico and can currently be generated from both MEI and MIDI data. The expressive MIDI sequences can then be saved as MIDI files or converted to audio (WAV, MP3).

![Figure 2: Screenshot of the meicoApp showing the integration of MPM. From the MEI data (left), MSM (top, dark blue) and MPM (bottom, light blue) data are exported. The MPM here contains only one performance, i.e., “MEI export performance” (bottom), that is activated. This allows the MSM data to be rendered “to Expressive MIDI” (top, yellow) and the MIDI data to be played directly and converted to audio (top right). At the bottom right, an external soundfont has been activated, which produces better quality instrumental sounds than the standard timbres provided by the Java Runtime Environment or the operating system.](image)

However, *meico* is not just a programming library, but also provides two application programs in the form of the *meicoApp* package. The command line application can be integrated into XML editors or a scripting environment (e.g., in Python programs) as an external invocation. The graphical application serves as a standalone tool and offers more interaction possibilities as well as more diverse conversion paths in a convenient graphical user interface. *meico*’s main application scenarios so far have included proof listening as part of the music editing process and the creation of sound samples for the music encoded. The MPM API has been integrated into both *meicoApp* applications (see Figure 2), so that users can now easily listen to performances and save them as MIDI or audio. At the same time, this fulfills a wish expressed several times, i.e., to translate not only the notes but also the performance instructions encoded in MEI into MIDI and audio.

### 4.2 Graphical Editor: MPM Toolbox

Furthermore, a graphical editor called MPM Toolbox is developed. Its user interface design was led by basically three major use cases:

1. the free creation of performances (creative use case);
2. the analysis of performance scores and interpretation or annotation of signs in the autograph (analytical use case); and
3. the analysis of audio recordings.

The graphical user interface (GUI) depicts four main widgets. An MSM tree representation of the plain note information is required to match time indications and voice assignments (e.g., to assign articulation X to note Y), but does not need to be editable. An MPM tree representation serves as a fully functional editor.

![Figure 3: MPM Toolbox’s main window. The colored overlays in the score (center) are references to notes from the MSM (green) and performance instructions from the MPM (light blue). Performances can be created either directly in the MPM tree (right) or in the score image.](https://github.com/axelberndt/MPM-Toolbox (accessed January 12, 2022).)
A score display is used to incorporate autographs, e.g., of performance scores, or any other instance of scanned sheet music. Notes from the MSM and performance instructions from the MPM can be assigned to pixel positions in the score, creating a rich, interactive overlay. It is also possible to create new performance instructions here. This makes the score widget the most convenient and, hence, primary interaction widget in MPM Toolbox's GUI. The fourth widget holds the playback controls. This is not only a means to render a performance description into MIDI and play it back with only one mouse click. Moreover, it enables the playback of an audio file alone or in parallel with the rendering of a performance, facilitating listening analyses of recorded performances, i.e., an analysis-by-synthesis process starting with a rough performance and iteratively refining it in order to match performance rendering and audio recording. A screenshot of the MPM Toolbox's main window is shown in Figure 3.

References to notes from the MSM and performance instructions from the MPM are displayed in the score as colored overlays. They constitute annotations that interpret the notated symbols and assign semantics to them. However, it is not practical to visualize all the information of the performance instructions directly in the score, since it is usually printed rather compactly anyway and the remaining space is important for readability. For this reason, the overlays created with MPM Toolbox are implemented as semi-transparent and compact symbols. The detailed specifications of the performance instructions are done in separate editor dialogs. Each of these dialogs not only provides the respective input options, but also actively ensures that the input is valid, i.e., validates against the MPM schema. Several further convenience functions and auxiliary visualizations support the user in this process. Examples of such editor dialogs can be seen in Figure 4.

Figure 4: Four exemplary editor dialogs in MPM Toolbox. These give detailed access to all attributes of the performance instructions or to the parameters of the underlying mathematical models. These individual customizations allow for differentiated analyses and descriptions of how certain instructions were realized by a performer.
5 Things to Come and Final Remarks

The development of the MPM Toolbox continues. There are many major and minor enhancements and convenience features that will be added in future updates. Beyond the mere listening analysis, further tools for audio analysis, possibly interacting with existing tools such as the Sonic Visualiser [7, 8], will be added throughout 2021 and 2022. We also think about a suggestion to incorporate a tool for analyzing MIDI-recordings of performances. The listening analysis tool (parallel audio and rendering playback) will be expanded, too. It should be possible to play the generated performances with custom soundfonts and to export them as MIDI and audio for further treatment, for instance in music production software / Digital Audio Workstations. We also want to provide the MPM Toolbox with documentation as well as tutorial (videos) and best practice tips.

Furthermore, we plan to add further sample encodings to the MPM repository. Currently, they do not include MEI-encoded music such as those from the MEI sample encodings collection. We are open to suggestions and contributions from the community.

We offer practical workshops to provide a more interactive introduction to MPM, its concepts, productive usage and tools. A first such workshop was already held at Edirom Summer School 2020 in Paderborn, Germany. While that event was mainly focused on the MPM syntax and working in the XML editor and meico, future workshops will build upon MPM Toolbox and its more efficient workflow.

Such workshops and user feedback will help us to further develop and improve MPM and its tools. Anyone interested in actively participating and contributing to this process is welcome. The repository includes a contribution guide and a Wiki where we collect suggestions for future additions to MPM's set of performance features.

Acknowledgments

The author likes to express thanks to Tilo Hähnel for the collaboration in the mathematical model development, Benjamin W. Bohl for his contributions to MPM's conception, Peter Stadler for his ODD support and Simon Waloschek for developing the doc_generator. The development of MPM and its tools is kindly funded by the Fritz Thyssen Foundation.

Works Cited

Tools and Perspectives for a Digital Critical Edition of Fourteenth-Century Polyphonic Music

Chiara Martignano  
University of Florence, Italy  
chiara.martignano@unifi.it

Antonio Calvia  
University of Pavia, Italy  
antonio.calvia@unipv.it

Michele Epifani  
University of Pavia, Italy  
michele.epifani@unipv.it

Abstract

The ERC funded project European Ars Nova aims to study the corpus of poetry in Latin, Italian and French set to music by the polyphonists of the so-called Ars Nova. Since one of the main research goals of the project is the comparative study of musical and poetic texts, we are currently developing a web application that will allow readers to visualize and interact with the TEI- and MEI-encoded editions of our corpus together. The adoption of MEI as the underlying format for the digital editions of the musical texts presented us with the challenge of designing an editing workflow that allowed us to critically edit the texts in a user-friendly software like Finale. In this article, we illustrate how the critical editions are transposed to MEI documents using an ad hoc tool developed within the project, and how they are visualized in the web application. Finally, we discuss the critical aspects of the workflow and possible next steps for our digital critical edition in relation to the state of the art of music encoding.

Introduction

The ERC funded project European Ars Nova² aims to study, through an interdisciplinary and comparative approach, the corpus of poetry in Latin, Italian and French set to music by the polyphonists of the so-called Ars Nova.

The project team is currently implementing the ArsNova Database, which intends to increase the basic resources and research instruments in this particular field of study. The database is also working as a VRE (virtual research environment) for the researchers on our team to share knowledge and ensure the reliability of their research results.

The ArsNova Database is hosted by MIRABILE, Digital Archives for Medieval Culture,³ and consists of three sections:

- Catalogue of Ars Nova Manuscripts, Authors and Texts (CANT);
- Corpus of Poetic and Musical Texts (ANT);
- Repertory of Metrical and Musical Structures (ANS).

One of the goals of the European Ars Nova project is to offer online critical editions (of both poetic and musical texts) that may be used by the scientific community as a reference point. Since the comparative study of mu-

---

1 The research presented here is an integral part of the Advanced Grant project “European Ars Nova. Multilingual Poetry and Polyphonic Song in the Late Middle Ages”. This project has received funding from the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation programme (grant agreement No. 786379). The work is the result of the collaboration between the three authors; specifically, §2 (except for §2.1) and §4 are contributed by Chiara Martignano, the introduction and §2.1 by Michele Epifani, §1 and §3 by Antonio Calvia.


© Published in 2022 with Humanities Commons under a CC BY-NC-ND 4.0 (https://creativecommons.org/licenses/by-nc-nd/4.0/) license.
sical and poetic texts is a central concern of the project, we plan to implement visualizations, through which the end user can read and interact with both the musical and the poetic texts. In the digital edition's graphic interface, musical and poetic texts will be juxtaposed, thus inviting readers to consult them together.

Within the ANT section, we are developing the digital edition of a selected subset of musical texts, starting with the works of Francesco degli Organi (Landini). We decided to base the digital editions of the musical texts on MEI encodings. We designed an editing workflow that allowed us to create the MEI encodings not manually, but programmatically. The goal of the editing workflow is to generate different outputs for both the printed and the digital edition. Setting up the digital edition required the use of external tools and the development of an ad hoc tool to transform the critical editions of the musical texts written in Finale into MEI-encoded documents.

In the next sections, we will specify some musicological and philological properties of the corpus of musical texts that will be part of the digital edition. Then, we will present our editing workflow together with the tools we have developed to: (a) assist the editors in the creation of scores encoded in MEI and (b) visualize the MEI files in a web application. Finally, we will discuss the critical aspects of the editorial workflow and the future perspectives for our digital edition.

1 Philological Aspects

The ANT corpus includes all secular polyphonic works in Italian, French, and Latin from 1309 to 1417 (about 1200 texts), excluding the works of Guillaume de Machaut, for which a digital edition is currently in progress. The texts, in comparison with other repertories, have a small manuscript tradition, mostly consisting of less than ten witnesses and lacking any autograph. The critical edition consists of a critical text (CrTe) written in common Western music notation (CMN) and an apparatus that shows the alternative readings (and the errors) found in the manuscripts and documents the editorial work.

A selection of the corpus was chosen to prepare a sample of the digital critical edition. The selected sample includes Landini's two-voice and three-voice ballatas, a homogeneous repertoire large enough (over 140 pieces) to offer a sufficiently variegated spectrum of manuscript transmission possibilities, types of errors, and variant readings.

The CrTe will be offered in CMN, with the addition of the few signs traditionally used by scholars for “translating” some of the details of the fourteenth-century mensural systems, such as the brackets for the ligaturae. The alternative readings will be offered both in CMN and in mensural notation, in two different apparatuses: the first one, a scholar-oriented critical apparatus (S-OCA), will present the alternative readings in mensural notation according to the consolidated editing practices of traditional printed editions; the second one, a performer-oriented critical apparatus (P-OCA), will provide a comparative view of all the alternative readings together with the “lemma” in the form of score extracts in modern notation.

2 The Development of the Digital Edition

Before starting the implementation of the ANT digital edition, we conducted a study on the state of the art in digital music editions, in order to identify best practices and the tools currently available.

---

4 The chronological span of the corpus and the methodologies of the project are discussed in [7].
6 Most of the Ars Nova repertoire's main collections are available online in high-quality digitizations or published in facsimile editions. See, for example, DIAMM (https://www.diamm.ac.uk) and the volumes published in the series Ars Nova (LIM). Access to digitizations and facsimiles reduces the need for diplomatic transcriptions, which should be limited to particular cases (e.g., the case of the difficult-to-read palimpsested folios of the San Lorenzo Codex [5]).
7 Our critical edition aims to restore as much as possible the meaning (and not the graphical appearance) of music written in mensural notation, and to this end it favours translation, a complex operation based on reading and understanding the source system.
8 In the present article, the term ‘lemma’ refers to critical reading in accordance with the definition of the <lem> element within the MEI guidelines (https://music-encoding.org/guidelines/v4/elements/lem.html, accessed January 12, 2022).
We decided to adopt MEI for three reasons:

1. the possibility of scholarly editing with MEI thanks to the MEI.critapp module\(^9\) to create critical editions of the musical texts;
2. the availability of a tool like Verovio\(^{10}\) [8] that allows to easily create dynamic and interactive scores from MEI files on a web page;
3. the use of MEI as a legacy format to preserve our digital editions in a non-proprietary format on the long term.

Once we decided to adopt MEI, we needed to find a solution that would help us encode the editions of the musical texts while reducing as much as possible the manual intervention in the encodings. The team that edits the musical texts consists of two musicologists and only one digital humanist. For this reason, it was vital to find a solution that allowed the musicologists to work autonomously on the musical editions.

We discarded MEI editors like MEISE, the DARIAH-DE MEI Score Editor web service,\(^{11}\) and Meix.js\(^{12}\) because they require the user to edit the MEI files directly in the XML code.\(^{13}\)

Once the option of using an MEI editor was discarded, we searched for a solution to produce the editions entirely with a notation software and then convert the files to the MEI format. By using Sibelius, together with the SibMEI plugin\(^{14}\), it would have been possible to export the files directly to MEI. However, most of the musical texts of our corpus had already been transcribed in Finale.\(^{15}\)

We looked for a plugin that allowed us to edit the alternative readings of the musical texts in Finale. The CriticalEd [6], a tool developed at the Danish Centre for Music, does not seem to have been further developed in order to support Finale nor is it available to a general public.

2.1 The Philological Editing in Finale

As stated before, we opted for a two-fold critical apparatus, scholar- and performer-oriented. While the S-OCA consists of diplomatic transcriptions of the readings of a given witness, like in a traditional printed edition, the P-OCA will offer such readings translated into modern notation, allowing a friendly tool for performers with scarce familiarity with black mensural notation. We decided to transcribe the critical text in score format, according to the number of voices, adding a staff for each voice of each witness, including the ‘manuscrit de surface’ chosen as a reference point for formal and graphic readings.\(^{16}\) Obviously, it is not necessary to transcribe each witness completely, but only when the readings diverge from the critical text.

In both apparatuses, we recorded the variant readings with respect to the following aspects: text underlay, pitch (including accidentals), durations, notation (mensuration signs, plicae, ligatures, color, rhythmic grouping, use of particular note-shapes, etc.). Therefore, the Finale document displays a synoptic edition, which records, along with the critical text, all the variant readings transmitted by the witnesses, including the ‘manuscrit de surface’, where it contains an error or a variant reading that the editor considers ‘deterior’.\(^{17}\)

\(^{9}\) [https://music-encoding.org/guidelines/v4/content/scholarlyediting.html#critApp](https://music-encoding.org/guidelines/v4/content/scholarlyediting.html#critApp) (accessed January 12, 2022).
\(^{10}\) [https://www.verovio.org](https://www.verovio.org) (accessed January 12, 2022).
\(^{13}\) Although part of the critical apparatus, the S-OCA, as we will discuss below, provides a diplomatic transcription of the variant readings, the editor created by the Measuring Polyphony project [3] cannot be applied to our project for obvious reasons, since the material to be encoded belongs primarily to two radically different categories: on the one hand, items written in mensural notation; on the other hand, critical editions written in CMN.
\(^{14}\) GitHub repository of the plugin: [https://github.com/music-encoding/sibmei](https://github.com/music-encoding/sibmei) (accessed January 12, 2022).
\(^{15}\) In any case, although the SibMEI plug-in saves a step in the conversion process in comparison to Finale, it introduces roughly the same issues that we shall discuss later.
\(^{16}\) For the notion of ‘manuscrit de surface’, see [2]. The choice of the ‘manuscrit de surface’ depends on various factors, among them the completeness of poetic and music texts, the importance of the manuscript within the tradition of a given composer, the correctness of the readings. In the case of Francesco Landini, we chose (whenever possible) the ms. Firenze, BNC, Panciatichi 26, which is the earliest and one of the most accurate witnesses for Landini’s manuscript tradition.
\(^{17}\) It should be noted that something could be lost in the process of transcribing, since in many cases music notation allows more than one way to graphically represent the same meaning. The difference between two distinct notational systems in a composition (say, the post-Marchettan notation and the [longanotation]) is virtually just a matter of graphic representation, but irrelevant for the substance of the musical text.
In the case of a two-voice ballata transmitted in a single witness, for example, we will have two staves for the CrTe and two staves for the transcription of the witness (WiTr), both in CMN. In this simple case, the WiTr will be different from the CrTe only for errors and other editorial elements, such as the editorial accidentals. Editing a three-voice composition preserved in three manuscripts requires twelve staves (three for the CrTe, and nine for each WiTr). In other words, we have two sets of staves, one devoted to the CrTe, and a set of ‘service’ staves where the editor recorded the varia lectio, each labeled with the combination of the manuscript siglum and the voice part (for example, SqCt = Squarcialupi Codex, Contratenor part). As an example, Figure 1 shows the transcription of the three-voice ballata “Gentil aspetto, in cu’ la mente mia” by Francesco Landini.

Figure 1: Transcriptions in Finale of the critical text and the witnesses of Francesco Landini’s “Gentil aspetto”, mm. 1–10.

In this case, one of the witnesses (Pit) needs a separate edition, since the Tenor part (staff “Pit T”) bears the text, thus configuring an alternative version of the work. This type of variant reading, which might be considered an extreme case of text underlay variance, necessarily involves the vocal/instrumental (or vocal/vocalized) dichotomy; in all likelihood, it originated from medieval performance practice and affects the modern performance too.18

To sum up, the CrTe is entirely transcribed in the Finale document, while each WiTr is written only where it differentiates from the CrTe, thus producing one complete CrTe and as many “negative” WiTrs as the number

---

18 In similar cases, the possibility to identify one of the versions as ‘correct’ or ‘better’ than the others is scarce. If, as in the majority of the cases, there is no way to determine what the author wanted, there will be no way to determine whether the text is missing (by accident or deliberately) or was added later. On the other hand, each version is essential for the history of the reception of the work.
of the witnesses. After the ‘synoptic edition’ is completed, each of the ‘service’ staves – corresponding to the WiTrs – and the CrTe are exported as individual MusicXML files using the “Extract parts” tool in Finale. The MusicXML files are then converted to MEI with the MusicXML Converter tool of Verovio. Finally, all the MEI files are merged into one MEI document with the tool ANTCollator developed within the European Ars Nova project. The final result of the workflow is a MEI-encoded critical edition, ready to be visualized on a web page. Before presenting how the editions are visualized, we will introduce the ANTCollator software.

2.2 The Merging Process with the ANTCollator Software

Since the result of the philological collation carried out by the critical editor is a Finale file containing the CrTe and all the “negative” WiTrs in separate staves, we needed a software to reassemble all this information in the form of a MEI-encoded critical edition.

The ANTCollator is an application, developed with the Javascript framework Angular, that compares a main MEI file containing the CrTe, used as the “base text” of the collation, with multiple MEI files containing the WiTrs.

By default, the name of a WiTr file is used as its siglum. For example, if the text of the witness Ab is stored in the file “Ab.mei”, the software will extract “Ab” from the file name and use it as identifier of the WiTr. The user can manually change the siglum of each witness.

The collation parser is, for the moment, rather basic: it compares the CrTe with a WiTr at a time and measure by measure. When the measure of the CrTe and the measure of the WiTr differ, the measure of the WiTr and the siglum of the witness are stored in a JSON object. This JSON object models the critical apparatus and all the other main information about the critical edition. Then, all the readings registered in the model are embedded as MEI elements in the output file. The output file is a duplicate of the base file that features elements in correspondence of the measures subject to variation. Inside the elements, the “lemma”, which is the measure of the CrTe, marked up as elements, are nested. The software outputs the merged MEI file and the JSON model.

The elements derived from the automatic comparison are the base for the P-OCA. We decided to implement also the S-OCA of each edition inside of the respective collated MEI file. In this way, one MEI file corresponds to the complete digital edition of a musical text. In order to facilitate the encoding of the critical entries of this apparatus, a dedicated module of the ANTCollator software was developed. Thanks to this module, the editors can add, edit, and remove critical entries from the apparatus. Each entry must be anchored to a measure by its number, so that on the web page the readers will be able to go from the score to the apparatus and vice versa. The text of the entries can be edited in a text area input with the help of a SMuFL characters keyboard to reproduce the alternative readings in mensural notation. The S-OCA and its entries are dynamically stored in the JSON model and also added to the output MEI file as MEI elements. The apparatus is converted into elements, which is inserted into a generic appended to the element. The entries are transformed into elements, which contain their texts and the SMuFL characters as elements. The number of the measure is stored in the attribute of the element (see Listing 1).

---

20 The software was named “collator” because, although the underlying collating algorithm is for the moment rather basic, it can be used to automatically compare a base text with multiple WiTrs encoded in MEI. In our editorial workflow, we use the ANTCollator not as an automatic collator but as a post-collation tool to merge the MEI files of the CrTe and of the WiTrs in one MEI file, to edit the S-OCA and the file metadata, and to create a JSON model that will be used in the ANTViewer.
22 There are other tools available to carry out the automatic comparison of different MEI files, for example MusicDiff [4] (https://music-diff.edirom.de/, accessed January 12, 2022); we decided, however, not to use MusicDiff and to develop from scratch a tool that could perform both a simple file comparison and the MEI encoding of custom data.
Because the S-OCA is meant to be visualized as plain text, like in a printed edition, we opted for a light-weight markup solution using SMuFL characters.

```html
<div>
  <list>
    <head>Scholar-Oriented Critical Apparatus Entries</head>
    <li n="MeasureNumber">
      <!-- if necessary, a secondary list for each single variant -->
      <symbol glyph.num="U+E958"></symbol>
      <!-- SMuFL character for the mensural notation readings -->
      <!-- Witness Siglum -->
    </li>
  </list>
</div>
```

Listing 1: A sample encoding of the scholar-oriented critical apparatus (S-OCA).

Finally, the ANTCollator software is constituted also by a module that helps encoding the metadata of the MEI file, like the title of the musical text and its author, the names of the critical editors, etc. The data inserted by the editors are then converted into MEI elements and added to the `<meiHead>`.

2.3 The Visualization of the Digital Editions

The digital editions are visualized in a web application called ANTViewer, developed in Angular. The digital editions will be integrated in the same digital archive of medieval culture, “Mirabile”, that hosts the catalogue (CANT) and the metric repertory (ANS) of our project.

The editions of the musical texts are available inside the GUI both individually and in a comparative view together with the poetic texts.

![Figure 3: Comparative view of the musical and poetic texts in the ANTViewer.](image)

The scores of the editions are rendered on the web page with Verovio. Thanks to the Verovio javascript toolkit, it is possible to interact with the score: browsing by page, changing the zoom level, and handling the layout and the score breaks. In some digital edition projects, like Measuring Polyphony and Tasso in Music\(^2\), it is possible to switch the score from modern to mensural notation. We decided not to implement this feature in our

edition, but to present the critical texts in modern notation only. However, as mentioned above, readers are offered two different apparatuses (S-OCA and P-OCA).

![Image](image1)

**Figure 4:** Rendering of the S-OCA in the ANTViewer.

The S-OCA is always visible in a panel below the critical text score. This panel displays all the critical entries together, allowing the user to scroll through them and get an overview of the variation of the text in the tradition. Each entry is preceded by the number of the measure it refers to. By clicking on the number, the corresponding measure is loaded into the score, if not yet visible, and highlighted.

In the score, the presence of a *locus criticus* for a given measure is indicated by an icon placed above the measure that displays the measure number. The icons of the *loci critici* trigger the opening of the two apparatuses depending on the option selected by the user. The option is set by default to “traditional mode”: When the user clicks on an icon, the S-OCA panel is opened and scrolled to the corresponding critical entry for the measure. When the option is set to “collation mode”, the P-OCA is loaded as a new panel on top of the critical text. In that apparatus, the reader can find in a comparative view the critical text and the witnesses. All the score extracts are given in modern notation.

![Image](image2)

**Figure 5:** Rendering of the P-OCA in the ANTViewer.
3 Problems and Critical Aspects

The critical aspects of the workflow outlined above are manifold. As mentioned before, the texts in our corpus have few witnesses, therefore we can afford to create the files and edit the witnesses for each single edition. For an edition based on a larger number of witnesses, the editing phase with Finale would not be sustainable.

Writing with a user-friendly graphical interface, such as that of Finale, Sibelius, or other music writing softwares, remains an essential asset for the critical editing of music. At the same time, encoding in MEI provides clear advantages, above all in terms of interoperability and long-term preservation.

Some of the problems of the editorial workflow that still need to be solved are purely graphic. The conversion of edition files between different formats (from Finale’s proprietary format to MusicXML, from MusicXML to MEI, and from MEI to SVG) results in considerable loss of graphic information. The difference between the score visualized in the web application with Verovio and the score edited in Finale is quite significant.

We offer here a few examples of the graphical problems: 1) brackets to represent ligaturae; 2) editorial accidentals; 3) encoding of synaloepha and aphaeresis; 4) nonstandard notational elements involved in the translation from mensural to current notational systems (double-dotted half notes and conflicting and/or nonstandard key signatures).

The use of brackets to represent the ligaturae of mensural notation is widespread in editorial practice. However, following the procedure described above for the conversion from a Finale file to an MEI file, the brackets are not encoded (Figure 6):

```
<measure n="3" xml:id="w29756ab1c17">
  <staff n="1">
    <layer n="1">
      <note dur="8" [11 lines]
      <note dur="4" [4 lines]
      <note dur="4" [11 lines]
      <note dur="8" [4 lines]
    </layer>
  </staff>
  <staff n="2">
    <layer n="4">
      <note dur="4" [11 lines]
      <note dur="4" [11 lines]
      <note dur="4" [4 lines]
    </layer>
  </staff>
</!--Bracket not transcoded.-->
```

Figure 6: Francesco Landini, “Abbonda di virtù”, MEI encoding of m. 3 compared to the rendering of the Finale file.

Entering ligaturae as a slur in Finale, however, allows us to encode in MEI the position of the beginning and end of the graphic sign, to which we then apply a different rendering.

```
<measure n="3" xml:id="w29756ab1c17">
  <staff n="1">
    <layer n="1">
      <note dur="8" [11 lines]
      <note dur="4" [4 lines]
      <note dur="4" [11 lines]
      <note dur="8" [4 lines]
    </layer>
  </staff>
</!--Bracket not transcoded.-->
```

Figure 7: Rendering of the ligaturae in Verovio.

The issue of editorial accidentals is addressed in the MEI Guidelines from the point of view of semantic mark-up. Our project would need a semantic marker “editorial” for an accidental to be associated with a particular
graphical rendering that allows one to recognize an “editorial” accidental at first glance. For example, the most common choice in paper editions, as is well known, is to place the accidental above the note. However, in a digital edition other graphic options can be found (colour, size, font).

We have encountered some problems in encoding the synaloepha in the transition from Finale, where a “hard space” is typically used, to MEI. In such cases we intervene directly with the Oxygen XML Editor\footnote{http://www.oxygenxml.com (accessed January 12, 2022).} to modify the affected syllable.

The problems concerning the rendering of the critical text and of the critical apparatus are the same. Generally, the results that can be obtained by using Finale in a ‘crafty’ way do not work in a different environment. A single example will suffice: “Gentil aspetto” is in \textit{tempus perfectum maius}, and one may want to use the double-dotted half note introduced by Willi Apel \cite{Apel} in order to transcribe a single brevis with a single note worth nine eighths. Such a note value does not exist in standard music notation as a single note-shape, hence the nonstandard double dot (see, for instance, the first two bars of Figure 8).\footnote{In this case, the double augmentation dot acquires a different meaning than in standard modern notation, which is why Apel places the dots on top of each other [:], and not in the standard way [.].}

Another issue concerns key signatures. In editions of fourteenth-century music, it may be necessary to represent both conflicting signatures (for example, the Tenor part with b-flat, the upper parts with no accidentals) and nonstandard placement (typically, one may want to place the b-flat in the bass clef an octave higher than usual). “Gentil aspetto” presents both issues: In one of the witnesses (Pit), the B section of the Tenor part alone contains two flats, b and e, with b an octave higher.

\begin{figure}[h]
\centering
\includegraphics[width=0.5\textwidth]{figure8.png}
\caption{Francesco Landini, “Gentil aspetto” (alternative version), CrTe, mm. 32–34.}
\end{figure}

Unfortunately, to achieve this result with Finale, it is necessary to configure the key signature not only as an independent element of the staff but also as a nonstandard signature to display the b-flat one octave higher than usual in the bass clef. The uncommon placement of the key accidentals is not recorded in the MusicXML file,\footnote{Even when importing the generated MusicXML file with Finale itself, the signature accidentals appear in the default octave setting.} while in the MEI-transcoded file the change of key signature is entirely neglected. This problem arises even when a standard key signature is assigned as an independent element of the staff.\footnote{It seems that the key signature of the first staff invariably affects all the others; putting a b-flat on the Cantus part, no flats in the Contratenor part, and two flats on the Tenor staff, the conversion from MusicXML to MEI generates a file with a b-flat on all staves.}
4 Conclusions and Future Perspectives

The workflow described has different limitations concerning the final rendering of the scores on the web page and the ability of transcoding what one sees in the Finale interface into appropriate MEI encodings. The current workflow requires some tweaks when editing with Finale and a manual review of the MEI files. An “MEI-centric” approach, i.e., a workflow that starts from the MEI-encoding of the editions, would likely have resulted in the creation of only one MEI file that could have been used, without further intervention, as a starting point for both the digital and printed editions. In this case, the online and printed versions of the scores might have been visually more consistent.

The limitations of our workflow are representative of two problems with digital editing of music. On the one hand, the lack of WYSIWYG29 tools for encoding in MEI. The currently available editing tools we know require music encoding skills. On the other hand, editors need significant computer science skills, in order to have a higher control over the rendering of the edition on the web page.

In the broader context of the focus of our research project, the implementation of the digital critical edition of the ANT represents an auxiliary task for the moment and is meant to test our corpus with the possibilities offered by the digital medium. The possible next steps for our digital critical edition, once we have refined the editing workflow, are: 1) extending the subset of our corpus that will be available in the digital edition; 2) grant open access to the MEI encodings in the long term by storing them in an online repository. Our hope for the future is to be able to encode the whole ANT corpus in MEI and preserve it in an interoperable format so that it may be used for further innovative studies.

Works Cited


---

29 Acronym for “What you see is what you get”, used to describe editing software that allows content to be edited in a form that resembles its final appearance when printed or displayed on a web page.
Encoding Traditional Spanish Music for Pedagogical Purposes Through MEI: Challenges and Opportunities

Sara González Gutiérrez     Javier Merchán Sánchez-Jara
Universidad de Salamanca, Spain   Universidad de Salamanca, Spain
saragnzlz@usal.es     javiermerchan@usal.es

Maria Navarro Cáceres
Universidad de Salamanca, Spain
maria90@usal.es

Abstract
This paper aims to highlight, as a case study, the encoding of a Spanish traditional music corpus using the MEI standard for the development of an interactive traditional music database focused on preserving and disseminating this type of cultural expression in the field of music education as well as ethnomusicology research. It analyzes the possibilities of the schemas and the guidelines followed to describe canciones at the bibliographic level, as an archetypal form of compiling traditional music in Spain, as well as the analytical aspects that highlight paradigmatic elements of this kind of music such as rhythmic patterns, tessituras, modal contexts, phrases, structural forms, etc.

Introduction
In the last decades, there has been a marked evolution towards the digital treatment of information collected on different media. Although most of the information discussed until relatively recently referred to textual information, currently, the analysis of artistic content such as music is gaining importance with works like [1, 3, 4]. There are several proposals, including MIDI, MEI, or similar approaches, that have tried to standardize the information encoded in a digital file. In this way, groups from different regions or even countries can understand the properties of the music encoded in those files.

However, initiatives have always focused on the generation of music belonging to the Western classical (and tonal) tradition and, for the best of our knowledge, there has been no relevant study that addresses the generation of oral traditional music and has involved the use of the idiosyncrasies of this kind of music.

Traditional music has a great importance in society, as it is part of our culture. Nevertheless, most of the repertoire of different European popular songs is almost unknown to the general public. Ethnomusicology, the science that studies folk music, has tried to collect and preserve all this valuable repertoire. However, this task has become more difficult due to the social changes in Europe, such as the depopulation process of rural areas as a result of population migration to urban centers, or the introduction of machines and automatic tools in traditional crafts.

Taking advantage of the digitization of society and the new forms of preserving information that they entail, this work is proposed to give a solution to this loss of our popular music (and therefore our cultural heritage) and to become available to any ethnomusicologist or anyone interested in folk music. The proposal aims to create the format of the Music Encoding Initiative (MEI), but with labels adapted to capture the oral traditional music features. MEI will allow our model to be understandable to research groups from different regions or even countries.
As a first test, we have compiled a corpus sample with a total of 250 songs (this number has been increased in future research stages) of the Spanish tradition from different sources, like songbooks (cancioneros) or archives. The corpus of songs holds a total of 588 pieces mainly from Spain and Portugal, and to a smaller extent to certain regions of Italy. In detail, most of the songs are from Spain, and have their roots in the the regions of Catalunya, Aragón, Cantabria, Castilla y León, and Andalucía. They were collected, edited, and registered from some of the most relevant ethnomusicologists of the 20th century. The most collected typologies were the nursery rhyme (canción de cuna), among which we must highlight the circle song, and the lullaby (nana).

Each song was encoded in the MEI format in a separate file, while we adapted and created new labels to show the musical features that characterize this kind of musical content. All the files were organized to create a dataset that is available for any research group interested in Spanish folk tradition or music encoding.

The paper is structured as follows: Section 1 presents the bibliographic information that has been included in our database. Section 2 will show the particular features of Spanish folk music and how they were encoded in the MEI files. Finally, conclusions will show the results of our proposal and outline future work.

1 Encoding of Bibliographic Information in the Repertoire of Traditional Music in Spain

1.1 Characteristics of Spanish Traditional Music

Spanish traditional music encompasses the entire repertoire of songs and instrumental pieces that have been preserved until now through oral transmission, mostly in rural areas, although it has also occurred in an urban context.

In order to understand how folk music and classical music can be related, ethnomusicologists have been investigating the evolutionary origins of traditional music. This is important for us to evaluate if some features are shared with classical music, and therefore, can be encoded under the same guidelines.

A comparative analysis of different repertoires of this geographical area with written classical music reveals numerous fundamental similarities, but also a series of notable differences between these two musical paradigms. The similarities become apparent when comparing traditional repertoires with the first author’s music, transcribed in a still incipient writing system invented for Gregorian chant and also used for the songs of troubadours, minstrels, Goliards, and traveling musicians. There are very close similarities in the ways of organizing sounds to create melodies, in the structures of melodic development conditioned by poetic texts, in some rhythmic structures and patterns, in the alternation of sung solo subsequently repeated by the listeners with short phrases and repetitions, and in the dynamics that results from all these elements together.

From the compilations of Spanish traditional music that have been made from the 19th century, some conclusions have been drawn as a result of the study and in-depth analysis of its historical, textual, and musical elements. We can consider that modal systems that appear repeatedly in songbooks (known as cancioneros) may be medieval survivals, although their sonority is often very close to paradigmatic sonorities from tonal contexts. On the other hand, textual content is organized mostly in octa-syllabic quatrains and short lines that also determine the rhythmic structure of popular pieces. These textual formulas are not very old, going back to the Renaissance period. Although there may be certain elements that can be hypothetically considered as very old, these are undoubtedly mixed with elements developed in modern times. Therefore, we may be facing a certain recent musical culture, just a few centuries old. In other words, several of its constituent elements that appear in popular songbooks are part of a common heritage, however, this heritage has been enriched and diversified with new contributions that give its uniqueness to the popular piece.

The analyses carried out by various authors [2, 6, 8] reveal that although traditional music had certain inspirations in cult music in its beginnings, the truth is that these characteristics remained stagnant and underwent a very different evolution from academic music, from which it received little influence in recent centuries. Therefore, this allows us to affirm that there are a series of elements that appear in traditional music, and they have not been used in classical music for more than 500 years. We highlight the following fundamental elements:
- modal melodic systems and their melodic profiles;
- a narrow range of melodic ambitus (in almost no case more than one octave, usually between the fifth and sixth);
- unstable or chromatic notes that do not follow the rules of academic music;
- harmony totally outside cadential space (or harmony completely non-existent);
- prosodic rhythm or very regular rhythm in the case of dances;
- repeated rhythmical patterns along with the tradition according to the genre and time signature;
- a particular structure of music based on text repetition of choruses and/or stanzas.

For the collection of traditional music, different types of notation had to be used, including Common Western Music Notation, which has certain peculiarities that can be used to capture the musical particularities of such repertoires. As was already the case in the 20th century with scores by some authors such as Stockhausen or Xenakis [5], the particularities of the different notations of each author do not follow any general standard, which can lead to confusion if there is no precise interpretation methodology. Thus, this methodology should be clear to capture the same representation on a computer.

Apart from the musical features, we have to consider the format in which the music was collected. Due to the inherent oral nature of folk music, this music has undergone transformations and changes over time. For instance, from the second half of the 19th century, the traditional repertoire began to transform its sound due to the influence of Flamenco singing and music genres such as Zarzuela, which were particularly well received by the people and were imitated with great frequency, assuming these changes as their own. Additionally, from the middle of the 20th century there was a progressive decline in oral tradition as a consequence of social changes and the invention of technological advances like the radio, which affected the repertoire. Other examples are related to the interpreter’s knowledge and style, or the region in which the music is reproduced.

Unlike classical music, where pieces are written and transmitted through a score that varies along the time with a lesser degree of indetermination in the content itself, the information about the sources in which a folk song is found (the songbook, the date, the region, the interpreter, etc.) is essential to understand the possible patterns, modal scales, or rhythms that appeared. Therefore, this information also has to be incorporated in the MEI files.

This question is of great importance due to the profusion of variants that develop on the melodic or textual level to the same theme or song, depending on the transformations that take place over time or due to the circumstances of the territorial environment.¹

1.2 Development of an MEI Model

As a generic principle, musical pitches and durations are encoded with MEI core elements. Additionally, we incorporated labels to capture the features of folk music both in the header and in the body. In the header, firstly, we added a general file description with labels about the compiler, the encoder, the publication statements, and the editor. The compiler is the person (usually an ethnomusicologist) who transcribed the song during their research work. The encoder is the person that musically analyzed and encoded the song. The rest of the properties are directly related to editorial and publication metadata (year, place, company, etc.).

Secondly, we described the bibliographic information, such as the formal title of the songbook (taken from the published editions), interpreter, informant, region in which the song was collected, etc. The individual song title is usually given by the interpreter (the person that sang or played the song when it was transcribed). If it is unknown, the title will be the first verse of the song or a factual title through textual incipit (first meaningful words). The region information collects the country and province in which the song was interpreted and transcribed or recorded.

Thirdly, we encoded the general music information such as the main scale of the melody (modal or tonal), explicit or suggested tempo, and the incipit for all songs.

Finally, we added some notes about particular transcription decisions if we encountered any unusual issue (for example, if we detected a transcription error in the primal source, or if we omitted some information).

¹ One example of these variations can be found at the following link: https://zenodo.org/record/6481545#.YmWiovNbwIU (accessed January 12, 2022).
In the body, we added the rhythmic pattern (a succession of rhythmical events that is repeated along with the song), the ambitus, the musical structure or form, indication of phrases and cadences, and the lyrics.

We encountered some difficulties deriving especially from current editions: At the bibliographical level, they are related to the intrinsic complexity and idiosyncratic particularities of songbook editions, are related to the different levels and sublevels that these documents have. At the work level, they incorporate divisions related to specific geographical scope, and at the responsibility level, they present different levels of authority and roles. Among these difficulties, the following issues can be highlighted by way of example:

First of all, the publisher is responsible for the edition of different songbooks related to different territories, or for the same song book but with editions in different years. In order to manage each of these volumes in the database, an xml:id is created incorporating this information (editor_year_geographical scope) as could be seen here:

```xml
<source xml:id="GarciaMatos2000CC">

The rest of the bibliographic information of the source is taken from the title page of the work, following the cataloging rules proposed by the Biblioteca Nacional de España (BNE), using the @role attribute for the description, in a literal way, of the functions of each authority. In the same way, the different levels and structural possibilities of the title element are used to reflect in a palaeographic way the information of the title page of this songbook (Listing 1).

```xml
<imprint>
<title>Lírica Popular de la alta Extremadura</title>
<title type="subordinate">folk-lore musical, coreográfico y costumbrista</title>
<title type="desc">436 documentos musicales inéditos</title>
<respStmt>
<persName role="recopilador">Manuel García Matos</persName>
<persName role="edición_introducción_e_índices">Mª Pilar Barrios Manzano</persName>
<persName role="Biografía_bibliografía_y_discografía">Carmen García Matos</persName>
</respStmt>
<publisher>Universidad de Extremadura, Servicio de Publicaciones</publisher>
<pubPlace>Cáceres</pubPlace>
<date>2000</date>
<extent type="pages">320</extent>
</imprint>

Listing 1: Example of bibliographic information encoding.

To identify each song as an individual item that makes up the songbooks, we have opted for the inclusion of two incipits: one at the level of the lyrics of the song and the other at the level of musical notation. The former also functions as a factitious title of the song, since in the absence of formal or standardized titles it is usually known by the first meaningful phrase of its lyrics. Through an xml:id identifier, the relationship between an incipit and its function as a factitious title is established. Furthermore, this identifier makes it possible to relate the incipit, and by extension the song to which it refers, to different versions of these compositions that appear in other songbooks or versions thereof (Listing 2).
In addition, musical information about the work has been included, related to the harmonic mode, metric, and tempo. In this case, the `@type` attribute is set to be indicative based on the knowledge of the encoder of the item since in many cases this type of information does not appear in the source text, although this attribute does not seem to be fully suitable to encode this type of information.

Although there is a specific label for identifying genres, regardless of the particular characteristics of the genre, ascribed to the field of bibliographic description, there is a lack of specific labels for the delimitation of specific musical forms. The solution proposed for this purpose is to categorize the tag used for the descriptor terms by the `@type` attribute.

A double indeterminacy arises in this case; on the one hand, such information cannot be established as an addition to the text at the editorial level (in the sense of the information that is usually included in the `<add>` element), because this type of works is reproduced without this type of indications and, on the other hand, because tempo in these works is always indicative and the degree of variability about its interpretation is unknown.

Musical incipit encoding is carried out using a specific element for this type of information. The proposed encoding scheme (Listing 3) for this goal is functionally optimal and demonstrates the flexibility of the system to make information explicit in an unambiguous way, even though this is an ambivalent label for musical or textual incipit. Likewise, a structured musical incipit is included through the inclusion of the identifiers of the measures that form it; several measures between 2 and 3 that form an autonomous and meaningful musical idea are established as a general rule. On the other hand, each song is categorized employing an identifying label related to its genre within the paradigmatic forms of traditional Spanish music and another related to its geographical origin. The subsequent idea is to elaborate a taxonomy of forms and styles for the analysis of the constitutive elements.
Finally, as an editorial declaration, some comments have been made about the transcription and encoding, particular issues, and editorial decisions. Generally, we have decided to replace quavers and semiquavers single notation (without grouping) with the current notation which should provide an improved and easy-going notation path for musicians and children.

### 2 Encoding of Analytical Music Information for Ethnomusicological Purposes

Apart from encoding the musical texts that make up the corpus of analysis, we have included a series of encodings related to implicit information in the works, made explicit by the encoder, with the aim of highlighting the most common characteristics between works belonging to the same genres or coming from the same geographical areas. The ultimate motivation is to try to identify whether there are common elements at the harmonic, rhythmic, or melodic level between such works, of the same genre or from the same geographical area. First, within the `<scoreDef>` element, the `<ambitus>` tag has been included to collect data about the *tessitura* of the analyzed corpus. The idea is to gather information about one of the defining characteristics of this type of repertoire, namely the presence of melodic material, generally developed in the octave range. The `<ambitus>` element is very conveniently adapted to this purpose and allows us to systematically treat this type of information in a very precise way (Listing 4).
The purpose of including this information is to analyze whether this essential characteristic remains uniform and stable among the majority of the works that make up the songbooks, regardless of particular genres or territorial settings, or whether there are significant variations about these two factors.

The second type of information made explicit through the analytical interpretation of the musical text is that of the rhythmic patterns that structure the work. At this point, it is a matter of identifying and marking the essential and indivisible rhythmic motifs that structure the songs and are repeated in each of them in an archetypal way in each of the songs. In this case, the aim is also to observe whether these structures appear as common elements concerning the development of specific genres, melodic phrases, or geographical areas.

To mark, for example, that the following basic rhythmic structure in Figure 1 is the metrical backbone of a work, we have chosen to include the encoding of the rhythmic formula before the first measure of the work, indicating that in this case this kind of information has been provided by the encoder and that it refers to the indicated rhythmic pattern. Specifically, the problem arises that when the musical text is about to be rendered, the system must be told not to display this type of information should not be displayed, since its only function is to be treated within the musical corpus for analytical or classification purposes on a large scale.

![Figure 1: a) Rhythmic pattern of the song; b) Example of rhythmic pattern encoding.](image)

This is another analytical functionality that needs to be developed through the formulation of categorized models. In this case, the rhythmic structure that makes up the rhythmic pattern generating the musical structure of the piece is encoded by indicating, first, that it is material outside the musical text itself, and second, that it represents the rhythmic pattern characteristic of the work. A specific label to delimit paradigmatic structures at the melodic, rhythmic, or harmonic level that characterizes a given work in some sense can help to make this type of information explicit more simply and effectively.

From a similar perspective, the information related to the form of the musical phrases, previously identified and delimited by the project transcriber, is encoded within a `<supplied>` element (Listing 5).
Listing 5: Example of musical phrases encoding.

In this case, the starting and ending points of each autonomous and meaningful musical idea are identified. Simultaneously, these types of musical phrases are classified by the `@type` attribute to describe the melodic structure of the work on a general level. The most appropriate criterion, in general practice, is to give a different capital letter to each musical idea with its own autonomy and musical semantics.

At the same time, explicit reference is made at `<note>` element level to the type of cadence that occurs at the end of each of these sections by including a `@type` attribute in the final note with the codes R for repose notes and SR for semi-reposing notes:

```xml
<note xml:id="dle320" pname="d" oct="4" dur="4" stem.dir="up" type="SR">
```

Here it was necessary to establish the description of the musical phrase or phrases that structure the piece on a melodic level, with the particularity of identifying the cadential notes (repose) and their particular context quality (conclusive or suspensive). There are no specific mechanisms for this type of musical analysis, and the labels that refer to the musical phrase as a predetermined melodic unit are related to its delimitation on a structural or visual level rather than in the sense of its purely musical or melodic function. In this case, the choice is made to delimit the notes that constitute the starting and endpoints of the musical phrase, transferring the information to the system of the rest of the XML tree between those two points containing the information of the musical phrase. This type of approach allows for a certain level of operability but introduces a certain degree of uncertainty with the events that may occur at the structural level between these two points.

At the lyrics level, Oxygen software (as the main XML editor used for encoding) identifies that each of the notes is associated with one of the syllables of the song's lyrics. For this reason, syllables are included as verses in the `<measure>` encoding. Replacement texts for the repeated verses are included in the `<note>` element (Listing 6).

```xml
<measure n="2" xml:id="dle188">
  <staff n="1">
    <layer n="1">
      <beam>
        <note xml:id="dle190" pname="a" oct="4" dur="8" stem.dir="up">
          <verse n="2">
            <syl x="1.316">pu</syl>
          </verse>
        </note>
      </beam>
    </layer>
  </staff>
</measure>
```

Listing 6: Example of lyrics encoding.
2.1 Construction of the Dataset

Most of the songbooks were available in physical format, so we had to scan the identified songs and transfer each one to a different file. With these files, we applied a commercial Optical Music Recognition tool (SmartScore Pro)² to transform the PDF or image files into MusicXML readable by any music editor. We edited the files to correct possible OMR recognition errors. We had to take into account that some scores were handwritten and therefore OMR recognition was unable to retrieve a perfect digital score. It should also be kept in mind that these are scores of oral tradition which, unlike classical music scores, have many annotations of all kinds in the margins. This reduces the efficiency and speed of this program and the digitization process of the scores.

After correction, we transformed the MusicXML files into MEI files. To do so, we first created a template with the standard MEI fields that must be filled in before converting the MusicXML file. To incorporate the template and edit the tags, we used the Oxygen software.³

In addition, we have worked through each songbook to automate the process and follow consistent criteria. The main criteria for naming and organizing the files are meant to respect the original name of the song. If the score had a title, it has been preserved with the original numbering. In many cases, short songs do not have a title, therefore it has been named by the number assigned by their compiler or by the page number in the songbook.

On the other hand, the data set has been stored in three folders. The first one is destined to each one of the songbooks, and in each of them there is a folder named XML. This folder contains the digitized scores exported in MusicXML format. Inside this folder, we created another subfolder called TIME that contains the intermediate files produced in the conversion of the files from MusicXML to MEI format. Finally, we created the FINAL MEI folder that contains the MEI files edited according to our template.

The whole dataset and the visualization of the data can be accessed online.⁴ The dataset presents two kinds of visualizations: The first one is a common search by parameters, in which the user can get the songs grouped by country, mode or scale, time signature, or even rhythmic patterns. The second visualization presents a cloud of colored dots showing songs that are more similar to each other according to their melodic and/or rhythmic features. To calculate the similarity between songs, we followed the computation of the Structure Induction Algorithm Measure (SIAM) described in [7]. The details of the visualization procedure will be described in a future paper.

Conclusion

The encoding of the pieces of Spanish oral tradition has allowed us to observe that there are no specific mechanisms to mark the musical phrases and their respective cadendial notes (repouse and semi-repose) in this and other types of repertoire. As we discussed earlier, the labels that refer to the musical phrase as a predetermined melodic unit are related to its structural or visual delimitation rather than its purely musical or melodic function. From this point of view, it would be highly desirable to include specific elements for formal or melodic analysis in future MEI developments.

In this perspective, a certain imbalance can be observed, which will surely be corrected over time, between the development of elements and functionalities available for the bibliographic and material description of musical objects as opposed to the possibilities of semantic and analytical description of musical questions outside the specific situations of the notation systems.

Plans for the future development of the project include the possibility to incorporate some of the few recordings available of some of the works that make up the analyzed repertoire, generally recorded during the research processes that made possible the compilation and transcription of this type of repertoire.

---

Acknowledgments

This work is part of the results obtained in the development of the project “Co-POEM: Platform for the Collaborative Generation of European Popular Music” (ES01-KA201-064933) under the program Erasmus+: KA201-Strategic Partnership.

Works Cited


Encoding Genetic Processes II

Johannes Kepper  
Beethovens Werkstatt, Paderborn, Germany  
kepper@edirom.de

Susanne Cox  
Beethovens Werkstatt, Bonn, Germany  
cox@beethovens-werkstatt.de

Abstract

Traditional music philology aims at establishing an edited text, which is supposed to stage a clearly identified and well-reasoned version of a musical work. Such a text will always depend on sources used for its preparation and decisions taken by the editor(s). However, the intention is to deliver a product – a static text, which resembles a specific combination of the transmitted sources of the work in question. In Genetic Editing, the focus lies elsewhere: Instead of justifying a specific product version, the intention is to trace the creative processes involved in the composition of that work. Obviously, those processes are only accessible through transmitted documents as well, but those documents do not need to contain full texts, nor are they only relevant when the composition has already matured enough to more or less reflect the final work.

The Beethovens Werkstatt project is one of the first endeavors to explore the applicability of Genetic Editing to music. Several years ago, a presentation at MEC 2015 in Florence introduced the first findings of the project and illustrated the then novel approaches of encoding genetic processes in MEI [2]. The discussions of the conceptual model proposed there eventually led to the introduction of several new elements into MEI. Since then, not only MEI has evolved, but also the project. The paper at hand reflects on data model considerations for the project’s current module.

Recapitulation

One of the most important aspects of the original data model of Beethovens Werkstatt was a clear distinction between the encoding of a work’s text and of the documents it is transmitted in. Conceptually, this reflects core aspects of the FRBR data model.1 This was implemented using a single MEI file that encoded the work’s musical text. All process-related information was covered there by using MEI’s <add>, <del>, and <genState> elements.2 Everything related to the scripture was encoded differently: Each penstroke on paper was digitally traced and stored as an SVG <path> element.3 That element bears no semantic meaning beyond that it’s describing an arbitrary shape, and so the visual appearance of the document was encoded as interpretation-free as possible. The connection to the musical text was then established using the @facs attribute on MEI <note> elements, etc., which referenced the corresponding shapes. That way, every note (as any other component of the music notation) explicitly stated by which shape(s) it was manifested in any given source. A user of the resulting edition could now easily access our findings and challenge the material with different interpretations (see Figure 1).

This general approach worked out pretty well for the first module of Beethovens Werkstatt, dealing with compositional revisions in manuscript material. However, the work on Beethoven’s 8th Symphony4 already surfaced some limitations which are now relevant again for the current work.5 Nevertheless, due to the clear separation of document and text, and the good experience utilizing SVG, the model was considered a good starting point for the work in the current third module of Beethovens Werkstatt.

2 See the current documentation of these elements at https://music-encoding.org/guidelines/v4/elements.html (accessed January 12, 2022).
5 The main difference between the 8th Symphony (Op. 93) and the other examples addressed in this module (excerpts from Op. 111, Op. 59/3, Op. 75/2, and WoO 32) was that here the project had to trace Beethoven’s revisions across multiple documents. This brought up deficiencies with regard to establishing the order of writing acts when those acts included simple copying. Then, no MEI ‘processual’ element like <add> or <del> could be inserted into the encoding that could identify the genetic state which this copying act would belong to: Mere copying triggered no changes to the work’s text. Eventually, this led to the omission of a transcription for this example from Op. 93.
Beethoven’s Revision Lists

The focus of our current module is on corrections and revisions that Beethoven generally could not carry out himself directly, but instead asked publishers to incorporate in their prints. Here, Beethoven not only corrected engraving mistakes but also produced substantially new variants. He did so by writing down instructions in revision documents. Those revision documents can take a number of different forms (see also Figure 2):

- Tabular revision lists consist of listings of the changes requested by Beethoven.6
- Quite frequently, Beethoven transmitted the changes he desired in letters to the publishers.7
- Finally, there are revision instructions that he entered directly into a music document, such as an engraver's copy, a proof, or a regular edition.8

Beethoven created revision documents at various stages of the compositional process – some were a response to sample prints he had access to, some were written several months after the publication, and some were written before the work was even printed. The publishers reacted in different ways to Beethoven's instruc-

---

6 See for example Beethoven’s list with revisions for the String Quintet in C minor, Op. 104 (D-BNba, Sammlung Wegeler, W 13).
7 See Beethoven’s letter of 3 June 1823 to Maurice Schlesinger, the publisher of his C minor Piano Sonata, Op. 111, in which he reported printing errors in the Paris original edition (D-BNba, Sammlung H. C. Bodmer, HCB Br 214).
8 For the Piano Sonata, Op. 109, a copy of the original edition has survived, in which Beethoven entered several revisions (D-B, Mus. ms.autogr. Beethoven, L. v. 39,1).
tions. Ideally, everything has been changed as requested. Some changes would have been too complicated to justify the effort. For the same reason, the requested changes were sometimes published as errata lists instead of incorporating them into the prints. In all cases, the revision documents provide interesting insights into Beethoven's compositional processes, as they served as proxies to describe changes that couldn't be carried out directly.

**Piano Concerto, Op. 73**

A relatively simple case is the revision document that Beethoven created for the Piano Concerto in E-flat major, Op. 73. Breitkopf & Härtel published the German original edition in February 1811. Beethoven proofread the edition and created (among other things) a revision list for the piano voice with 25 requested changes. Most of them are corrections of errors, but some also clarify dynamics. In May 1811, a revised edition was published, in which all issues brought up by Beethoven had been incorporated into the original plates. Copies of both the original and revised editions have survived.

The fifth issue on Beethoven's revision list for Op. 73 (Figure 3) illustrates the typical components of Beethoven's *monita*. First, Beethoven often identifies the region where the change is to be made with a marginal note, e.g., "2tes großes tutti" [second great tutti]. The very segment to be changed is then given only by the music excerpt, which provides the context for the actual correction. On the right margin, Beethoven explains this correction ("Änderungsimpersativ", imperative of change). The "C/C statt E/E" written there indicates that a chord has incorrect pitches. Within the music segment given on the left, Beethoven adds an "X" at the fourth beat to clearly state which chord is wrong. Only when considering all these components does it become possible to understand Beethoven's instruction. The results can be seen when comparing the second-to-last chord in the left hand of the piano in the original and revised edition (Figure 4):

---

9 One example is the Piano Concerto in E-flat major, Op. 73, for which a revised edition was published (see below).

10 Beethoven's Irish folksong settings, WoO 152, were published in March 1814 by George Thomson in Edinburgh. In September 1814, Beethoven sent a revision list to Thomson (D-B, Mus.ms.autogr. Beethoven, L. v. 29 V, fol. 173–175). In a revised edition published in late 1814 or early 1815, only a part of these revisions were implemented.

11 Beethoven's revision list for the String Quartet in E-flat major, Op. 127, was published in the journal *Cäcilia* (Intelligenzblatt no. 24, April 1827) by Gottfried Weber. The revisions were not incorporated in the printed editions.

12 D-BNba, C 73/9.

13 A-Wn, SH.Beethoven.323.

Encoding Revision Lists

The case of Op. 73 shows relevant similarities to the work done in the first module of the project. In order to trace all metatexts\(^\text{15}\) such as the marginal notes and clarifications, it was clear that, at least for the (usually manuscript) revision documents, their scripture needed to be encoded using SVG. An encoding of the affected shapes in all relevant documents (including prints) would better illustrate the context and results of a requested change, and make it easier for users to follow and challenge the edition. Accordingly, the project decided to use SVGs as in the first module.

However, there are also some significant differences to the first module. One of the most notable ones is the number of revision layers for the segments\(^\text{16}\) of the text considered: Whereas in the first module, where we sometimes faced more than a dozen revisions as stacked writing layers,\(^\text{17}\) the revision lists usually describe a single change, with occasional exceptions of two such revisions for the same position (see below). As all changes requested in any single revision list conceptually come into effect simultaneously, establishing the genetic order of those changes is easy.

Another important difference is the mere presence of multiple documents. This leads to interesting shifts in the relation between document and text. When writing down the context for a change into the revision list, it was sufficient for Beethoven to clearly identify the segment, without having to copy all the musical content of the section. This, and the occasional mistakes he made when he actually did copy content, sometimes resulted in ambiguity, or at least differences between what those documents read and what the text should actually be like. Likewise, not all changes Beethoven requested were implemented properly by the publishers, leading to a difference between the intended text (“Zieltext”) and the achieved text (“erzielter Text”). While the first module had a very clear 1 to 1 relation between document and text, where the differences between both were sufficiently covered by the use of SVG, we now see a more complex relationship that necessitates an equally elaborate encoding model.

While it seemed still appropriate to cover the scripture of documents by using SVG shapes, it became clear that this isn't sufficient to distinguish document and text properly in situations with more than one witness involved. Our first reaction to this insight was to explore the option to use individual MEI files for each document, plus another separate file for the work text. That way, we could more easily transcribe different texts for each document – initial print \textit{ante revisionem}, revision document, and print \textit{post revisionem}. In addition, the revision instructions that Beethoven included into letters as verbal instructions called for TEI-based markup, which required some kind of structural separation anyway.


\(^{16}\) While the proposed encoding model is designed to allow full encodings of the works in question, \textit{Beethovens Werkstatt} actually only provides the music encoding for those segments that are subject to Beethoven's revision instructions, while everything else is addressed on the level of measure positions in the facsimiles only. This already gives a 'skeleton encoding' of measure elements which allows to encode all clefs, meters, and key changes in the places where they actually occur in the document, instead of compiling the then current parameters into a fictive score definition at the beginning of each revision instruction. That way, the encodings are selective more or less by chance, but enriching them with additional content later will not require any changes to the material encoded now – conceptually, the encodings are complete already.

\(^{17}\) In the first module, those passages were referred to as "Textnarben" (textual scars). See https://beethovens-werkstatt.de/glossary/textnarbe/ (accessed January 12, 2022).
However, using multiple files for an edition comes at the price of having to coordinate them through dedicated markup of some kind. One option considered by the project was to implement something very similar to Frei-schütz Digital's core model. There, a central MEI file (the ‘core’) holds the encoding of the musical text, but no information about its visual appearance in any of the sources. These sources are captured by individual encodings of their own, which share the main parameters of the text and link to the corresponding elements in the core file instead. For example, in that model, a note’s pitch is to be encoded in the core, but the stem direction is addressed at the source level. This results in a plethora of links between these files, which are both hard to generate and maintain. However, there are very few alternatives to reduce those efforts. One such approach could be to follow the Enhancing Music Notation Addressability (EMA) concept. EMA is designed to reference music snippets independent of the data format they are encoded with, by specifying one or more ranges of measures, staves within those measures, and finally beats. The resulting EMA statement is expressed as a URI, and as such can be easily used in conjunction with MEI. With this concept, it would be possible to create links not on the level of individual notes, but for full text segments that are subject to a revision instruction. As current EMA lacks precision to selectively address only some of the content in a linked region, like dynamic markings only, it seems reasonable to mimic the concept with MEI markup that allows greater precision.

Listing 1: Schematic encoding of two file regions identified and linked by MEI <annot> elements.

Listing 1 shows how MEI <annot> elements are used to identify specific regions within two files through time-stamps, which are then linked using a <relation> element and the isRevisionOf relation taken from FRBR. An additional @plist (participant list) attribute on <annot> could be used to specifically identify some content in those regions. This model requires much less effort to connect the different files involved, and therefore seemed like a reasonable approach to follow.

**Diabelli Variations, Op. 120**

In order to challenge this provisional data model and to verify its applicability, the project had a closer look at other, significantly more complex examples. One such example is a revision document for Beethoven’s *Diabelli Variations*, Op. 120, which is included in the “Engelmann sketchbook” that Beethoven used in early 1823. This document significantly differs from the list transmitted for Op. 73 described above. It was not written after but while the work was printed. It holds revisions for a revised copy and the autograph – so it is targeting manuscripts, not a print. Both of those manuscripts were temporarily out of Beethoven’s reach at different times.

---

20 See https://music-encoding.org/guidelines/v4/content/introduction.html#timestamps (accessed January 12, 2022).
21 D-BNba, Sammlung H. C. Bodmer, HCB Mh 60, p. 16–18.
22 The autograph served for some time as an engraver’s manuscript for the publisher Anton Diabelli, and the revised copy was to be the engraver’s copy for a planned London edition. Therefore, the revised copy was handed in for transmission to London in the first days of May 1823, but was not actually sent until early July 1823.
so he needed to coordinate the ongoing revision of the work between those documents. In contrast to the Piano Concerto, the *monita* here do not only concern corrections of errors, but also introduce new compositional variants. Implementing the changes later was much easier: Beethoven could do it himself, e.g., by overwriting. At the same time, this type of revision makes the original text much less accessible than is the case for Op. 73, where it is available as a separate printed document.

Another important aspect is that the revision list was written into the “Engelmann sketchbook” – Beethoven’s private working document of the time that he maintained for his own use. Accordingly, the revision instructions are just reminders for himself, and they could be written down in a very rudimentary form as long as they were functional as an *aide-mémoire*. The localisation is less precise, less context is given, less explanations are provided of what exactly is to be changed, and Beethoven’s handwriting is more sketchy.

A major complication is that at some point Beethoven revised the revision list itself, as can be seen from the red ink he used for it. This means that the source and target documents flip their functions – for some revisions, Beethoven changed his mind about the ‘correct’ solution several times. As a result, the identification as source, target, or revision document is not possible at the document level, but needs to be determined for each individual revision instruction. It also means that any single document may hold more than a single state of the work’s text, which results in the need for process markup like `<add>` and `<del>` to be used as well in the MEI files for each document. This requires coordinating such process markup across multiple files.

A good example for such complications can be found in measure 16 of the 26th variation of Op. 120 (Figure 5). Here, the source document is the revised copy, and the target document, in which the change is to be inserted, is Beethoven’s autograph.

![Figure 5: Diabelli Variations, Variation 26, m. 16, in the revised copy, the revision document, and the autograph (D-BNba, Sammlung H. C. Bodmer, HCB Mh 55, fol. 26v; D-BNba, Sammlung H. C. Bodmer, HCB Mh 60, p. 17; D-BNba, NE 294, p. 52).](image)

The revised copy originally read a dotted eighth note $g$ in the lower system. This initial reading is no longer present in the manuscript shown in Figure 5 (left), as Beethoven implemented the change in place by erasure. Beethoven’s first change was to remove the augmentation dot. Accordingly, he stated “kein . bej g” [no dot at g] in the revision document (see Figure 6, right).

![Figure 6: Left: Initial reading of the revised copy of Diabelli Variations, Variation 26, m. 16, left hand. Right: Transcription of the revision document in its first version.](image)
Beethoven planned to make this revision in the autograph, to which he had no access at the time of writing because it served as the engraver’s manuscript for the edition published by Anton Diabelli. Apparently, Beethoven decided differently when he got the autograph back. He did not only delete the augmentation dot, but also changed the duration of the eighth note to a sixteenth. The second revision was done in the reverse direction: This time, the autograph was the source document, and Beethoven had no access to the revised copy, so again, he kept a note about the intended change in his sketchbook (Figure 7).

Beethoven used red ink to update the revision instruction by adding a second flag to the note in question (Figure 7, center). Later, he copied that flag to the revised copy, so that all documents ultimately read the same final version of the text.

Coming back to the encoding, the challenge of having to include the textual development not only in the encoding of the text but also in the document files becomes very apparent in this example: All four files provided in the original data model would have about the same content. The text file would embrace encodings of the original version (A), the first revision (B), and the second revision (C). The autograph would hold at least versions A and C, the revised copy would provide versions A, B, and C, and the revision document would hold versions B and C. Both the autograph and the revised copy serve as source and target of separate revision processes, and so their respective encodings need to do the same. The markup between those files could be coordinated using `<genState>`, which also clarifies the chronological order. But still, this results in a very widespread duplication of content across multiple files. At first sight, this seems to contradict the intended separation of document and text. Accordingly, we continued our considerations of the different alternatives and tried to explore if we couldn’t simplify the model again by returning to the conceived clarity of the first module’s data model, which was based on a single MEI file, used in conjunction with SVG only.

**Going Back to a Single File?**

Instead of utilizing a separate file for each document, plus one file for the text (which can easily end up in almost all other files as seen above), it seemed reasonable to return to the basic principles of the first module’s data model. When multiple files end up having the same content, why not restrict oneself to just one file? Such a step would fully avoid duplication of both markup and marked up content. This single file would then focus on encoding the work’s text. Multiple `<facsimile>` elements used to capture the page setup of all documents could be easily integrated into this file – a single `<measure>` element could be linked to `<zone>` elements within the `<surface>` encodings of different documents. Retrieving the location (or absence) of each measure within a given document would be equally possible (see Listing 2).
Determining the right level of ‘diplomatic accuracy’ for the encodings seemed more challenging. When the encoding aims to capture the textual development across multiple documents, most of the content will be found in more than one source. Just like measures, individual notes (and other components) may link to their respective SVG shapes in multiple documents. *Explicit* metatexts found in individual documents, like marginal notes, revision instructions, and so on, are encoded using MEI’s `<metaMark>` element. This element offers the @source attribute, so it is easy to identify in which document any given explicit metatext can be found.\(^\text{23}\)

But even when leaving those specifics aside, during our considerations of using only a single file again revealed problems with differences in the text provided by the documents: Occasionally, Beethoven unintentionally introduced ‘variants’ or made copying errors when copying the context into his revision lists. For him, it might not have been important to faithfully copy every slur etc., but for a reader, it is sometimes hard to separate those pragmatic omissions from intentional change requests. Accordingly, those differences need to be made explicit in an encoding. They can be captured using a combination of `<orig>` (original, i.e., correct reading) and `<sic>` (markup for apparent errors), nested inside a `<choice>` element.\(^\text{24}\) As with any other editorial markup, those elements may use a combination of @source (the document in which this reading can be found), @resp (identifies the agent responsible for this content), and @state (helps to clarify the chronology of the composition).

But even such an elaborate data model may not always allow to create satisfactory transcriptions. The third complaint from Beethoven’s list for Op. 73 illustrates yet another significant challenge. Whereas the original text for the piano is written on both systems of the grand staff, Beethoven provides only the right hand as context for his complaint and puts it on a single staff, with clefs added to avoid at least some ledger lines (see Figure 8). While the notes stay the same, they are written in quite different ways. With a single file approach, it is not possible to reproduce these individual layouts – unless `<app>` and `<rdg>` elements are used to trace such alternatives. This, however, would result in quite complex files, contradicting the intended ease of use with only a single file. The example also shows that an earlier assumption should be reconsidered: Beethoven’s handwriting is not easy to decipher here, and it requires some time to figure out how the two sources relate. If

\(^{23}\) *Implicit* metatexts, such as an uncommon spacing of individual signs on a page, cannot be encoded easily in this model, though, as they often rely on positional attributes. Covering them would result in excessive use of `<app>` and `<rdg>` elements for rather marginal information, and was thus considered dispensable. Moreover, high-res facsimiles are always available as a fall-back.

\(^{24}\) Like TEI, MEI has no explicit element to encode the absence of a feature (available in different witnesses or elsewhere) – the `<gap>` element available in both formats is somewhat related, but has a slightly different meaning. It is usually connoted as an editorial omission, not an authorial one.
the transcription reflects only the content of the notes, but not necessarily their layout as found in the sources, it provides only limited support. In situations like this, it would be extremely helpful for users of the edition who are not deeply familiar with Beethoven's handwriting to have faithful transcriptions for each document. It would be equally important to have one-to-one relationships between the notes across documents. A graphical user interface to these encodings could highlight the corresponding notes between multiple documents and their respective transcriptions. In the first module of the project, this type of 'in-place assistance' was considered very helpful to better understand the genetic processes taking place in manuscript documents. Even though the examples at hand for the third module usually don't offer the complexity of multiple writing layers stacked in one place, it seems sensible to prepare for such more challenging examples in the upcoming modules of the project. Accordingly, it seems counterproductive to pursue the idea of using a single file only – the limitations of that approach seem too severe to ignore, and the potential benefits turn out to be smaller than expected.

**Data Model Conclusions**

After all these considerations, it becomes evident that any solution needs to be a compromise. There are three independent requirements for the envisioned encoding model, and they will not combine easily.

First, the data model should support a significant level of diplomatic accuracy. While incorrect stem directions might be tolerable, a wrong distribution of notes across staves is certainly not. Examples from the current project module, but certainly also from future modules will be incomprehensible otherwise. Resorting to a common encoding in one file is not only a step back compared to prior achievements of the project, it would also complicate sufficiently accessible editions that faithfully transcribe the available documents as products. Any solution based on a single file would require the use of `<app>` and `<rdg>` elements to a level where this solution is no less complex than an approach based on multiple files.

Second, from a genetic perspective, it is crucial to fully trace the textual development: How do notes in multiple documents relate to each other? If everything were encoded in a single file, process markup like `<add>`
and `<deletion>` would be sufficient to clarify these relations. Any approach based on multiple files requires explicit linking, which not only takes significant work to generate, but is also challenging to maintain.\textsuperscript{25}

The third requirement is something very generic: A data model should be as intuitive and easy to implement and process as possible. The less markup is used, the less error-prone the resulting edition is – making it easier for editors to generate the required markup and to proofread their files afterwards. In addition, it will be easier for developers to write the code for the edition's user interface. So, a 'simple' data model has a value in its own right, as the lower implementation effort is likely to allow more content to be covered.

Obviously, it is not possible to match all three requirements at the same time: A data model may allow diplomatic faithfulness in a relatively simple form, but that will not cover the relations between multiple documents properly. Another approach would be capable of covering these genetic connections in an elegant and simple form using a single file, but that will not allow diplomatic accuracy at the same time. Finally, the combination of diplomatic accuracy for multiple documents and the coverage of the overarching genetic development of the work through some form of linking will inevitably result in a fairly complex data model – an example of a complex problem requiring a complex solution.

As our considerations have shown that the project must not forego requirements one and two, the only option is to minimize the effort of utilizing a complex data model that covers both of them. The most promising starting point for that was discussed in the context of Op. 120. Here, one MEI file was intended to trace the textual development of the work. In addition to that, one file per document would provide an (editorially enriched) transcription of the textual development accessible in that document. While this approach introduces a lot of redundancy especially for easy-to-read printed music documents, it supports a helpful distinction between a more diplomatic transcription of what is actually written (the 'achieved' text) and a more normalized version of what was originally requested (the 'intended' text). As both perspectives often do not align properly, it seems legitimate to allow for both of them independently. The main challenge was then to create the necessary connections between the text and document encodings. As mentioned above, entering thousands of UUID-based references manually seemed dangerously error-prone. Here, the project sidestepped the problem by utilizing the available SVG shapes.\textsuperscript{26} This process simplified data entry for those links significantly. The resulting annotations (see Listing 3) provide great flexibility in linking content at any level of granularity. While this approach does not simplify processing the data model, it simplifies data entry considerably and makes the adoption of this data model actually feasible. With individual files for each document, it allows diplomatic precision in the encoding where necessary. The duplication of content that comes with this approach is actually necessary to allow a proper distinction between (achieved and intended) text and the documents. The very scripture of each document is still encoded using SVG shapes. The genetic processes, though being traced to some degree in the document encodings as well, are properly treated in a separate file, which can establish order of and relations between the documents.

The data model of the project's first module is mostly compatible with this on a conceptual level, so that existing data may eventually be converted to the current model. At the same time, the model seems sufficiently robust to handle research questions and materials that are scheduled for upcoming modules of the project, including an edition of a sketchbook in relation to the works contained. Those modules will serve as a benchmark to evaluate if Beethovens Werkstatt is actually approaching a consistent model for encoding genetic editions with MEI.

\footnote{Especially when using 'non-speaking' identifiers like UUIDs (see \url{https://en.wikipedia.org/wiki/Universally_unique_identifier}; accessed January 12, 2022), it is barely possible to manually control such links. An additional challenge is that links may need to operate on different levels of granularity: While right now probably all links may operate on the level of individual notes, rests, and so on, for sketch material larger musical 'ideas' or 'phrases' may need to be linked. It is evident that such linkage is based on human interpretation and may neither be auto-generated nor auto-validated.}

\footnote{SVG shapes are available and linked to their corresponding MEI elements for all document encodings using a very simple web application called "Genetic Sandbox", which was developed in the project's first module. Here, a single page of a document is shown as facsimile, with all SVG shapes laid on top. The user may now click on individual shapes and highlight them in different colors. A click will bring up the ID of that element, which can then be inserted into the corresponding `@facsimile` attributes in the MEI file. While mistakes are still possible, this mutual reference has proven to help reduce their number significantly, resulting in very reliable connections between encodings and shapes. This approach can be used to support the entering of links between document and text encoding as well. Although the SVG shapes of notes, etc., are relevant for the document files only, they can be temporarily associated to their corresponding elements in the text file using the same Genetic Sandbox.}
Listing 3: Semantic reference between elements from the text file and their corresponding elements in the document encodings. Here, the note with ID xe542fc95-cb05-4731-90d0-146cd0b06246 is linked to the elements by relation/@target. FRBR terminology (hasEmbodiment) is used to describe the relationship: The documents serve as manifestations for the abstract text file. Here the attributes annot/@type and relation/@label serve as temporary aids to proofread the automatic generation of these links. Since they don’t contribute in relevant ways, they will be removed for final publication. All links are relative within a file only as XInclude is used to combine the contents of all files into one logical MEI document.

Acknowledgments

This paper is also based on work of all other project members of the Beethovens Werkstatt team: Bernhard R. Appel, Kristin Herold, Ran Mo, Andreas Münzmay, Elisa Novara, Salome Obert, Agnes Seipelt, Richard Sänger, Joachim Veit, and Jan-Peter Voigt. We would also like to thank Jens Dufner, Perry Roland, and Mark Saccomano for their invaluable input.

Works Cited


METAdata and metaDATA

Clemens Gubsch
Austrian Academy of Sciences, Vienna, Austria
clemens.gubsch@oeaw.ac.at

Dennis Ried
Max-Reger-Institute, Karlsruhe, Germany
ried-musikforschung@mail.de

Abstract

Metadata are a very broad and extremely differentiated subject and ranges from rudimentary catalog data to deeply indexed scientific catalogs (e.g., catalogs of works). In this paper, the concept of metadata in the context of MEI is first examined, before two examples are used to show that metadata are more than just rudimentary descriptions. These examples are also intended to illustrate the extent to which metadata are encoded in the field of music philology and thus represent an attempt to create a little more awareness for the work of the Metadata and Cataloging Interest Group of MEI.

The examples deal on the one hand with the encoding of performance resources and on the other hand with watermarks. In both cases, the possibilities of metadata encoding with MEI version 4 are exhausted and it is discussed which steps are useful and necessary to create an even deeper, machine-readable structure so that these sub-fields of the MEI metadata can also be used for larger scientific purposes such as analyses.

Introduction

The range of disciplines that use MEI is wide, and the requirements placed on the MEI format vary greatly. The variety of metadata that are encoded is also quite large, ranging from rudimentary catalog data to deeply indexed scholarly catalogs (e.g., catalogs of works). However, MEI metadata, primarily generated in music libraries and edition projects, is not limited to these application areas.

But what are metadata in the context of an MEI file? In this paper, the concept of metadata is first examined, before two examples are used to show that metadata are more than just rudimentary descriptions. These examples demonstrate not only how diverse encoding options could be, but also the need for formalization – for example, through best practice suggestions. They are also intended to illustrate the intensity with which metadata are encoded in the field of music philology and thus represent an attempt to create a little more awareness for the work of the Metadata and Cataloging Interest Group of MEI.

The examples deal on the one hand with the encoding of performance resources and on the other hand with watermarks. In both cases, the possibilities of metadata encoding with MEI version 4 are exhausted and it is discussed which steps are useful and necessary to create an even deeper, especially machine-readable structure so that these sub-fields of the MEI metadata can also be used for larger scientific purposes such as analyses.

1 Two Ways to Understand Metadata

As anyone familiar with MEI knows, a MEI document can have two main sections: <meiHead> and <music>. It depends on the use case one is focusing on. We like to discuss two different ways of using MEI in general. In the first case, the MEI header is used as a storage of (basic) metadata. In the second, the MEI header represents the data containing part of the MEI file. We are aware that there are other forms of metadata. However, we focus here specifically on two extremes that have come to our attention in numerous discussions in recent years (GitHub, mailing list, etc.): music encoding with metadata as ‘just METAdata’, and metaDATA as holistic data representing the core content of, for example, catalog data.

METAdata: If we focus on encoding sheet music, the result is likely to be a rich <body> section and, in direct contrast, a lean header if the header is used only to describe what is in the body or the file. We do not focus here on MEI documents with full header and full body as these are ideal cases and out of the scope of this current paper. We rather focus on the descriptive function that metadata often have in the context of sheet music encoding.

© Published in 2022 with Humanities Commons under a CC BY-NC-ND 4.0 (https://creativecommons.org/licenses/by-nc-nd/4.0/) license.
So in this case, METAdata is some additional information to the ‘main’ data on an abstract level and follows the original meaning of the word ‘meta’. This means that metadata describe the <music> section in more detail, but neither belong to it nor change the meaning of the main data. A description at this level often contains, e.g., the title of the file, responsibilities, and so on.

On the other hand, metaDATA can be self-contained and thus self-describing, which can be much more than data pointing to the content of the <music> section. Even though the description here also takes place on an abstract level detached from the body, a great deal of information (such as source descriptions, performance lists, work information or paratexts) can be described, so that the metaDATA itself must be considered as research data – especially if the header is used in ‘standalone’ mode, as it is the case with work catalogs.

Even this simple comparison shows: When we go beyond the description that is generally subsumed under the term metadata to a status which is an extended scientific description (such as detailed descriptions of work and sources), then ‘metadata’ is not simply metadata – but it is data in its own right in a meta context.

With the transition from MEI version 3 to version 4, numerous options were created in <meiHead> that allow machine-readable encoding of metadata. Although the extension of FRBR (Functional Requirements for Bibliographic Records) already made it possible to describe works and manuscripts in detail, the Metadata and Cataloging Interest Group has been discussing for some time gaps and problems where markup options are not yet sufficient to meet the requirements for machine-readable research data. In the following, two issues are picked out to serve as examples.

2 Two Exemplary Construction Sites on Encoding Metadata

2.1 Reflections on the Performance Medium

If we look up the definition of the <perfMedium> element in the MEI guidelines, we learn that this element “[i]ndicates the number and character of the performing forces used in a musical composition” [9]. Let us first consider the first subset of this definition: the indication of the number of performing forces. But what is the number of the performing forces when we have, e.g., two flutes? Is it the number of (physical) instruments, the number of parts (flute 1 and 2), or the number of performers (e.g., double setup at the world premiere)?

Some information about the number of performers is given by the definition of the subordinate <perfRes> element, which should specify the “[n]ame of an instrument on which a performer plays”. There an attribute (@count) is available to indicate “the number of performers” [10]. However, the fact that the coding possibilities offered by this can quite quickly reach limits can be illustrated by an example: The MEI Guidelines contain the following wording: “Where multiple instruments of the same kind are used, the @count attribute on <perfRes> may be used to encode the exact number of players called for” [6]. This means that the value specified in the @count attribute always indicates the number of performers. With reference to the example given in Listing 1, two players are required to play the piccolo, two for the flutes.

```
<perfMedium>
  <perfResList>
    <!-- concert band -->
    <perfRes count="2">Piccolo</perfRes>
    <perfRes count="2">Flute</perfRes>
    <perfRes count="2">1st Clarinet</perfRes>
    <perfRes count="2">2nd Clarinet</perfRes>
    <perfRes count="2">3rd Clarinet</perfRes>
    <!-- and so on -->
  </perfResList>
</perfMedium>
```

Listing 1: Example for <perfRes> encoding from the MEI Guidelines [6].

---

Following this logic the clarinets have three parts, each with three players. Already here the question arises what exactly should be indicated with @count or with <perfRes> in general. An exact indication of the number of individuals playing the parts (single, double, etc.) is not excluded, but rather unusual in the literature available, at least for earlier centuries.

If we apply this logic to the violin, we get the following constellation:

<table>
<thead>
<tr>
<th>Encoding Sample</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;perfRes&gt;violin&lt;/perfRes&gt;</td>
<td>1 performer on violin</td>
</tr>
<tr>
<td>&lt;perfRes&gt;violin&lt;/perfRes&gt; &lt;perfRes&gt;violin&lt;/perfRes&gt;</td>
<td>2 performer on violin playing different parts (e.g. violin I &amp; II)</td>
</tr>
<tr>
<td>&lt;perfRes count=&quot;2&quot;&gt;violin&lt;/perfRes&gt;</td>
<td>2 performer on violin (same part = doubling)</td>
</tr>
</tbody>
</table>

Table 1: Encoding samples (<perfRes>) for violin.

However, the interpretation of the encoding samples presented in Table 1 only works under the implicit (i.e., nowhere defined) assumption that a player always operates on one single instrument. This can be a particular source of confusion when we focus on keyboard instruments or, more generally, on instruments that can inherently be played by multiple performers simultaneously (see Table 2).

<table>
<thead>
<tr>
<th>Encoding Sample</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;perfRes&gt;piano&lt;/perfRes&gt;</td>
<td>1 performer on piano</td>
</tr>
<tr>
<td>&lt;perfRes&gt;piano&lt;/perfRes&gt; &lt;perfRes&gt;piano&lt;/perfRes&gt;</td>
<td>2 performers on 2 pianos playing different parts (primo/secondo)</td>
</tr>
<tr>
<td>&lt;perfRes count=&quot;2&quot;&gt;piano&lt;/perfRes&gt;</td>
<td>2 performers on (2) piano(s) (same part)</td>
</tr>
<tr>
<td>?</td>
<td>2 performers on 1 piano playing different parts (primo/secondo = four hands)</td>
</tr>
</tbody>
</table>

Table 2: Encoding samples (<perfRes>) for piano.

(More recently, in contemporary and especially experimental music, there are no limits to the number of individuals on an instrument or the way it is operated!) Following this logic and definition of @count above, <perfRes count="2">piano</perfRes> represents one piano part performed by two different performers (on two pianos) at the same time. It is therefore the doubling of the same instrumental part, as seen for the violin in Table 1. What about compositions for keyboard instruments for several hands? Finally, piano reductions of symphonies are often set for piano four hands (e.g., Joseph Joachim Raff, *Symphony No. 1 in D Major, Op. 96, “An das Vaterland”,* arranged for piano four hands by the composer) but also transcriptions (e.g., Max Reger’s transcriptions of the Brandenburg Concertos, BWV 1046–1051, by J.S. Bach, RWV Bach-B9). The problem that arises from this form of encoding is that the really common case of piano for four hands cannot be specified.

Of course, by nesting one can build structures that probably represent what piano four hands means, but does the result then still meet the definition of <perfRes>?

<perfRes label="piano">
  <perfRes>primo (1st part)</perfRes>
  <perfRes>secondo (2nd part)</perfRes>
</perfRes>

Listing 2: Encoding piano for four hands by nesting <perfRes>?

---

Finally, in this way of encoding we would use `<perfRes>` both as a wrapper element to describe an instrument and as child elements to describe two different parts. Thus, the idea that `<perfRes>` specifies the “name of an instrument on which a performer plays” is actually alienated. This is because in our example we do not specify an instrument label with ‘primo’, but rather the part and ultimately the function that a performer performs. Thus, by nesting the performance resources within each other, a content inaccuracy arises (see Listing 2).

Let’s go one step further: There are not only works for piano with two or four hands. Besides keyboard works like Max Reger’s *Spezialstudien für die Linke Hand allein* (“Four Special Studies for the Left Hand Only”), WoO III/13, there is also the configuration in which a piano is to be played with three hands, for example.

![Figure 1](image)


Figure 1 shows a score that requires three hands. It is obvious that more than one player is required to perform this (on one instrument)!

According to the title page, this Trio by J.W. Holder is composed for three performers at one piano, even though the chord brackets (stave 2 and 3) indicate an assignment for two performers. Thus, even in this simple example, several instrumentations are possible, which should also be encoded as alternatives. Regarding the distribution of the hands (which performer plays which staves), however, the problem arises that here neither an instrument, nor a performer, but a body part of the performer, namely his hands, must be addressed. As an example: Johann Friedrich Schwencke adds to his “24 postludes in all keys and 24 transitions” a three-hand performance option “for organ with obbligato pedal, or for pianoforte (2, 3, or 4 hands)” [14]. Thus, the need to address body parts is obvious. And that is not an isolated case as shown by the examples in the MEI sample-encodings repository [13] that the authors of this text have collected to provide approaches to how these cases currently have to be solved due to missing markup possibilities. By the way: There are also examples of works for piano six hands (by Carl Czerny and Sergei Rachmaninov).

With regard to the organ, we have yet another entity to consider when addressing ‘body parts of a performer’: the feet. For much of the organ literature, this may sound absurd, since one performer usually plays one organ. But there are also works in the organ literature that require a certain configuration, for example, works for two players at one organ, where only one of them operates the pedal, or where each player is assigned certain manuals. There are also organ works for solo pedal, such as the “Pedal Exercitium”, BWV 598, by J.S. Bach, or for four feet (Johann Strauss Jr., “Waltz”, arranged for two performers on one organ [15]). In order to index these configurations as machine-readable research data, a certain level of detail in indexing is indispensable.

The few examples of organ and piano literature alone already show the need for greater differentiability within the encoding, from which, for example, the description of percussion, to name just one more, would also benefit.
In the second subset of the definition of `<perfMedium>` we learn that “the character of the performing forces” should also be described. In our understanding, this includes, e.g., the tuning of an instrument (e.g., clarinet in E-flat) – the use of the attributes `@trans.diat` and `@trans.semi` could be an approach here – but also the presupposed range. The question here is what do we encode in `<perfMedium>` and for what purpose. We can record a historical performance situation (e.g., a world premiere), but we can also record which instruments (and how many musicians) are necessary to perform a work (which is absolutely relevant for conductors, for example). Furthermore, it is extremely important to encode requirements for the performance. Although it is possible to find out from the score itself whether a double bass part requires a five-sided bass or whether a four-sided bass is sufficient, it must be remembered that this information is only available as ambitus indications in a work catalog.² But in a catalog, there is not necessarily a `<music>` section that contains this kind of information. Especially for vocal parts, it is extremely important that the required voice ranges can be specified. For example, Wolfgang Amadé Mozart wrote the role of the Queen of the Night in his “Magic Flute” for soprano, but this role cannot be filled by any soprano! The reasons for this are obvious.

So if we already have the possibility to encode ranges in a machine-readable way, it should be possible to use the `<ambitus>` element in an element like `<perfRes>`, in order to be able to describe the character of the performing force more in detail. At this point, the requirements for the description of the performing forces are already so diverse that the question should be asked: Why don't we define the instruments themselves?⁴ — Think of the percussion instruments mentioned above, for which a detailed description of any modifications is indispensable, or for other kinds of instrument modifications (the prepared piano).

To conclude this chapter, let us take one last additional step. The best practice recommendations of the Metadata Interest Group suggest that the MARC Instruments and Voices Code List [5] or the UniMARC Medium of Performance List [3] should be used for the identification of instruments, but these lists are too incomplete in some places for general use. Therefore, the question arises whether it is not up to the music encoding community to remedy this deficiency with the creation of its own authority files which are usable for scientific purposes. We envision here, for example, a taxonomy that is made available to, and maintained and expanded by, the entire community. This would allow the community to create its own instrument database that meets the requirements of musicology and can serve as a reference for all MEI related research (#standardization). In addition, such a central database could list instrument names and their equivalents in various national languages and eventually serve as an interface to other formats such as MARC.

### 2.2 Reflections on Watermarks

Watermarks in music manuscripts⁵ present a particular challenge for description in the metadata section of MEI and discoverability in general. In addition to the verbal description of the watermark itself, other parameters such as the position of the watermark on the page, its completeness, information about the paper mill and the possible production period, and the period of use by the composer are necessary for drawing conclusions about the composition periods of the notated works and the composer’s use of the paper.

In general, carefully catalogued watermarks provide dating assistance for compositions that are not autograph-dated but may have been written on the same kind of paper as dated works. This does not only apply to papers used by the same composer; if, for example, contemporaries such as Ludwig van Beethoven and Franz Schubert obtained their paper from the same merchants, comparisons can also be made across composers.⁶

Even such a basic comparison, however, presupposes that, on the one hand, a relatively large corpus of watermarks described in the same quality is available for a given composer of interest and, on the other hand, that other researchers and projects carry out the indexing according to the same criteria, since only then it is

---

3 Ambitus indications are most common for vocal music, although there are many other instruments where such indications would be helpful.
4 There is the element `<instDef>`, but this can’t be used for describing instruments because it is a “MIDI instrument declaration” [8].
5 The following section mainly describes watermarks in music manuscripts of the late 18th and early 19th centuries. The authors are aware that manuscripts from earlier periods are only comparable to the manuscripts discussed here to a limited extent due to changes in paper production.
6 The mere dating on the basis of watermarks generally allows a range of ± 5 years, but the example of Schubert shows that this range can also be reduced.
possible to compare different watermarks and thus music papers. This provides enough justification for discussions about systematizing and standardizing the description and long-term archiving of watermarks.

**Description and standardization**

For several MEI versions, it has been possible to store verbal descriptions of watermarks by means of the `<watermark>` element in the `<physDesc>` of the source description. In order to ensure standardization and improve discoverability at this point, the vocabularies and taxonomies from watermark research should always be preferred for the verbal description of watermarks.

The Bernstein Project, for example, offers a multilingual vocabulary for describing watermarks as well as a basic three-level system for describing the symbols or motifs of watermarks themselves [1, 2]. If a watermark description is based on such a standardized vocabulary, the interoperability of the collected data increases enormously, even if the three-level systematization reaches its limits due to the complexity of individual watermarks.

Another standardization approach for cataloging watermarks is the IPH (International Association of Paper Historians) standard, whose task is described as follows:

The IPH Registration Standard ensures the registration of all historical and modern papers and paper items, with or without watermarks. Therefore some criteria are defined which apply to single paper types only. The IPH Registration Standard is both a technical standard of normalized criteria as well as a standard defining criteria and paper or watermark types, fixing their name and specification in the most important languages, in order to ensure the international compatibility of the paper data [4, p. 2].

The underlying rules are: (1) the use of the extended ASCII character set for machine readability, (2) fixed codes for field names, (3) prescribed subcodes for field contents, and (4) a number of required fields that must be collected [4, p. 22]. The degree of complexity that this encoding of characters can reach, can be seen in the following example:

Description in full (according to 3.1.5): Eagle, crowned, overlaid with italic capitals FWR, holding sceptre and sword in the left claw, an orb in the right claw, at the top the name “G W Loeschge”, at the bottom “in Ansbach”.

Completely coded description (according to 3.1.4 and appendix I): D5/1 [ R3/1 - {c: (i: X ‘FWR")-C1: p06 [ R7 - M14 - {r: p06 [R4 - {t: → Y "G W LOESCHGE" - {b: → Y "IN ANSBACH" } ] }7, p. 18].

Such a description – and the example cited here does not yet include a fragmentary or composite watermark – can apparently not be generated manually but only with a specially created editor and input mask. Therefore, software-specific restrictions were created with the aim of standardization, even though the structuring and description possibilities for watermarks can be described as extraordinary.

It is obvious that this kind of indexing – using ASCII codes – can only be implemented with great effort in an XML-based markup format such as MEI; however, a first attempt at implementing IPH-norms in XML-based formats as a TEI extension module was published by Ermenegilda Müller [11, 12]. Müller adopted the basic structure of the IPH standardization, but relied on linking and nesting in the description of (partial) watermarks as well as meta information such as the paper mill and the paper manufacturer. A similar structure seems to be feasible for MEI.

Because the indexing of a watermark can be extremely complex, it is important to first mention some parameters that are either recommended or optional, such as: (1) the verbal description, (2) whether it is a stand-alone, fragmentary, or composite watermark, (3) possible countermarks and the connection to them, (4) the size of all symbols, (5) the positioning of the symbol on or next to the chain lines, (6) the distance between the chain lines, (7) the positioning of the symbol on the paper, (8) information on the period of use of the paper, (9) information on the paper mill and the paper manufacturer, and (10) the inclusion of graphics on the specific watermark.
The abundance of information already indicates that the existing markup options for `<watermark>` within `<physDesc>` or `<support>` should be more structured for further machine data processing. Here, it would be preferable to create a wrapper element such as `<watermarkDesc>`, which stores – analogous to Müller's conception – the information on the mark itself and the meta information on the paper production in two separate child elements `<watermark>` and `<papermill>`.

The information on the paper manufacturer as well as the references to it can be marked by `<persName>`, `<corpName>`, `<geogName>`, and `<bibl>` in the usual way, whereby especially the production period of the paper and possible changes of ownership of the paper mill make a `<history>` element seem more sensible. Because changes of ownership often involve changes in the watermark itself – such as a change of initials or additions to the motif – either a secure period of use should be indicated or the described watermarks should be linked to an owner.

The verbal description of the motif of the watermark should, as mentioned, be based on existing vocabularies, but since these databases are by no means complete, the existing form of description by means of free text should be retained. In addition, there is the option of adopting systematizations by means of `<identifier>` and `<termList>`.

As Müller already proposes, an attribution with a closed list is suitable for differentiating between main, counter, or partial watermarks [12, p. 15]. Equally relevant is the question of whether it is a composite motif (see below), whereby a distinction must rather be made as to whether the motif was described as a composite motif or its parts individually. Therefore, it should not be asked for part 1 or 2, but whether a reconstructed watermark is already present or not (`@composite="true/false")`.

In order to define the connection between the main watermark and its countermark(s), the common solution in the MEI header using `<relation>` can be applied – contrary to Müller's suggestion of a `<countermark>` element [12, p. 8]. This way, a linking mechanism similar to the FRBR relations could be established here, utilizing `<@rel>` with values `hasCountermark` and `isCountermarkOf`.

For the description of the size of the motif, existing markup forms can be used by means of `<dimensions>`, `<height>`, and `<width>`. Likewise, the mere verbal description of the positioning on a page and on which page could be mapped using `<locus>`. The integration of already existing depictions of the watermarks – whether taken with transmitted light, thermography, manually, or by means of radiology procedures – could be realized trivially via `<graphic>`, but seems to be particularly important in view of the analysis of large watermark corpora through signal processing and machine learning.

The need for a precise description of the position of a watermark on the page becomes particularly clear when attention is directed to fragmentary motifs:

![Figure 2: "Watermark DE0960-Schubert13_5". Source: https://www.wasserzeichen-online.de/?ref=DE0960-Schubert13_5 (accessed January 12, 2022).](image)

---

7  Cf. file `../Watermark/watermark_ex_03_connecting_to_WZIS_and_using_terms.xml` in [13].
8  Exact linking for page positioning can also only be established for composite marks if the sub-characters are described separately.
9  Cf. file `../Watermark/watermark_ex_02_countermarks_and_fig.xml` in [13].
The two fragmentary parts of the Welhartiz watermark (Figure 2) are each located at the margins of the sheet, so that it is only through reconstruction that the ‘original’ and uncut paper as sold by the papermaker can be ‘restored’. Particularly when composers used many separate individual sheets for their compositions, as in the case of Schubert, the reconstruction of the ‘original’ and uncut paper can provide information about the compositional process and period. However, with a larger number of individual sheets, ‘analogue puzzling’ out possible matches manually is subject to restrictions – due to access to the manuscripts and their preservation protection –, the digital storage of this information offers advantages both in terms of time economy and manuscript conservation.

Information on layer order and physical additions (such as cutouts and patches) to a manuscript can be already stored in foliaDesc. So it would make sense to encode the positioning of watermarks on the manuscript page in the same way as a patch is contained by folium or bifolium, so that watermark is also possible in foliaDesc and thereby also takes over the attribution of patch. The existing limitation for patch that “at this point, it is not possible to specify rotation” [7] needs to be overcome for watermarks as well, since the rotation of the motif matters in a special way for the reconstruction. A more detailed description of the motif could then be linked via xml:id to watermarkDesc.

Similar methods will probably have to be used to indicate the position on the chain lines, the number of chain lines, and the distances between them. For the purposes of mere discoverability, it seems useful to add the position of the watermark on or next to the chain lines as an attribution to the description of the motif itself (chain "in.between", "on.it"). And also to add the number of visible chain lines on which the motif is positioned by means of quantity. However, it is a description of the surface so that it seems consistent to store this information also in foliaDesc. As Müller suggests [12, p. 13], an element chainLines could be used as a child of folium, specifying the starting point of the first chain line on the page (start), the number of chain lines on the page (quantity) as well as the distance between these lines (distance), and thus making it possible to precisely determine the positioning of the watermark on or next to the chains. In addition, it is necessary to specify whether the chains run horizontally or vertically (orientation), which is determined by the orientation of the paper after scooping.

Of course, it should not be forgotten that there are already databases for recording watermarks and linking to them using ptr may be sufficient for certain use cases of manuscript description. For more complex and extensive source databases, however, it is necessary to create the description of the watermarks yourself, since only with the standardized recording of these metaDATA, analyses can be carried out on the use of papers by a composer. It can be shown that the standardized recording of watermarks requires a multitude of parameters that go far beyond the mere verbal description of the motifs. For music manuscripts in particular, MEI should therefore be the format that enables the capture of such data.

Résumé

The question of what metadata are and why they should not be provided just because of the ‘mere’ documentation obligation was to be discussed. Even the simple distinction between METADATA and metaDATA makes it clear that data recorded in the MEI header should not only be understood as mere additional information for encoded musical content, but also be regarded as research data. In many aspects, standardization and machine-readability are the main focus when collecting such research data, as this is the only way to ensure comparability within one’s project data as well as exchange within the community.

In the case of the performance medium, it becomes apparent that a multitude of possibilities of musical practice are not yet adequately recorded with best practices. Existing organ and piano literature and their instrumental disposition can only be recorded up to a certain point, which prevents further (automatic) analysis, e.g., to inquire about repertoires for certain arrangements. Likewise, another possible problem to be solved is the linkage between existing part material and historical performance scenarios as well as the

10 To complete the reconstruction, the two other folios of the original sheet are also needed, as they contain the eponymous writing "Welhartiz" and refer to the location of the paper mill.
linking between persons who, for example, performed a work in piano arrangement for four hands would be a necessary further development, in order to connect work information with musical (historical) practice. An important goal on this path would be a database for instruments managed by the MEI community.

The markup of watermarks also shows that, on the one hand, the high complexity and number of parameters to be considered requires a high degree of standardization. Although, on the other hand, existing databases offer many of these possibilities – i.e., composition of the watermark or its verbal description – and watermarks can be included as graphics, MEI as a format for describing music manuscripts should both offer a best practice suggestion for indexing watermarks and be interoperable with other indexing systems, in order to exploit the synergy effects that can arise, for example, from the exchange with other paper-related research areas.

**Works Cited**


United, Linked, Connected – A Data Model for the Inventory of the Former Detmold Court Theatre (1825–1875), or: How Library Inventory History Can also Be Told

Kristina Richts-Matthaei
Paderborn University, Germany
kristina.richts@uni-paderborn.de

Irmlind Capelle
Paderborn University, Germany
irmlind.capelle@uni-paderborn.de

Abstract

Library forms of cataloging may differ greatly from the cataloging requirements of musicological research projects: They are often not detailed enough and do not take a close enough look at aspects of content relevant to research, such as handwritten entries in materials, etc. Library catalog entries of individual documents stand on their own for historical reasons, but usually do not reflect relationships to other surviving materials. This observation was the starting point for the Detmold Court Theatre Project, a six-year research project (September 2014 – January 2021) that looked at the interconnectedness of different surviving materials of a 19th century theatre company that existed from 1825 to 1875. This project dealt with a very detailed form of inventory indexing in order to hand over and make accessible the formerly related materials in their entirety. This form of indexing was called ‘contextual deep indexing’. This special form of indexing took into account not only the pure performance materials but also the surviving theatre files, such as fee books, revenue and expense documents, stock lists, director’s books, role and costume books, etc. All information was recorded based on autopsy (in the case of musical records on the basis of already existing RISM records from the 1980s). It was the first attempt to carry out such a form of indexing on the basis of the MEI and TEI encoding standards for a large repertory. For this purpose, a data model was needed that focuses on the linking of MEI and TEI data and enables the linking of different surviving library holdings, with a focus on a FRBR-based indexing of performance materials and associated performers as well as the structure of the theatre. Using a custom ODD-based schema, separate records were created for all works, expressions, and manifestations (in this case preserving the unity of materials kept under a common signature) and given unique identifiers so that they can now all be referenced individually.

The paper summarizes the results of this pilot project. It addresses the particularities and requirements of an inventory development that does not focus on individual objects, but on the relationship between different objects (and subjects). It presents a document-oriented (not object-oriented) data model that uses library materials to revive an entire network of a long-gone organization.

Introduction

The field of music research in the digital age is currently subject to substantial change. Libraries provide more and more of the materials preserved by them in digital form. They combine the digitized files with metadata sets or provide links to other catalog entries such as the RISM OPAC with its detailed descriptions of music manuscripts. In the best case, music researchers nowadays can get digitized material and bibliographic descriptions with just a few mouse clicks. With the current developments in the Digital Humanities, the potential for new research methods is also growing. This development is reflected primarily in editorial projects, such as Beethoven’s Werkstatt,1 and catalog projects, such as RISM2 or Bach digital.3 It has been accelerated by the


© Published in 2022 with Humanities Commons under a CC BY-NC-ND 4.0 (https://creativecommons.org/licenses/by-nc-nd/4.0/) license.
data format of the Music Encoding Initiative (MEI),\(^4\) which offers promising opportunities for indexing metadata of musical holdings and the long term preservation of these data. One of the distinguishing features of the MEI XML schema recommendation is that it accommodates very comprehensive metadata in the file header. The abundance of data ranges from formal components like title information or names of persons, times and places of origin to comprehensive information about instrumentation, watermarks, the conservation status of historical musical sources, and much more. This is an essential prerequisite for the indexing of data of musicological research projects in an adequate depth. The schema also allows multiple degrees of integration of the linkage to digitized data and/or transcriptions. Extensive linkage and mapping possibilities create a connection to existing library data and thus enhance an extensive use of the indexed data. MEI also supports the encoding of multiple incipits, which provides significant opportunities for using mark-up that go far beyond the capabilities of other formats like Plaine & Easie,\(^5\) for example by supporting multipart encodings, text underlays, and instrumentations.

The Detmold Court Theatre Project\(^6\) funded by the German Research Foundation (DFG) was the first research project that used MEI as a data container for cataloging and for archival purposes at the same time. The most important aim of this project was to demonstrate the potentials of the MEI standard by a new method of cataloging on the basis of the holdings of the former Court Theatre, which existed from 1825 to 1875, and to develop an MEI and TEI based model of contextual deep indexing of musical collections. The remarkable thing about this collection, which is stored in the Regional Library of Detmold,\(^7\) is that the material components of the Court Theatre have been retained in an astonishing variety and completeness. Additionally, almost all contextual archival materials (mostly written file material, such as revenues and expenditures, fee booklets, inventories, role and costume books, daily reports, theatre journals, etc.), which have been preserved in the Regional Library of Detmold as well. In the context of cultural history, these combined collections prove to be extraordinarily informative for cultural research.

The collection of performance materials from the days of the Detmold Court Theatre consists of 268 individual materials related to opera and incidental music, the most of them comprising a score, parts, scripts, libretti, role books, text books, director’s books, and soufflé books. These materials were all cataloged with MEI based on autopsy to illustrate the advantages of more in-depth cataloging of materials in comparison with traditional catalog records, where the focus is always on a single document, and mostly on the physical properties of the document rather than the content. But what does an indexing actually require that does not prioritize the individual object but the relationship between different objects (here: documents) and subjects?

1 A Data Model for Contextual Deep Indexing

1.1 What Means ‘Contextual Deep Indexing’?

The holdings of the former Detmold Court Theatre are nowadays kept in a very classical, library-specific manner: Each material is stored as an independent collection unit under a shelfmark that conveys to those familiar with the collection that it belongs to the Court Theatre collection. In the boxes typical for this collection, all the handwritten and printed performance materials used by the former theatre company are found in close proximity to each other. Taken together, they tell their stories [2] – each material separately, but also all the materials as a whole. Stories about plays that were performed, stories about how these plays were performed, and – related to this – stories about people who were significantly involved in these performances. But: Their history remains hidden in the materials if it is not coherently reconstructed.

This history of the former Court Theatre and its employees and participants can of course be compiled by looking through the materials in the reading room, as it has been done for decades, but this approach would be rather tedious, time-consuming, and also not very practicable when dealing with certain questions, since there are no less than 268 materials. Establishing (content-related) cross-links between the materials in this


\(^5\) In librarian practice, the Plaine & Easie Code is used for the encoding of musical incipits.


\(^7\) http://www.llb-detmold.de (accessed January 12, 2022).
way is therefore really out of the question and could only be done on the basis of a selection. In addition, the other surviving sources, such as file materials or director's and costume books, would have to be examined and added to the stock of performance materials in order to obtain further valuable information about the performances.

The Detmold Court Theatre project therefore dealt with a completely new form of inventory cataloging to show how combined library and musicological indexing could look like in the long term. Based on various factors, all the surviving materials of the former Detmold Court Theatre were subjected to a kind of ‘overall view’, i.e., each was looked through individually, re-recorded, and related to each other on the basis of certain recurring parameters such as personal names, work names, role names, or (performance) dates. However, this approach first required the creation of a digital environment in which the materials were recorded and described in a way that enables cross-connections between them. This required individually and unambiguously referenceable objects, a profound data model, and a common presentation interface that brings out the connections between the individual objects.

According to our understanding, ‘contextual deep indexing’ thus means: to record the information of the individual sources with content mark-ups so deeply that they can be linked and related to other objects.

1.2 The Data Model

In the field of digital editions, it has become standard practice to make contextual material available as full text and to link it via markup (see the presentation of Freischütz Digital, which is exemplary for the field of music theatre). However, in the field of indexing collections, proprietary databases or library standards are still used to a great extent and cannot be made accessible on the web without further ado, and above all, different types of sources are recorded separately in their own systems. This is evident, for example, in the current discussion in the library sector about the recording of ephemera [4, 5]. Especially in the field of the indexing of playbills, which is important for theatre research (see, for example, the Düsseldorf Playbill Database developed at the University and State Library of Düsseldorf, the Weimar Playbill Database or the Bremen Playbill Database), this has led to numerous isolated solutions, which makes a comprehensive search – e.g., for actor names – impossible.

![Figure 1: Data model of the Theatre Tool.](https://freischuetz-digital.de/ (accessed January 12, 2022).)

The Court Theatre Project has developed a data model for the contextual deep indexing of the various materials that have been preserved. This data model (see Figure 1) forms the basis for the "Theatre Tool" software that was specifically developed for the presentation of the data. Starting from the performance materials, which were initially split into works, expressions, and manifestations according to the FRBR principles, references to all the persons and works mentioned in other materials were systematically entered. As a result, one can now see which person played which roles in which works, which people were part of the theatre company in which years, what the playbills looked like in the individual years, at which venues which plays were performed, etc. In contrast to the full FRBR model, via which relationships to persons and objects can of course also be mapped, only the entities of FRBR group 1 (work, expression, manifestation, item) were used in this data model. The individual persons were referred to with the help of identifiers; more detailed descriptions of the persons were collected in separate data records, with reference to already existing authority data records (if available). The roles or functions performed were recorded in MEI with the help of the @role attribute, which was semantically sufficient in this case.

By using the FRBR model for works and expressions of the work, the data collected meet both library and academic requirements. The indexing of the sources is carried out according to FRBR on three different levels: The work files record the basic data, if applicable with the date of first performance and a standardized indication of classification. The source files (corresponding to the FRBR entity manifestation) describe the available sources, which in this case are combined into a 'componentGroup', since the performance materials form a unit.

The link between work and source is always the expression file, because the respective performance material of the theatre in Detmold is as a unit an expression of the work, while that of another theatre would be another expression. An adaptation of an opera for wind ensemble, for example, would also be another expression of the same work. The relationships between the files are described with relations as specified by FRBR: has-Realization, isEmbodimentOf, hasEmbodiment, isPartof, etc. (see Figure 2).

In addition to these files, which are necessary for source indexing, files for persons and dramatis personae were created. By means of a unique ID for each file, clear identification is possible in case of any recurrence of a work, role, or person name. While these files are typical catalog indexes which, at least in the case of the source files, go far beyond the usual library indexing, the extensively handed down contextual material is indexed partly as registers, but mainly in full text. Both forms of indexing are based on the XML standards TEI and MEI, so that all data can be marked up (see Figure 3).

In addition, authority files (GND, VIAF, GeoNames) are used for persons, works, and, if applicable, places, so that external information can be integrated. However, since many persons and works are not well known or cannot be clearly identified and thus cannot be clearly assigned to an authority data ID, the use of project-specific IDs remains necessary.

---

Linking the data across all source boundaries results in various connections in terms of content: Thus, for the persons of the Detmold Court Theatre, on the one hand, the data on salary and possible special allowances, length of employment, and additional occupations in the theatre business can be retrieved, and on the other hand, the works and even the roles in which these persons were employed. For the performance materials, information on dating and scribes is linked from the files, and the entries in the costume and director's books provide the first clues to the presentation of individual works on stage.

In summary, it can be said that the theatre network (see Figure 4) is a network which focuses on the individuals and relates them to their activities that can be traced back in the surviving sources. This sets it apart from other, event-based\(^\text{15}\) or performance-based\(^\text{16}\) models that are thought of in terms of the individual event or the individual performance. Individual events only appear in the theatre programs from 1820 to 1847, which could be reconstructed on the basis of the theatre's surviving revenues and are now generated by the Theatre Tool on the basis of TEI indexing.


\(^{16}\) A discussion on presentation models of performance-based informations took place at the DHd Conference 2020 "Spielräume: Digital Humanities zwischen Modellierung und Interpretation", March 2–6, 2020, in the panel of Kathrin Dennerlein "Datamodelling Drama and (Musical)theater", see [3].
2 Presentation of the Data/Visualization

All data collected in the project context are visualized with the help of the software Theatre Tool and presented in the Detmold Court Theatre Portal.¹⁷ This is based on XQuery and JavaScript. Within the portal, information is compiled and displayed in bundles with the help of references that query the assigned IDs of individual person, work, and role data sets. For example, users can search specifically for individual works or select them from the repertoire list and receive a work record as a result, which provides references to those areas of the portal in which the work occurs. For example, Figure 5 shows the references for the work La Dame Blanche, including links to the theatre program, income, expenditure, theatre journals, director’s books, taxations, daily reports as well as role and costume books.

If facsimiles are available for a work or if performance materials have been indexed, the user will see this to the right of the work information. General information on the material as well as descriptions of the individual sources are displayed under the respective library signature. The expression file connecting work and manifestation is not visible, but contains the incipit encodings from which the incipit displayed in the portal is generated.

The facsimiles are displayed with the help of a Leaflet plug-in, a library for displaying interactive maps on the web. So far, the software offers a simple search for persons, roles, and works, created with a Fuse.js plugin, a fuzzy-based library.¹⁹ As recommended in web presentations, the contents can be downloaded as XML files to enable further work with the data (search, query in a larger context, etc.). Of course, the data can also be used as examples for indexing in other projects. The works, sources, persons, and roles can be directly referenced by other projects with the help of permalinks.

Since the Detmold Court Theatre Project has so far mainly indexed materials on music theatre, some music-specific applications have been integrated into the software. For example, the beginnings of individual musical numbers are reproduced with note incipits to make them quickly comparable. In order to provide the musicologist with information on the original score arrangement, clef, spelling of the instruments, etc., not only a voice or piano score is reproduced – as is traditionally the case –, but the first measures and the vocal part are reproduced in full score. The encoding of the musical incipits is done with MEI, the presentation with a Verovio20 plug-in (see Figure 6).

Another special feature of the project is the exemplary deep indexing of some selected performance materials: For these, the facsimiles of the sources are also made available, in a format that allows access to the exact measure. Only this form of indexing makes it possible to indicate interventions in the musical text not on the basis of materiality (e.g., deletion on p. 4v to 5r penultimate measure) but on the basis of content (e.g., deletion of mm. 17–20 in no. 1) and thus in a comprehensible way for the user. The software Edirom is used for an automatic identification of measures, and the Theatre Tool is linked to Edirom Online21 for the presentation. Although Edirom was developed for the preparation of musical material, it can also be used to map text sources, e.g., by scenes or even lines.

The Theatre Tool is designed to represent these complex text and data structures but can easily be adapted to other requirements: For example, the material recorded in the project is predominantly handwritten material, which is why the fourth level envisaged by FRBR, the copy (item), is not taken into account according to the ‘manifestation singleton’ rule.22 Of course, this level could also be represented. Since the main interest of the indexing lies in the working methods and personnel of the Detmold Court Theatre Society, the locations mentioned in the sources are marked, but there are no independent files for them (with the possibility of references) and so far no search option.

22 Manifestation singletons are at the same time source and item, because there is only one copy of them, e.g., autographs.
With the increasing digitization of library collections, the digitized files could be integrated into the Theatre Tool via IIIF, which would solve a number of legal problems. How these can be synchronized with the sources indexed in terms of content still needs to be examined.

3 Contextual Deep Indexing: Needs and Perspectives

Based on project experience, it can be said that there is a need for further coordination between science and libraries, but this is of great interest to both sides. It is clear that such a form of contextual deep indexing as was carried out in the project described here as well as the work on composer work indexes, for example, can only be undertaken by specialists. Nevertheless, there is great interest in making this detailed information on individual sources accessible via the owning libraries as well. The use of standards and authority files as tested in the Court Theatre Project is a first step in this direction, but more thought must certainly be given to interfaces for data exchange. There is a great need for interfaces for (automated) data exchange, especially on the part of research projects. The connection to library workflows is still completely lacking here. The heterogeneous library and catalog landscape found in Germany, for example, does not simplify things, but instead complicates them. In the course of the project, we also noticed that it is not so easy to obtain appropriate work authority files, not to mention data sets for individual expressions of a work. It was a particular concern of ours to link all the works in the repertoire of the Detmold Court Theatre with authority files. In reality, however, we had to learn that only a small part of our work datasets could be linked with good conscience to already provided data sets, and that (in the case of Germany) it is also not so easy to contribute data to the Integrated Authority File (GND) as a research project because there are no workflows for automated processes for this so far. So once the technical hurdles have been overcome, we hope that we will still be able to contribute a great deal to the expansion of the GND Werknormdaten after the end of the project.

A similar problem exists with regard to the handing back of RISM data. Thus, for source indexing, we were thankfully able to make use of the LOD datasets provided by RISM, converted them to MEI and enriched them with many, many details on the individual source components. But how can we return them? And if we do return them, are the data structures on the RISM side sophisticated enough to fully present such data depths? Here, the question must be raised how cooperation can be organized in the future, and whether it would not make absolute sense, at least in the case of the work and expression files, to obtain authority files for works and expressions as a research project via RISM.

And there are other areas that should be improved in the cooperation between library and science in the future: We noticed that library and source indexes are not structured enough. Almost all information important for research (e.g., information on scribes, reflections or references in the literature on dating, references to supplements) is described, if at all, in unstructured free-text fields.

When using MARC codes, for example, the codes for professions in the field of music are not differentiated enough, and the possibilities for ‘classification’, which are obligatory in the library field, are either very general or more exact than can be determined by specialists. Instead of a ‘classification’, researchers need the original genre designations in order to be able to describe the individual terms with sufficient precision.

However, many developments can also be highlighted positively: With musicconn.performance, a powerful platform for searching events is being created, to which we will contribute our events from the theatre programs. With IIIF technology, numerous rights problems regarding the use of digitized material are solved, and the GND is slowly opening up with regard to the joint development of authority data.

Conclusion

The data of the Court Theatre Project are to be considered an important contribution to the research of the German theatre landscape in the 19th century: They are open and freely available, clearly structured and comprehensible, because they are available in TEI or MEI 4.0, including approximately 900 score incipits, and are made available not only in the project’s web portal but also on Zenodo [1]. They are ready for evaluation, be it classical or digital, for musicological, theatrical, historical, or sociological research.

Works Cited


Beethoven in the House: Digital Studies of Domestic Music Arrangements

Kevin R. Page  
University of Oxford, United Kingdom  
kevin.page@oerc.ox.ac.uk

Johannes Kepper  
Paderborn University, Germany  
kepper@edirom.de

Christine Siegert  
Beethoven-Haus Bonn, Germany  
siegert@beethoven.de

Andrew Hankinson  
RISM Digital Center, Switzerland  
andrew.hankinson@rism.digital

David Lewis  
University of Oxford, United Kingdom  
david.lewis@oerc.ox.ac.uk

Abstract

Performance of music in the home was the means by which most works were received before the advent of audio recordings and broadcasts, yet the notation sources that form our primary record of this culture have not been the subject of comprehensive or methodical study. Choices made by arrangers adapting music for domestic consumption – of instrumentation, abbreviation, or simplification – reflect the musical life of the 19th century, and can inform our understanding alongside contemporary accounts such as newspapers, adverts, and diaries.

This position paper gives the background, motivation, and proposed approach of research currently being undertaken within the Beethoven in the House project.¹ This will include a study of Steiner editions of Beethoven’s 7th and 8th Symphonies and Wellingtons Sieg, making a detailed comparison between arrangements, systematically identifying a core common to multiple versions, and asking if this reflects the stated values of the publisher. A second survey will look for patterns across a larger sample of lesser-known and poorly catalogued scores, collating emergent indicators of arrangers’ motivations within a narrative of the domestic market – the music industry of its day. Both studies will innovate digital methods which characterise arrangements as music encodings, including ‘sparse’ approaches to notation and annotation.

1 Introduction – Domestic Music and Arrangements in the 19th Century

Domestic music-making was for centuries the principal way many people discovered and explored music. With no recordings, a piece such as Beethoven’s Große Fuge, which was performed publicly once just before 1850, could only be heard by those who bought and performed from the edition or from arrangements. Orchestral works such as Beethoven’s symphonies might only reach the larger concert venues; operatic works were also difficult to access, although extracts and reductions were commonly given in concert halls. Domestic performances allowed a more leisurely, detailed, and participatory exploration of the music. They also gave enthusiasts access to music that might not otherwise be played locally.

Concert pieces often required adaptation to accommodate the home environment and reductions for chamber ensembles, or for piano solo or duet, were common. Although some composers arranged their own music, most publishers would employ in-house arrangers to satisfy the huge demand for adapted versions of works originally created with the concert hall or opera house in mind. This transformation from public to private space brought other changes reflected in arrangements. Most obviously, amateurs replaced professional mu-

© Published in 2022 with Humanities Commons under a CC BY-NC-ND 4.0 (https://creativecommons.org/licenses/by-nc-nd/4.0/) license.
sicians. But in an age when instruments were perceived to be suitable for performers of a specific gender, new arrangements could also bring musical works to domestic performers of the other sex.

Despite its importance this part of musical life, especially concerning arrangements of instrumental music, is not well studied. The lower status this music has taken in historiography of the period has also meant little attention has been given to the arrangements themselves, and the music has received little attention in the literature. Notable exceptions are the articles by Christensen [2], Bashford [1], and the workshop presentation by Lewis [7]. The situation is significantly better for operatic arrangements (see, e.g., Christensen [3], Feder [5], Hinrichsen and Pietschmann [6], Pietschmann [10]; also Siegert [11] and [12]), but in comparison to the vast amount of material knowledge is remarkably scarce.

The quality of arrangements was clearly of great importance to composers, but presumably also to purchasers if the adverts by publishers and printed reviews in both national and music journals are to be believed. The presence of a named arranger on the front page of an edition was an attempted reassurance of high musical standards to the extent that the publishers Cranz used a pseudonym to give the appearance that all their arrangements were carried out by the ever-reliable G. W. Marks – in reality a group of house arrangers whose numbers included the young Brahms. Mozart planned to make his own wind arrangement of Die Entführung aus dem Serail, because no-one else should have “den Profit davon” (letter from 20 July 1782), while Beethoven rejected the arrangement of the Große Fuge op. 133 for four hand piano that his publisher Artaria had commissioned and made his own arrangement instead (op. 134). Mozart also referred to the aesthetic difficulties in arranging a piece, the balance of maintaining the effect of the music and making it suitable for the new scoring – central aspects that should be addressed in any study of the material.

Arrangements were marketed in terms of availability and price, but also quality and authenticity. Reviews and adverts for new editions give evidence for the importance of these values and of what constituted quality, which is largely spoken of as fidelity to the model, and clarity and preservation of lines. The publishing house of Sigmund Steiner and Tobias Haslinger is important with regard to both of these values, as it could trade on a (variably) close relationship with Beethoven himself [15] and made statements in adverts about their arrangements of his works that explicitly set them in contrast to other versions on musical criteria.

Another challenge to study this element of musical culture is the sheer volume of material, although the digitisation of Hofmeister’s catalogue of 1844 makes it easier to see the breadth of music available to amateur musicians. Hofmeister alone lists almost 9,000 piano duets, a vast number of which are reductions, with the most popular composers being Czerny followed by Beethoven. Famous orchestral works such as Beethoven’s symphonies or Weber’s overtures could have many arrangements, with instrumentations ranging from piano solo to flute and string sextet. Christensen [2] quotes a claim that, by 1871, there were around 60 different reductions of some of Haydn’s symphonies; and that in 1872: “The arrangements of Beethoven’s works already existing are well-nigh countless in number, and ‘the cry is still they come’”. With so many versions, so much music and, for the most part, quite nuanced musical differences in a text that is largely the same music in all cases, manual comparison is labour intensive and scales poorly for those who want to maintain an overview.

---

2 Strings were restricted to male players in many places for much of the 19th century; piano could be either, but frontispiece imagery and dedications strongly associate it with women.

3 Beethoven’s symphonies presented “for the first time in a truly artistic way, in regard to their sound quality and purity of setting [...] in which the features of voice leading [...] are here never obscured” (advert in the Leipziger Allgemeine Musikalische Zeitung 11, no. 20 [May 17, 1876], col. 320, quoted in Christensen [2, pp. 271–272]).

4 “Czerny packed both hands full, so that very often the possibility of making single tones and voices prominent ceases; [...] a continual screaming discord tortures the nerve of hearing” (Dwight’s Journal of Music 4, no. 6 [November 12, 1853], p. 41, quoted in Christensen [2, p. 270]).

5 Self arrangers include: Beethoven, Brahms, Mendelssohn, Schumann, Dvořák, and Tschaikovsky.


7 Monthly Musical Record 2, no. 10 (October 1, 1872), p. 152, quoted in Christensen [2, p. 259].
2 Project Approach

The *Beethoven in the House* project is currently exploring the issues introduced in the previous section. It is doing so through an interdisciplinary collaboration between musicologists and technologists, broadly organised into four areas of research: two musicology studies (described in sections 2.1 and 2.2, below), accompanied by two pieces of technological innovation (2.3 and 2.4) which take their requirements from, and will be evaluated through, the musicologists' studies. Taken together they present a new and novel integrated approach which will advance the field of digital musicology. In this section we describe the research proposed for each of these four areas, which are currently progressing through their initial stages of realisation within the project.

2.1 A Study of Beethoven Arrangements Using Digital Music Encodings

The original prints of Beethoven's symphonic *Wellingtons Sieg* as well as the 7th and 8th Symphonies were presented in various 'editions' (*Ausgaben*), as his publisher Steiner called them, distinguishing them from simple 'arrangements' (*Bearbeitungen*): for orchestra (score and parts), piano (two hands, four hands and two pianos), piano trio, string quintet, and wind music. They have, however, not yet been studied as a whole and will now be contextualised with Steiner's advertising strategies as well as the discourse linked to them, especially in letters and reviews. Through this example, we will be able to depict a more detailed picture of the musical life in Beethoven's Vienna drawing attention to an often-neglected musical practice. It might, moreover, help us to better understand the huge success of the three works in the years just before the Congress of Vienna.

This close study is investigating whether there are significant differences in the musical structure between the Steiner 'editions' and other arrangements (including those published in newspapers and collections), analysing the most interesting points of these works as well as of the arrangements. It will use a digital environment and the novel application of digital methods to undertake the exploration and structure the research output, evidencing findings for digital publication and dissemination. In doing so we aim to provide a complete, focussed, motivation and evaluation of the new digital methods developed across the project.

Digital approaches to musicology allow and urge us to reconsider the conceptual basis of our research. It might, for example, be a fruitful approach to rethink the genre of opera as a network structure with single pieces moving from one opera to another, works can be considered and, ideally, edited as the sum of their single and diverse manifestations.

Here we will consider the idea of a 'core', developed by the *Beethovens Werkstatt* project, which identifies and encodes that which can be found in multiple versions (in this case: arrangements) of the same work. We propose this investigation since Steiner used the term 'editions' in a completely non-hierarchical way; he gives no emphasis on the 'original' scorings of the works, so all 'editions' are obviously of the same value for him. Our encoding must therefore outline this core and should be located on the work level concerning the Functional Requirements for Bibliographic Records (FRBR) standard, an abstract entity that has an abstract core to which the concrete 'editions' are related.

In presenting the results of this study we will test what is possible to be encoded using MEI, and what has to be contextualised or presented in other ways. Points of special interest will be, for example, the initial fanfares, the 'moving' of the sound, and the quotations in *Wellingtons Sieg*, performance indications like 'pizzicato', pedal use, specific instrumental restrictions which aim to describe the characteristics of these arrangements. Concerning the central aspects of sound, especially in *Wellingtons Sieg*, it seems necessary to demonstrate the sounds; which makes clear that the study has to be presented in a digital way.

---

8 Nancy November has recently published a series of editions of Beethoven's symphonies, including some Steiner editions [9].
9 E.g. an arrangement of the second movement of the 7th Symphony for piano published in *The Harmonicon* 2, part 2 (1824), pp. 69–71 (see [4, p. 598 (vol. 1)]).
10 This was suggested in a paper by Andreas Münzmay and Christine Siegert presented at the congress of the *Arbeitsgemeinschaft für germanistische Edition* in Frankfurt in February 2018 [8]. In this context, the concept of authorship has also to be reconsidered. For a model of shared authorship concerning the genre of opera see Siegert [13].
11 A (german) description of the concept of a musical core (‘Satzkern’) can be found in the project’s glossary at *https://beethovens-werkstatt.de/glossary/satzkern/* (accessed January 12, 2022).
Comparing multiple arrangements of the same work will also help to identify different approaches and priorities that arrangers may have had, which in turn may help to better understand the audience or contexts for which a particular arrangement was made. Even without full encodings of all arrangements to be considered by the project, the concept of a work's core might be applied during the comparison, and in doing so the musicological study benefits from this new digital methodology.

2.2 A Study of Musicological Patterns in Domestic Arrangements

A domestic audience represented a complex market for publishers. Levels of musicianship could vary substantially, and the arranger would have weighed the requirements for idiomatic performance on new instrumentation with the use of the music as *aide memoire* or study score for its model. This balance will have involved conscious decisions, but these are offset by the pragmatic realities of a rush to publish a popular new work and the abilities of the arranger.

Drawing on work gathering contemporary descriptions of home music-making [1], along with published reviews and adverts for reductions of works, plus title pages and illustrations in the books themselves, we are exploring the relationship between the music of domestic arrangements on the one hand, and the market and the narratives around it on the other. This will be achieved through technological approaches which support and enhance the processes of finding, gathering, comparing, and annotating materials. Thus the musicological research will both be assisted by the digital methods and provide a case study of a rich humanistic investigation to direct the technical innovations. An important aspect of this exemplary role is the incomplete state of cataloguing and digitisation for much of the material, which forces an incremental, investigative approach on the researcher as in common with much research but not always reflected in digital tooling.

Whereas the previous study (section 2.1) is framed as a close reading of a smaller number of arrangements, this research area exemplifies a broader, shallower survey, aiming to investigate as-yet unexplored collections of domestic music arrangements. In undertaking this more speculative study we hope to make this little understood material more accessible for future scholarship.

We have begun to select and digitise pieces with multiple arrangements in the Bodleian Libraries and Beethoven-Haus collections, or arrangements elsewhere that are of specific interest. In each case, we will create full or partial transcriptions of the notation. Where possible, we will also acquire licence-free audio. With reference to contemporary descriptions, we will gather the material together and explore how the arrangements have been produced, how they differ from each other and the model, and speculate on why that is. This will take into account timbral aspects of reinstrumentation, player competence, preservation of voice leading, and other musical and practical factors. These observations will initially be structured around annotations made using the tools, models, and representation outlined below.

This objective complements the study of Steiner's 'editions', where the material has already been identified. Between these two objectives we aim to validate an end-to-end digital approach to musicology, both in the sourcing of materials for study and by the close analysis of those materials.

2.3 A Prototype Digital Environment for Musicological Exploration of Digital Material

Discovering, collating, and annotating music-related historical materials is a central element of many musicological investigations (Figure 1), including the two studies introduced above. Digitised materials may come in the form of images (whether facsimiles of pages of scores, texts and pictures, or photographs of objects), audio-visual files, or encoded texts and scores. Deploying, coordinating, and collaborating with these materials on a computer can be cumbersome, especially given limited screen space.

In support of our musicological research studies, we are devising a novel prototype digital environment to support quick, accessible, juxtaposition and combined annotation of materials, making efficient use of any catalogue and descriptive information as it is available – as opposed to making metadata improvement a precondition for research, valuable through it is. In a similar vein, it will support the addition of *partial* transcriptions of music and text as well as complete additions, allowing a thesis to be developed during the process of exploration without requiring a full transcribed encoding of notation.
Where possible, the materials being delivered will be coordinated automatically in ways that increase the likelihood that related parts are presented together by the system; where helpful in aiding such alignment and creating musicological annotations, computational analysis will be made available to the musicologist. We will primarily work to find instances of candidate materials held in libraries that use the International Image Interoperability Framework (IIIF) specifications, which can then be supplemented by music encodings. Annotations will always be saved, and organising and retrieving these will be supported.

2.4 Innovative Digital Annotation Methodologies Supporting Musicological Studies

An investigation of the musical world surrounding home music-making requires an approach that combines traditional archival and library work with digitally supported scholarship. Such work will also accumulate a secondary layer of note-taking and record keeping by the researcher, recording observations about and connections between different materials.

In conjunction with our musicological studies – which we are using to both motivate and evaluate our proposed digital environment – we are refining and expanding several methods to support the musicological annotation of domestic music collections from the Beethoven-Haus, Bodleian Libraries, and beyond. Each of these will provide a specific solution to a research need and, in doing so, potentially act as a methodological template for future practice in other areas of digital musicology.

We are creating the necessary encoding extensions to MEI so that an unchanged core can be marked as consistent between arrangements and editions, enabling comparison of differences; and for partial ‘sparse’ encoding of fragments during the gathering and relating of digital source materials as they are discovered and collated. MEI-based encoding practice will need to cover a range of situations, beginning from sparse encodings containing nothing more than information about the performing forces or numbers of measures, and ending with full and detailed encodings of printed copies with manuscript entries and corrections.

The combination of such diverse encodings in a single project is new and will require very careful modelling in order to allow consistent validation of data, a necessary precondition for reliable musicological use. We have begun this process by encoding selected arrangements nominated in the course of the two studies described above, and identifying commonalities in their relevant musicological ‘features of interest’. Taking these prototype models we will manually derive an ODD customisation, which will then be used to validate encodings of a wider range of arrangements associated with the studies. This will be an iterative process, with novel observations from newly modelled music documents resulting in changes to the ODD customisation, converging on a stable encoding which can be sustainably documented.

Beyond that which is solely encoded in MEI, we are working to identify and declare musicological concepts which bridge the varied resources from across our studies, enabling them to be consistently referenced in our scholarly findings, using Linked Data technologies to formalise an ontological model. By applying the same model to both of our focus studies (which are representative of different stages of research maturity – see 2.1 and 2.2), we will demonstrate a single model applied across the research lifecycle – from the resource finding and triaging activities exemplified in the second study, to the close study and analysis found in the first.

The most significant aspect of this contribution will be the encoding of a (rather than ‘the!’) high-level musicological framing, consistently applied to materials which themselves were not created or conceived as explicitly conforming to a common model or schema. We will realise this work through an extensible and self-describing
machine readable model according to Semantic Web and Linked Data principles: with semantic relationships and constraints implemented using the Web Ontology Language (OWL); and anchored to digital materials using Web Annotations. In this way we will connect evidential resources in a manner which can be incorporated consistently within future digital workflows, encompassing digital materials not only encoded using MEI, but also those with less detailed or currated records.

We have begun this work by identifying an initial set of digital resources necessary to progress our musicological studies and their narratives (sections 2.1 and 2.2), then creating a first draft model which can distinguish both the type of material (e.g. score, edition, advert, etc.) and granularity of association necessary to disambiguate scopes within that material (e.g. work, score, note). Next, we will elicit the cross-cutting musicological concepts required to formulate a single narrative incorporating all types of material from our studies. We anticipate these relationships may be indirect, i.e. intermediate abstract concepts may need to be instantiated to maintain a clear and consistent model. The method will be ‘MELD-compatible’, such that the annotations can be visualised using the Music Encoding and Linked Data framework [16].

3 Conclusion

This paper gives an introduction to the Beethoven in the House project, which encompasses two novel and complementary studies into domestic music arrangements of the 19th century, a digital research environment which will be co-developed alongside the studies, and the innovative application of digital musicology methods within this environment. Above, we have given the motivation and our proposed approach for our four interrelated areas of work, which are currently being undertaken within the research project. We look forward to reporting on our progress and results from each of these in future papers, so advancing the state of the art in digital methods. Working within and across each of these contexts, we hope to highlight the opportunities provided by such combinations of scholarship, technology, and collections.

Acknowledgements

Beethoven in the House is gratefully supported by a UK-Germany funding initiative: in the UK by the Arts and Humanities Research Council (AHRC), project reference AH/T01279X/1; and in Germany by the Deutsche Forschungsgemeinschaft (DFG), project reference 429039809.

We also thank project team members Christin Heitmann, Mark Saccomano, and Elisabete Shibata for their ongoing contributions to project discussion and research.

Works Cited


The PROFMUS Application: Development, Status, and Future Progress

Maria João Albuquerque
Universidade Nova de Lisboa
Faculdade de Ciências Sociais e Humanas, Portugal
mjdalb@fcsh.unl.pt

José Luís Borbinha
Universidade de Lisboa
Instituto Superior Técnico, Portugal
jlb@tecnico.ulisboa.pt

H. Sofia Pinto
Universidade de Lisboa
Instituto Superior Técnico, Portugal
hsofiapinto@gmail.com

Luís Filipe Santos
Universidade de Lisboa
Instituto Superior Técnico, Portugal
luis.96.santos@hotmail.com

Tiago Teixeira
Universidade de Lisboa
Instituto Superior Técnico, Portugal
tiagu.teixeira@hotmail.com

Abstract

PROFMUS is a collaborative project that aims to carry out the research and consolidation of information to support further research about the Portuguese musicians active in the period from 1750 to 1986. The information to be collected must include as many relevant attributes as possible, especially about their academic background, professional careers, and personal details. This project considers a large amount of data from a wide time-period, which means there will be various attributes for each object, which will evolve over time or differ from source to source. There is also an issue with the lack of uniformity of existing sources in multiple institutions, museums, archives, and databases, each with its own data scheme. Since PROFMUS has a long-term perspective, it does not try to create a uniform and prefixed scheme for the data to consolidate but accepts every different scheme and stores all data in a controlled knowledge base. We describe here an application for that purpose, using MediaWiki and Wikibase for storage, and a back-office specific application to manage and publish the data.

Introduction

The PROFMUS application aims to store information gathered about Portuguese musicians, works, professional careers, etc. The goal is to allow researchers to store, retrieve, and publish data in a flexible way on top of a knowledge base reusing open-source solutions. The target audience of this application will be the researchers affiliated with the research project PROFMUS, who will collect data on Portuguese musicians. It is expected that the volume of data in the database will have to cover a large time period. The data upload workflow is designed to be collaborative, with an administrator to validate the uploads made by the researchers. This administrator will be responsible for ensuring the quality, origin, and validity of the records. All versions of the entities will remain in the database with an indication of the source (project, third-party system, etc.) in order to allow tracing the sources of each piece of data.

There are very few similar Portuguese projects in musicology or in other Digital Humanities domains using this technology. The vast majority use relational databases. We know of two projects that use this technology to support their databases. In Portugal, the TechNet EMPIRE project¹ (Nova University of Lisbon) aims to “look to the agents of the colonial ‘dispositif’ experts and institutions – throughout time and space in order to understand how they created and shaped technoscience networks across the Portuguese Empire”. And in Spain,


© Published in 2022 with Humanities Commons under a CC BY-NC-ND 4.0 (https://creativecommons.org/licenses/by-nc-nd/4.0/) license.
The Ricardo Viñes + Wikibase Project [2] (University of Lleida) intends to trace the famous Spanish pianist's life using Linked Data.

PROFMUS may in the future aggregate, organize, interconnect, and contextualize authors within the scope of cultural heritage studies to support researchers and the general public alike.

1 Conceptual Solution

The PROFMUS application considers the following main requirements:

a. The application should be simple enough so that less technically skilled users can use it with no major difficulties.

b. Different data structures from diverse sources should all be supported. Through the meetings with the researchers, it was clear that Excel files were heavily used, so the system needed to support this kind of file.

c. Data collectors most of the time use forms to collect data, so the application should have support for some kind of forms that can be shared across users.

d. It should be possible to create subsets of data and share or publish them.

e. Reuse of information is important, so the application should be able to export data (items, properties, forms, etc.) so that it can be used in other applications or scenarios.

Wikibase empowers MediaWiki to store structured data into a data repository and access it [1]. Wikibase consists of two extensions, Wikibase Repository and Wikibase Client, which can be enabled either simultaneously or individually [3]. The Wikibase Repository extension turns a MediaWiki build into a repository of structured data, which can be stored and edited. Wikibase Client creates a client for a structured data repository allowing for data access and visualization. With Wikibase, it is possible to create a knowledge base about any subject and with concepts and properties that are more suitable to the respective domain. The Wikibase data model has entities as the most basic element. An entity can either be an item or a property. Items are the entity that represents anything that needs to be described. Every item has its web page, an identifier, a label (which is usually the name of the item), a description and aliases. Each item can have its own set of statements. Statements are pairs of properties and values. This is how facts about an item are described.

Properties have a structure similar to items. This includes a label, a description, aliases, and even statements. Additionally, properties have a data type associated with them. This means that each property can only accept values of one type. These types can be a string, an integer, coordinates, an object, etc. Statements can also include qualifiers and references. Qualifiers are pairs of properties and values that can add context and additional information to a statement. For example, if a statement says that a musician was married to a specific person, two qualifiers might be the date and place of marriage. The purpose of references is to indicate the source of the information in the statement. A reference can be an URL to an external website, a book reference, or even a Wikibase object.

For our purpose, two new concepts were introduced: views and forms. A view is a sub-collection of the knowledge base that consists of a set of items and properties. Views are created and managed by users and their purpose is to be published. Publishing a view means that the content of that view will be displayed on its web page, where items and properties can be viewed in a more visually pleasing way. Any file imported to the system can originate new items or a new form. The re-utilization of forms reduces the work needed to upload data into the system. These two concepts are managed in the PROFMUS application. This avoids future compatibility problems if, for instance, the MediaWiki deploy is updated to a newer version.
2 Architecture

The application consists of four different components (Figure 1). The Object Controller is a MediaWiki deploy that will manage and support the Object Repository. Additional information, such as users, views, forms, etc., will be stored in the Management Database. These three components will be coordinated by the Application Controller. The Application Controller was built using Django and contains all the logic of the application for both the back-end and front-end. This contains the interface users will mostly interact with. The application will process data from imported files, connect to MediaWiki and Wikibase through the use of bots\(^2\), and interact with the Management Database to manage its content.

The Management Database is an SQLite database connected to the Application Controller and serves as the storage for views, forms, users, usage logs, and to keep track of deleted items. This database is completely independent of both the Object Controller and the Object Repository.

The Object Controller corresponds to a MediaWiki deploy and is the place where most of the data, such as items, properties, and all the relationships between them, are stored. It also provides more granular management of the data, but it also adds more complexity to the operations.

---

\(^2\) A 'bot' – short for robot – is a software program that performs automated, repetitive, pre-defined tasks.
3 Use Cases for Data Management

Given the requirements and the knowledge acquired from the meetings with the researchers, the following set of use cases was created:

a. Create account and login: A user creates an account, logs in, and has access to objects in the system. At this point, only some of the functionalities will be available. Users will have read-only permissions until this is changed by an administrator.

b. Import item set: A user imports a spreadsheet file where each line represents an object and each column the values for a property. This also might create a new normalized form.

c. Export item set: A user selects a set of items to export, and the system creates a file with the objects selected and the properties that hold values about those objects.

d. Create normalized form: To facilitate the import of data, a normalized form is created by importing a file containing a template of the form to be used. This will be saved in the system and can be exported at any time by users with permissions to access it.

e. Create view: A user intends to create a collection of items, called a view, that can be managed and published. So, the user either selects the items or uploads a file with the items that will belong to the view, gives it a name, a description, and defines its privacy. A new view will be created with its own web page.

f. Export view: A user can export the items in a view to an Excel file. This file can then be used to edit the view or generate a new one.

g. Manage item: When items are stored in the knowledge base, they can then be managed by users that have permissions to do so. This includes:
   i. Delete items from the system, by selecting them from the list of items or importing a file with the QIDs of the items to be deleted.
   ii. Restore items from the deleted items list. Wikibase does not allow for items to be truly removed from the knowledge system. So, for example, if an item was deleted by mistake, it can easily be restored later.
   iii. Edit items already imported to the system. The changes made to the system are made using the interface provided by MediaWiki.

h. Manage permissions: To change the permissions of a specific user, an admin can choose the best fitting set of permissions that the user should have.

4 Retrieve Data

The most effective solution for extracting information would be through a query execution system directly in our repository, but the original capabilities and operations available through the API are limited. For operations to which the API cannot correspond, alternative solutions had to be developed [5].

A view is a subset of all the objects in the system, where each object is made up of a set of properties and their values. For example, we can consider as property “Date of Birth” and as its respective value a date that represents the date of birth of the corresponding object. Views can be created and managed by users of the system, plus they can be published meaning that each view will have its own website, which can be shared with anyone.

The creation of views, for publication, is done in another application also developed within the scope of this project, which uses its own SQLite database to store the necessary management data. It is also in this data management application that the ‘connection’ to the view visualization application is made. The Object Controller is created by an implementation of MediaWiki that will manage and support the Object Repository. All kinds of additional information such as the views elements or the aesthetic definitions of each view will be saved in an SQLite database. These elements are coordinated by the Application Controller. This controller is a Django application that connects to MediaWiki via HTTP and to the database via an internal API. The Application Controller is also responsible for the interface shown to the user through an Apache web server.

3 QID is a Unique ID that identifies the item.
The administrator has these options to manage the views:

- edit names, titles and subtitles;
- choose the properties to be displayed in the table;
- edit the display information order;
- choose the geographical properties to be projected on a map;
- choose the objects to be projected on the map;
- choose the color of the geographic marker corresponding to each object;
- choose the objects to be shown on the timeline.

For the construction of the final visualization, the Application Controller extracts from the database all the settings saved by the administrator and applies them to the base HTML page, which serves as the basis for all views. As a web framework, Django needs a convenient way to generate HTML code dynamically. The most common approach depends on templates. A template contains the static parts of the desired HTML output as well as some special syntax that describes how dynamic content will be inserted. A Django template is a text document or Python string marked using Django template language. Some constructs are recognized and interpreted by the template’s mechanism. The main ones are variables and tags. A model is rendered with a context in the form of a Python dictionary, which contains the information that will replace the variables present in the template with the values due. The first step taken by the application to build the interface is to extract an array from the database with the objects’ identifiers of the view in question. After that, the application makes an HTTP request to the MediaWiki API to receive a JSON file that contains all the properties and the respective values as well as qualifiers if they exist. This step is repeated for each object.

**Conclusion**

The PROFMUS application aims to be a platform to import, export, and publish data related to musicians. It was chosen for the project due to its scalability and the fact that successful projects, such as Wikidata, use the same technology. A file type widely used by researchers in this field, Excel, is converted into Wikibase items. This allows for a straightforward experience for the users while still having the flexibility of working with complex data.

There are some current limitations regarding the format of files to be imported, since it only accepts .xlsx files at the moment, and also some limitations on data editing. There were also some problems in the import process because the system had some limitations importing numbers. In this sense, after a complex analysis of the functional code responsible for importing the data and after several debugs, it was possible to make this process independent of the type of data imported. Consequently, it was possible to import a first block of the actual data provided. Despite these identified limitations, this system has been subject to an evaluation that has yielded positive results.

**Works Cited**


Objective: A database of Portuguese musicians active from 1750 to 1986.

About the Data: Data to be gathered after research in archives, museums, literature, etc., to identify the musicians and collect as many relevant attributes as possible (academic background, professional careers, family, events, time spans and geographic locations, etc.).

The challenges: How to manage data, representing facts in a wide period, to be collected during several years, where each new data source might bring not only new objects but also new kinds of relevant attributes? How also to do that if long-term technological support might not be assured?

Implications: Hard do design a definitive data schema and stable software that will not result a constraint at some future moment in time...

The solution: A flexible application using MediaWiki and Wikibase to store the data as an ontology, and a back-office simple application to import, export and publish the same data according to flexible schemas, defined in spreadsheets (thus, not requiring IT expertise)...
The OpenScore Lieder Corpus

MEC 2021 BEST POSTER AWARD

Mark Robert Haigh Gotham
Technische Universität Dortmund, Germany
mark.gotham@tu-dortmund.de

Peter Jonas
MuseScore / OpenScore, UK
peter@musescore.com

Abstract

The OpenScore Lieder Corpus is a collection of over 1,200 nineteenth century songs encoded by a dedicated team of mostly volunteers over several years. Having reported on the initial phase, motivations, design, and community-oriented aspects of the project before [3], we present here the first, stable, large-scale release of this corpus specifically designed for MIR researchers, complete with comprehensive, structured, linked metadata. The corpus continues to be available under the open CC0 licence and represents a compelling dataset for a range of MIR tasks, not least given its unusual balance of large-scale with high-quality encoding, and of diversity (songs by over 100 composers, from many countries, and in a range of languages) with unity (centred on the nineteenth-century lieder tradition).

Introduction and History

The OpenScore Lieder Corpus project began as a pilot initiative in 2018 to explore the potential for more coordination of the wide-spread enthusiasm for encoding public domain scores [3]. We sought to bring encoders together to the mutual benefit of all musicians, including professionals, amateurs, and academics alike. Since that first phase in 2018, the corpus has continued to expand beyond what we could have expected at that time. It now offers over 1,200 songs, each of which has been encoded by one contributor, reviewed by another, and released under the maximally permissive CC0 licence to enable uses of all kinds. Given that there is still a relative paucity of corpora that combine scale, quality, and truly open licence, we consider that ‘quite enough lieder for now’ and plan to start redirecting our efforts to other repertoires. As such, the time had come to prepare the lieder corpus as it stands into a dataset ready for use by the MIR community. This paper serves to announce the first public-facing version of that dataset, now freely available at https://github.com/OpenScore/Lieder. This repository explicitly serves corpus study: It is downloadable at once (unlike the collection on musescore.com which it mirrors), and comes complete with curated, linked metadata as described below.

1 The Corpus in Numbers: Centres and Outliers

In this overview of the corpus, we seek to give an overall sense of its contents and to identify the attributes of relatively ‘typical’ entries, but also to draw special attention to some notable outliers which can be highly valuable for comparative MIR tasks like clustering by style.

1.1 Composers, Works, Length, Sets

Since the beginning of this project, we have sought to represent a wide range of composers and works, including lesser known and never-published pieces alongside the famous cycles of Franz Schubert and others. In some cases, this involved a modest amount of research, tracking down hard-to-access publications and working in tandem with IMSLP to develop their holdings alongside ours (and providing direct links to the corresponding IMSLP edition for every song). In total, the corpus now comprises:

- 1,222 individual songs,
- grouped in 235 sets of song ‘cycles’ or equivalent (discussed further, below),
- by 105 composers of which 59 are female and 46 male.

© Published in 2022 with Humanities Commons under a CC BY-NC-ND 4.0 (https://creativecommons.org/licenses/by-nc-nd/4.0/) license.
Naturally, there is variation in the respective contribution from these composers. This ranges from Johannes Brahms with the most (100 songs) to some composers represented by a single entry. Other metrics show a similar but slightly different balance. For instance, while the songs are generally of around 2–4 pages and 50 measures in length, there is variation here too, both systematic and otherwise. For instance, most of the texts set in these songs consist of several verses in the same accentual pattern. This leads to an important kind of systematic variation in the (apparent) length of songs, as composers choose between either a strictly strophic setting (with the music set out once to be repeated for each verse) or something more through-composed. For instance, many of Louise Reichardt's songs fall into the former category and can be as short as 8 measures in length, with many verse repetitions as there are measures of music. Fanny Hensel's songs by contrast often exhibit a reprise of material for subsequent verses with intriguing changes to the details of harmony and/or texture. For instance, see “Verlust” (3 Lieder, No. 2) for a wonderful example that deftly exploits the harmonic ambiguities set up the first time around.

Beyond this, we include some more extreme cases, notably Erik Satie's Socrate, which the composer describes as a 'symphonic drama in three parts'. While the overall work's length is comparable to some of the longer cycles, the division into fewer, larger parts is perhaps one of the 'symphonic' aspects of this piece, and the dialogue between characters makes it arguably more of an opera than a song cycle (though it is commonly performed by one singer).

Finally, most of the songs represented here have been published or otherwise gathered into a set (often called a 'cycle', German: 'Kreis' or 'Zyklus') that is typically performed as a whole in the same way as any other multi-movement work like a sonata or symphony. That said, the corpus also includes many 'one-off' songs that are not grouped in this way. For convenience, we collect those songs into another nominal set with one such (unnamed) collection per composer.

Once again, the number of songs in each of these sets (real and nominal) varies widely. The mode value is 6, and the overall range is from 1 to 30. Among the large 'sets', we have both real collections (such as Corona Schröter's 25 Lieder) and also nominal groupings: the largest 'set' is one such convenience collection of Cécile Chaminade's songs, all of which stand alone.

1.2 Time, Geography, Language

Temporally, the works span what is sometimes called the 'long' nineteenth century, broadly meaning the period from the French revolution to the first world war (1789–1914). That said, public domain status is the more meaningful cut off for our purposes and so we include some later works as long as the work was published (or otherwise released), and the creators (both composer and poet) died early enough to meet public domain criteria (for instance, in the US and the EU respectively in these cases).

Most of the songs are in English, French, or German, with the poet and composer being native speakers of those languages. More accurately, the song language typically reflects the nationality of the poet (almost all cases) and often but not always the composer. For instance, the American composer Amy Beach is represented by her 3 Shakespeare Songs (Op. 37) and 3 Browning Songs (Op. 44) which are both in her native English, but also by her 4 Songs (Op. 51) which set German and French texts. Other languages represented more occasionally include Latin (Schubert's Ave Maria), Swedish (several songs by Helena Munktell and Laura Netzel), Italian (Gabrielle Ferrari), Hawaiian (Liliuokalani's Aloha Oe), and Brazilian Portuguese (Chquinha Gonzaga).

1.3 Style (or 'What Counts'?)

Given the above, we consider this corpus to represent a useful and interesting balance between contiguity and diversity of style. There is certainly a 'core' to the collection (of extended common practice Western tonal music), but a range both within and beyond that. Among the songs that might be considered outliers for strictly musical reasons (i.e. not considerations discussed above) we are pleased to include Webern's Op. 3 and Op. 4 (ten songs, less than 1% of the corpus). While some studies will need to exclude these songs (and that is easy enough to do), many studies will benefit from having such outliers, for instance as a basic check that stylistic similarity metrics are broadly working as expected before a closer look at the details: for instance, if the system can't tell Webern from Weber, then it is unlikely to provide insight into the differences between Robert and Clara Schumann.
Outliers help us to confront our curatorial definitions of what to include, and as with any repertoire-specific undertaking, this corpus has faced some difficult decisions about ‘what counts’. Clearly, lieder doesn’t strictly cover everything here. We’ve included French mélodie from the start and now include as many English-language songs along with a few other languages. Broadly we are using the term ‘lieder’ as shorthand for ‘art song’, but then, what is an ‘art song’ exactly? Basically, we take that term to mean a piece for one voice (or occasionally more voices) with piano, written down by its creator in Western classical notation. That excludes folk songs (though many ‘art’ songs include transcriptions of folk melodies) and also jazz lead sheets (which represent a different musical and notational tradition). We recognize that the definition is both too narrow and too vague at the same time, though we do not see any clearly better alternative. Publication is helpful for us administratively in terms of keeping track of what we have, but it doesn’t contribute anything to the ‘art song’ definition and we are proud to include lesser-known, never-published works. In summary, the collection includes a range of songs, both short and long, from trivially simple ditties to ferociously complex utterances.

2 Data and Directory

Apart from ease of download at scale, the corpus mirror offers a slightly expanded set of files and metadata in a carefully constructed directory structure specifically to assist MIR research.

2.1 Scores

The corpus centres on a subfolder called ‘scores’ with files arranged in the structure: <composer>/<set>/<song>. Specifically,

- <composer> gives the composer’s name in the form ‘Last_First_Second...’.
- <set> provides the name of the extended work that the song belongs to, if any. Stand-alone songs go in a single set called _ (a single underscore).
- <song> is the name of the song, including a possible prefix. Prefixes (such as 1, 2, 3a, 3b, etc.) are added to songs that have a defined order within the set. Prefixes are zero-padded (01, 02, ..., 09, 10, 11, etc.) where necessary to ensure the correct sort order.

Score files within each song directory are named in the format lc<id>.<fmt> where lc stands simply for ‘Lieder Corpus’, <fmt> is the file extension (.mscx for MuseScore) and the <id> is a unique ID number we assigned to each song to help identification. That ID also provides a direct link to the public-facing version of the score as hosted on MuseScore.com via either of:

- https://musescore.com/openscore-lieder-corpus/scores/<id>
- or equivalently: https://musescore.com/score/<id>

MuseScore files are convertible to other formats (MusicXML, MIDI, PDF) individually or in bulk using a plugin in the app or via the command line. We provide a contents file (corpus_conversion.json) in the format required for the command line option (run “mscore -j corpus_conversion.json”) and also a script for updating this file (corpus_conversion.py). We decided against including multiple copies of the files in the repository given this ease of conversion and a desire to keep the repository containing this corpus to a manageable size. We do include some duplication of data, however, in that we provide separate lyric files alongside the scores. These text files are automatically extracted from the score to guarantee agreement (see lieder.net for a sense of how prevalent version variants are in these songs) and to facilitate text-based studies as well as cross-referencing with other datasets.

We consider the score and lyric files to cover the essential, core provision and we elect to avoid overloading this repository with anything else. Some users may wish to consult another repository – the ‘When in Rome’ corpus1 – which includes another mirror of the score collection in the context of providing harmonic analyses for many of the songs (as well for other corpora). ‘When in Rome’ also provides a set of ‘slices’ files that repre-

---

sent the pitch and rhythmic-metrical content of the song in a tabular format with a new entry for every change of pitch, given in the form of vertical ‘cross-sections’ of the score. The score-to-slice conversions are produced with code from [2] building on the ‘chordify’ function in music21 [1] with all of this modelled on the design of the Yale–Classical Archives Corpus2 [5].

2.2 Data

The data reported here is stored in a dedicated folder (Data/), separate from the corpus itself (Scores/), and encoded in both TSV format and StrictYAML: a simplified variant of YAML that preserves the order of keys within object mappings. There are separate files for composers, sets, and songs with each providing a structured and easily retrieved representation of the relevant metadata. Each file organises its entries with a unique, numerical identifier. The composers file records entries for the:

- path: effectively providing the composer’s name as given in the first level of the ‘songs’ directory structure described above;
- name: the ‘correct’ way round (first name before last) without commas or underscores;
- gender: Male or Female, for keeping track of the corpus coverage as discussed above;
- born: composer birth year, where known;
- died: composer death year, where known;
- desc: a short description such as ‘US pianist, singer and composer’;
- link: an URL to the composer within http://musescore.com/openscore-lieder-corpus/;
- wikidata: a reference for the composer on wikidata.org (e.g. Q60159846),
- wikipedia: a link to the composer’s Wikipedia page, in English where possible, otherwise in whatever language is available;
- imslp: the composer’s ‘Category’ page on imslp.org;
- image: an URL providing an image of the composer, where publicly available.

Values are simply left blank where they are not available (such as the image) or not known (such as birth year). The sets document follows a similar format, now including a cross-reference to the composer_id as used and recorded in the composers file with the path naturally now extending to the format <composer>/<set>/. Likewise, songs extends to the full <composer>/*set*/<song> path, includes cross-reference entries for both the set and the composer IDs, and adds an IMSLP edition number (like #530500). Including the IMSLP number at the song- rather than set-level involves a certain amount of duplication, though it is preferable for consistency overall given that it is clearly necessary for those stand-alone songs without composer-assigned sets. Finally, the corpus file provides some basic overall statistics for the corpus, including the total number of songs, sets, and composers.

Notable here are the external links (to IMSLP, Wikipedia, and Wikidata) which endow the corpus with a reasonable degree of Linked Open Data (LOD) relative to the existing provision elsewhere. This is an area we would like to see expand. For instance, Wikidata’s coverage was relatively strong for composers, so we were able to adopt the reference codes from there as our own (local) ID, only needing to add relatively few new pages. The same cannot be said of poets/lyricist, and certainly not for the sets, or songs. We include a URL for the poet/lyricist where available, but this is an area that needs further work both within the corpus and without.

Conclusion and Outlook for Research

We consider this a highly useful dataset for a range of tasks within MIR and beyond. There are several work-in-progress projects currently using this corpus, not least of which is the parallel corpus of harmonic analyses mentioned above. Among the many other research possibilities, we see particularly strong potential for studies investigating:

- why composers may have chosen to set these texts and not others, for instance through cross-reference with a large text corpus like the Deutsches Lyrik Korpus;[4];
- the correspondences (or otherwise) between text and music;
- prospects for using this corpus’ clear and explicitly identified IMSLP source editions to test, develop, and evaluate OMR. For instance, it would be interesting to OMR each edition of the work in question, test the similarity of those OMR results against our transcription, and see how often the OMR technology (and similarity metrics) are robust enough to positively identify the source used.

While our focus will now move on to applying the systematic approach we have gradually developed here to other repertoires, we will continue to welcome contributions to the lieder corpus specifically and will review them as time allows. Naturally, we also welcome suggestions and pull requests from the MIR community for improvements to the provision as it stands and are open to suggestions for collaborations in developing and/ or using the corpus.

Acknowledgements and Contributors

We wish to thank all who have contributed to making this collection what is. Thank you to all the transcribers (individually credited on the musescore.com webpage for each song) and the core team of reviewers (credited in addition to the transcribers on our coordination spreadsheet). Thank you especially to the indefatigable Dan Rootham for extraordinary efforts across the board since the beginning. While contributors have mostly given their time and efforts as volunteers, we also very gratefully acknowledge in-kind support from musescore.com (pro accounts membership to contributors) and funding from the University of Cambridge's 'Arts and Humanities Impact Fund' and the Central New York Humanities Corridor.

Works Cited


---

7 https://docs.google.com/spreadsheets/u/1/d/1jec8rYm3Z7PqnxmFWztfqLAV0NVYwn2D5-umTGeatiU/htmlview# (accessed January 12, 2022).
OpenScore Lieder Corpus

- 1,222 songs,
- 235 sets or equivalent,
- 105 composers (59 female; 46 male)

Community website, scores playable online (+ download one at a time):

musescore.com/openscore-lieder-corpus/

Corpus mirror (download all at once):
- Plus lyrics and metadata (e.g. below)
- Support for batch conversion
github.com/OpenScore/Lieder

Franz von Schober

An die Musik
D.547

Frans Schubert

235699:
path: Beach,_Amy
name: Amy Beach
gender: Female
born: 1867
died: 1944
desc: American composer and pianist
sets: 3
scores: 10
link: https://musescore.com/openscore-lieder-corpus/235699
wikidata: Q235699
wikipedia: https://en.wikipedia.org/wiki/Amy_Beach
imslp: https://imslp.org/wiki/Category:Beach
image: https://upload.wikimedia.org/wikipedia/
MEI Meets NFDI4Culture

Kristina Richts-Matthaei
Paderborn University, Germany
kristina.richts@uni-paderborn.de

Martin Albrecht-Hohmaier
Paderborn University, Germany
martin.albrecht.hohmaier@uni-paderborn.de

Daniel Röwenstrunk
Paderborn University, Germany
roewenstrunk@uni-paderborn.de

Andreas Münzmay
Paderborn University, Germany
andreas.muenzmay@uni-paderborn.de

Abstract

NFDI4Culture is the consortium for research data on material and immaterial cultural heritage and offers a user-centered and research-led infrastructure. By focusing on the digital capture as well as data-based research of cultural assets, NFDI4Culture brings together different disciplines in their research interests but also in terms of their infrastructure needs. The consortium has been funded since October 2020 for an initial period of 5 years. Paderborn University is connected to NFDI4Culture via the Center for Music, Edition, Media (ZenMEM), which bundles activities in the fields of digital musicology, music and film informatics, media science, media technologies, and in several areas of computer science with a special focus on Digital Music Edition and Digital Musicology. With its many projects, ZenMEM contributes a lot to the further development of MEI. MEI will therefore also have a special place in NFDI4Culture. In addition to general community aspects and enhancements of the format, activities will focus on the development of best practice recommendations and training materials, the further development of MEI-related tools, such as MerMEId and the MEIGarage, and concepts for data quality in MEI, but also on the improvement of standard data for musical works.

The poster will show the organizational structure of NFDI4Culture and its representation at Paderborn University, its interaction and networking with national and international infrastructures, combined with several examples where MEI is involved.

Introduction

The National Research Data Infrastructure (NFDI)¹ is a digital, distributed infrastructure that is currently under construction and will offer services and advice to the scientific community in Germany on all aspects of research data management. Its aim is to provide the German science system with a “national collaborative network that grows over time and is composed of various specialised nodes” [2, p. 2], taking the form of, for instance, services and advisory offers for research data management. Even though the focus of the NFDI is in Germany, the NFDI is integrated into international contexts and structures and will develop and provide the services, tools, and materials in international cooperation. The promoted consortia shall:

- establish rules for the standardized handling of data in close feedback with the respective specialist community;
- develop cross-disciplinary metadata standards;
- develop reliable and interoperable measures for data management and a range of services tailored to the requirements of the subject community;
- increase the reusability of already existing data, also across disciplinary boundaries;
- connect and network with partners in foreign scientific and infrastructure systems who have expertise in the area of research data management; and
- collaborate in the development and establishment of generic, cross-consortium services and standards for research data management.


© Published in 2022 with Humanities Commons under a CC BY-NC-ND 4.0 (https://creativecommons.org/licenses/by-nc-nd/4.0/) license.
In three successive rounds of calls for proposals, consortia applications are to be submitted and a total of up to 30 consortia funded. These will then jointly form the National Research Data Infrastructure [1, §6]. NFDI4Culture is one of 9 consortia that succeeded in the first round of bidding and was successfully launched in October 2020. In NFDI4Culture, researchers from the fields of musicology, art history, architecture, theatre, dance, film, and media studies work together.

Over the last 15 years, the guidelines and the encoding format of the Music Encoding Initiative (MEI) have become the de facto standard in the German academic landscape for representations of musical sources, but also for the description of source-related documents and the interlinking and preservation of music-related knowledge. For this reason, the Music Encoding Initiative is widely represented in the NFDI4Culture consortium, both as a community and with its guidelines and tools.

1 The NFDI4Culture Consortium

The NFDI4Culture consortium for research data on material and immaterial cultural heritage focuses on the interdisciplinary discoverability and accessibility as well as the long-term preservation and continuous maintenance of the very heterogeneous data from the entire field of culture and the associated humanities disciplines. This includes digital 2D copies of paintings, photographs, and drawings as well as digital 3D models of culturally and historically significant buildings, monuments or audiovisual data of music, film, and stage performances. NFDI4Culture cooperates with national, European, and international infrastructures and is compatible with them. The structure of NFDI4Culture, which is closely linked to the academic community, is intended to make existing and new services accessible to the broad research community.

The long-term preservation and availability of these research data in accordance with FAIR principles2 is of central importance, not only for future scientific success in the humanities, but also for the cultural self-image of individuals and society as a whole. So far, there is no coordinated initiative for professional research data management at the national level in this subject domain. NFDI4Culture aims to create a user-oriented, research-led infrastructure that covers a broad spectrum from musicology, art history, and architecture to theatre, dance, film, and media studies. To this end, partners from research, memory institutions, professional associations, and infrastructural institutions of various sponsorships from the federal government, the states, local authorities, and the private sector have come together (Figure 1).

![Figure 1: The NFDI4Culture Community.](image)

The consortium consists of a geographically, thematically, and institutionally balanced network of nine co-applicants and more than 60 participants. It aims to ideally represent the broad spectrum of different actors in the cultural heritage domain. The co-applicants comprise four universities (Cologne, Heidelberg, Marburg, Paderborn), three infrastructure institutions (FIZ Karlsruhe, TIB Hannover, SLUB Dresden), Germany’s largest institution in the GLAM sector (Stiftung Preußischer Kulturbesitz), and the Academy of Sciences and Literature, Mainz. This group is joined by 11 academic societies each representing one of the research domains that together make up the community of interest.

---

Paderborn University is connected to NFDI4Culture via the Center for Music, Edition, Media (ZenMEM), which bundles activities in the fields of digital musicology, music and film informatics, media science, media technologies, and in several areas of computer science with a special focus on Digital Music Edition and Digital Musicology.

2 MEI Relevant Topics in NFDI4Culture

The Music Encoding Initiative plays an essential role not only in the context of musicology. In theatre and dance studies and, for example, film studies, connections can also be created via musical content and data. MEI also plays a key role in these contexts, bringing together experts from different subject areas. These researchers work together with librarians, software developers, and infrastructure providers to further develop the MEI encoding format in close collaboration with the MEI community, to discuss questions of the quality of MEI-encoded data, to develop tools and services around musical data, and to improve knowledge and skills in dealing with music encoding in general with the help of best practice guidelines and teaching & training offers.

Supporting Community Activities

One of the key objectives of NFDI4Culture is to support communities that are relevant in the context of material and immaterial cultural heritage data. In this context, the consortium regards itself as a promoter, trying to connect these communities and to give them a platform for exchange with each other. In a forum on data standards and data quality, NFDI4Culture would like to bring experts and communities together and discuss in regular workshops especially those issues that cannot be solved within just one community.

By advising and, in some cases, directly supporting projects, NFDI4Culture can and will also promote more far-reaching developments in the context of the MEI.

Additions and Extensions to the Encoding Format

NFDI4Culture is aware of a continuous adaptation of the encoding format of the MEI. Various extensions to the data format resulting from the needs of the researchers will be designed, discussed in the community, developed in workshops, documented, and finally implemented. This includes, for example, the question of standardized metadata for works and the use of authority data. This issue is already being addressed within the Music Encoding Initiative and can now be linked via NFDI4Culture with activities in other research subjects and communities and thus be integrated, for instance, into the context of the GND4C of the German National Library.

A second topic for connecting communities and activities in research projects is the integration and synchronization of audio-visual media in MEI. In the context of popular and electronic music, the question of suitable encoding formats arises in order to be able to describe hardware and instruments with adequate metadata on the one hand, and to be able to semantically represent non-notated music on the other. In addition, the linking of notated music with semantically described audio and/or video data, for example in the context of film music, is a desideratum that will be addressed in NFDI4Culture, too.

Aspects, Tools, and Criteria for Quality of Encoded Music Data

Since the creation of digital corpora with encoded music (among other data concerning tangible and intangible cultural assets) will be a central task of science and memory institutions for the next few years, the question of the quality of the encoded data is becoming increasingly important. NFDI4Culture aims to develop criteria and tools to simplify the process of producing high-quality data. What aspects of encoding music are important for the quality of the data? NFDI4Culture will take a wide range of perspectives to approach this question. In addition to the constant examination of the data model and its description, this also includes, for example, the semantic validation of the codings with the help of scripts that are built into MEIGarage, as well as coordina-

tion processes in the field of metadata, provision of encoding guidelines and schemas together with data, clear licensing of the corpora, persistent and unique identifiers, and version control of digital objects.

**Further Development of Tools and Services Around MEI**

The MEI community has produced essential tools and services in the context of various projects over the last two decades, some of which will be supported and further developed in the context of NFDI4Culture. One of these tools is the Metadata Editor and Repository for MEI Data (MerMEId), which was originally developed for the collection and publication of a catalogue of works by Carl Nielsen at the Danish Centre for Music Editing and has since been reused by several projects worldwide. After the funding for MerMEId expired, a community has emerged – due to the still ongoing demand in research projects –, which is now further developing the software and optimizing it for subsequent use. NFDI4Culture participates in the work of this community and supports the development and provision of the software with working time and infrastructure.

In addition to the further development and expansion of the transformation and validation platform MEI Garage, there are also plans concerning the enhancement of, for instance, deep learning methods to recognize measures on images of scores and store the coordinates in MEI's facsimile tree.

**Best Practice Guidelines and Teaching & Training Offers**

Probably the most important goal in NFDI4Culture is to teach competencies and skills in dealing with digital data and methods. For this purpose, the Cultural Research Data Academy (CRDA) was established, which aims to transfer knowledge to the disciplinary communities with the help of best practice recommendations, workshops, teaching materials, summer schools, and other participatory formats such as code sprints, book sprints, or hackathons. In this way, NFDI4Culture hopes to help increase code literacy in our communities, including the area of GLAM institutions.

NFDI4Culture will establish a communication platform that enables and facilitates access to knowledge, and will function as sort of a one-stop shop for researchers, providing support through various help desks and advice agencies. Tools for automatically enriching musical data with annotations, or licensing issues are just two examples from the comprehensive range of topics dealt with in NFDI4Culture.

**Works Cited**


---

MEI meets NFDI4Culture

NFDI4Culture is the consortium for research data on material and immaterial cultural heritage in the German National Research Data Infrastructure program (NFDI). NFDI4Culture will provide knowledge, methods, tools and infrastructure throughout the research data lifecycle, from data capture and enrichment, to standards and data quality, tools and services, publication and preservation all the way to education and training.

Supporting community activities

NFDI4Culture sees itself as a promoter for connecting communities and is giving communities a platform for overarching topics.

The needs of scholars for MEI structures to encode audio- and video-based data, and to incorporate additional metadata for electro-acoustic music and electronic instruments, are possible topics for funded workshops, furthering collaboration between communities.

Aspects, tools, and criteria for quality of encoded music data

Quality of encoded music is a central issue within NFDI4Culture – not least for the development of repositories that meet scholarly standards.

In addition to defining which aspects are relevant in the context of encodings of musical content, the consortium aims to design methods and tools to test and enhance the quality of encoded music.

Additions and extensions to the encoding format

Regarding the extension and linking of the MEI format, NFDI4Culture will provide bridges between communities and link experts together.

Contexts which are already in progress are, for example: standardised metadata for works using authority data from the German National Library, and a DFG viewer for musical resources that combines MODS and MEI in METS containers.

Further development of tools and services around MEI

NFDI4Culture will provide both financial and human resources support for the further development of tools and services around MEI.

Support for the further development of MEI tools and services ranges from accompanying the transition of MerMeId from an in-house effort to a community development model, to the extension of MEI/Garage to completely new developments from the communities.

Best practice guidelines and teaching and training offers

The Cultural Research Data Academy (CRDA) aims to teach competencies and skills in dealing with digital data and methods – not only but also essentially in the field of digital musicology.

Included will be best practice recommendations, workshops, teaching materials, and summer schools, as well as more participatory formats, such as code sprints, book sprints and hackathons.

Dr. Kristina Richts-Matthaei
krystina.richts@uni-paderborn.de
https://orcid.org/0000-0001-8569-1995

Prof. Dr. Andreas Münzmay
andreas.muenzmay@uni-paderborn.de
https://orcid.org/0000-0002-8373-4055

This poster is available under CC BY-SA 4.0, https://creativecommons.org/licenses/by-sa/4.0/
Towards a Unified Model of Chords in Western Harmony

Johannes Hentschel  
École Polytechnique Fédérale de Lausanne, CH  
johannes.hentschel@epfl.ch

Fabian C. Moss  
École Polytechnique Fédérale de Lausanne, CH  
fabian.moss@epfl.ch

Andrew McLeod  
École Polytechnique Fédérale de Lausanne, CH  
andrew.mcleod@epfl.ch

Markus Neuwirth  
Anton Bruckner University Linz, Austria  
markus.neuwirth@bruckneruni.at

Martin Rohrmeier  
École Polytechnique Fédérale de Lausanne, CH  
martin.rohrmeier@epfl.ch

Abstract

Chord-based harmony is an important aspect of many types of Western music, across genres, regions, and historical eras. However, the consistent representation and comparison of harmony across a wide range of styles (e.g., classical music, Jazz, Rock, or Pop) is a challenging task. Moreover, even within a single musical style, multiple theories of harmony exist, each relying on its own (possibly implicit) assumptions and leading to harmonic analyses with a distinct focus (e.g., on the root of a chord vs. its bass note) or representation (e.g., spelled vs. enharmonic pitch classes). Cross-stylistic and cross-theory comparisons are therefore even more difficult, particularly in a large-scale computational setting that requires a common overarching representation. To address these problems, we propose a model which allows for the representation of chords at multiple levels of abstraction: from chord realizations on the score level (if available), to pitch-class collections (including a potential application of different equivalences, such as enharmonic or octave equivalence), to pitch- and chord-level functions and higher-order abstractions. Importantly, our proposed model is also well-defined for theories which do not specify information at each level of abstraction (e.g., some theories make no claims about harmonic function), representing only those harmonic properties that are explicitly included and inducing others where possible (e.g., deriving scale degrees from root and key information). Our model thus represents an important step towards a unified representation of harmony and its various applications.

Introduction

Harmony constitutes an essential aspect of many Western musical styles [1, 6, 12, 18, 30, 32, 34]. There are a large number of harmonic annotation systems in music theory and analysis, including absolute chord labels, Roman numerals, Riemannian functional symbols [16, 23], pitch-class sets [7, 15, 28], and Tonfeld labels [8, 21]. Such annotation systems provide sets of symbols that assign specific properties to a segment of music and relate that segment to an underlying harmonic theory [9]. Harmonic labels can be assigned manually or algorithmically, and the systems make different assumptions about harmonic and non-harmonic notes, underlying tonality, the level of abstraction, and the style in focus. In recent years, several formal annotation standards were proposed for digital corpus research and Music Information Retrieval (MIR) [2, 3, 4, 11, 14, 17, 19, 20, 24, 25, 26, 27, 31, 33, 35, 36] (see also [10] for a recent overview and discussion). These standards were designed for different purposes, or are tailored to different styles of music, and therefore encode different sets of features (e.g., a standard for encoding chord progressions in rock music would reasonably not include the specificity of one designed for functional harmony annotations or chord extensions found in jazz). Despite this diversity, however, many systems overlap in a substantial number of the features they express.
For our purpose of an overarching harmonic notation and encoding system, as well as for the practical sake of musicological cross-style or diachronic analyses and comparisons, it is necessary to have a unified representation for diverse harmonic practices. Along these lines, the most relevant standard is the absolute chord syntax proposed by [11], which has been widely adopted in MIR. Here, roots are explicitly stated as (spelled) pitch classes, and the bass and the upper interval structure of the chord are given as (specific) intervals relative to the root – e.g., a Cmin7/Eb chord is represented as “C: (b3, 5, b7) / b3”. They are thus independent of underlying key or scale contexts. This annotation standard forms the basis of several formal ontologies [22, 29], which represent chords as graphs and most closely resemble our general proposal here. The objective of this paper is to propose a unified representation of chords for the comparative purpose outlined above, making it possible to characterize, query, and translate different features across standards.

1 Problem Setting

The assumptions for distinct systems of harmonic analysis define the problem setting for our unified chord model. Figure 1 shows a modern engraving of Arcangelo Corelli’s Sonata a tre, Op. 1, no. 8, “Largo”, mm. 7–12 (1681 princeps edition), along with several harmonic annotation systems underneath. These include figured bass (taken from the print) with bass degrees [13], Roman numerals, Riemannian function symbols [5], and absolute chords. Each of these annotation systems expresses harmonic phenomena from its own perspective, encoding some features explicitly, others implicitly, and others not at all. A universal chord model must be able to encompass all of these systems and more.

Another challenge for an overarching model are chord annotations stemming from non-diatonic or non-triadic musical contexts, or from settings where diatonic chords cannot be labeled other than enharmonically (e.g., in a MIDI file). Figure 2 shows an example of a non-triadic chord from Sofia Gubaidulina’s String Trio (as reduction). It is the only chord within a segment of roughly ten bars, and it is played homophonically by the three instruments without rests. This chord defies analytical categories linked to triadic music, such as the root of
a stack of thirds. The central note of the chord is D4, since the passage begins in m. 21 with a D4 unison in all instruments and builds gradually in a downward manner. After the introduction of our chord model, section 3 illustrates how such a challenging case can be represented.

## 2 A Unified Chord Model

Our proposed model represents chords as graphs, where the chord label and its position in a piece form a central node, and properties of the chord are given as labeled edges and attached nodes. A formal definition of our model, a comparison of several harmonic annotation standards, as well as more example graphical chord diagrams, can be found in our supplementary online material.¹ Figure 3a shows how the first chord from the Corelli example in Figure 1 is graphically represented. Here, categorical properties are represented by white rectangles, the key by a green parallelogram, and pitches and octaves are given in red and white circles, respectively.

![Chord Diagram](image)

**Figure 3:** a) Our model's full representation of the first chord from the Corelli example in Figure 1 (left). b) The different pitch types (GPC, SPC, EPC) in our model, together with the result of transposing them by some interval type (arrows), and the interval types (GIV, SIV, EIV) that measure the distance between two pitch types (labeled ovals).

Fundamentally, the model is based on viewing chord labels as selections of pitch classes (PCs, red circles in Figure 3a) plus octaves. To account for the different chord encodings described above, we model different types of pitch classes: Generic Pitch Classes (GPCs; A–G), Spelled Pitch Classes (SPCs; GPC plus accidentals), and Enharmonic Pitch Classes (EPCs; MIDI note number mod 12). An SPC can be converted into an EPC or a GPC, but not vice versa.

Each of a chord’s PCs can also be represented as an interval above some reference PC (e.g., the chord’s root or bass). Analogous to PCs, these intervals may be generic (GIV; the difference between two GPCs, e.g., any 3rd), specific (SIV; the difference between two SPCs, e.g., a major 3rd), or enharmonic (EIV; the difference between two EPCs, e.g., 5 semitones). Figure 3b illustrates the relationships between pitch and interval types in our model. Arrows indicate the resulting pitch type when transposing some PC by an interval (e.g., an SPC transposed by a GIV results in a GPC), and the large ovals specify the resulting interval type when measuring the distance between two PCs (e.g., the distance between a GPC and an SPC results in a GIV). These transformations are useful both for viewing a chord’s pitches as intervals, and for calculating PCs given a chord label.

that denotes only intervals. Similarly, the model can represent a chord's pitches as scale degrees (SDs) of a mode. If the key (essentially a set of SIVs with a reference tonic PC) is known, an SD is equivalent to an SIV over the reference tonic PC (see Relative Pitch Classes below for more details). In what follows, we describe different aspects of the model using the first chord from the Corelli example (Figure 1).

**Score Level.** This least abstract representation of a chord consists of the set of pitches that are taken from all the notes within the segment referred to by the chord symbol. On this level, pitches are typically represented as SPCs, although we can also model annotated MIDI files at this level with pitches viewable only as EPCs. Each PC is also associated with one or multiple octaves. This representation allows the model to group pitches into pitch classes.

**Pitch Equivalences.** The model allows for abstraction from the score level by applying different equivalence operations, e.g., octave or enharmonic equivalence. To apply octave equivalence, the octave(s) of a pitch are simply ignored. This allows one to interpret a chord as the set of unique PCs occurring in the chord: a pitch-class set. In cases where octave information is not available because a symbolic pitch representation is missing or does not include all pitches expressed by a chord label, octave equivalence is therefore necessarily assumed. Enharmonic equivalence is represented as a flag that may be associated either with individual PCs or the entire chord, converting the corresponding PCs to EPCs. Conversely, pitch classes and complete chords that are expressed as EPCs only (e.g., those from MIDI files), by default, come with an enharmonic equivalence flag.

**Pitch Functions.** Pitches and PCs can be assigned functions within the chord. Importantly, each can be classified as either a chord tone or a non-chord tone. The possibility of ignoring non-chord tones, such as suspensions or ornaments, is common to many annotation standards. Other common pitch functions are, for example, root, bass note, and leading tone, but this set of categorical pitch functions can easily be extended.

**Relative Pitch Classes.** All absolute PCs can be expressed in relation to a particular PC and/or to an ordered collection of intervals such as a mode or a Tonfeld. In the following, we refer to any ordered collection of SIVs as a “Mode”, and the combination of a mode and a tonic pitch as a “Key”, represented in our graphs as a green parallelogram. The tonic of a key, if present, may be represented as an absolute PC or again relative to another tonic or key, e.g., for secondary dominants and other chord borrowings, or for indicating a local key. Various levels of a tonal hierarchy may be disambiguated by a “Type” feature on the key. If viewed relative to a given tonic SPC, every PC type (GPC, SPC, EPC) can be represented through its corresponding interval type (GIV, SIV, EIV). In cases where the mode defines exactly one SIV for each GPC (e.g., major, natural/harmonic minor, phrygian, hypodorian, etc.), pitch classes are commonly represented as generic scale degrees (GSDs; 1–7, starting from the tonic, a one-to-one mapping from GPCs to SIVs). In these cases, each PC can also be expressed as a specific scale degree (SSD; GSD plus accidentals). Conversely, if a chord label expresses its PC content through GSDs (e.g., the generic Roman numerals in Figure 1) or SSDs (e.g., the specific Roman numerals and the bass degrees), key information is required to convert the scale degrees into PCs. In the case of other modes (e.g., octatonic, hexatonic, and pentatonic scales), the relative representation of chordal PCs as SDs is not common and, in our model, corresponding relative PCs remain defined in terms of intervals.

**Chord Functions and Properties.** Chord-level properties in addition to key information or enharmonic equivalence may also be added to each chord. These include chord type, inversion, and chord function (e.g., tonic, dominant, or predominant); one may also define and use custom functions and properties. To disambiguate functions that occur in multiple harmonic theories, a chord function can also have a “Theory” property indicating the particular theory through which the function has been assigned. This allows, for instance, dominant chords from Tonfeld and Riemannian theories to be grouped together (if the theory property is ignored) or expressed separately.

### 3 Applications

Figure 4 shows the application of our model to the post-tonal example in Figure 2. Here, we represent the SPCs of the chord together with their EPCs and attach the “Central Tone” function to D as provided by an analyst for the musical passage under consideration.
Since our model is represented as a graph, musicological research questions can be reformulated as graph queries, where all chords with a matching graph structure are valid answers to such a query. Figure 5 shows the corresponding graph for a possibly diachronic or cross-stylistic example query, where the queried element is represented as a question mark. Exploiting the model's flexibility and its various possible representations of PCs, formulating queries such as this on a wide range of annotations would become a straightforward task, regardless of their underlying representations.

Conclusion

We proposed a unified chord model and showed that it can generalize over a number of existing harmonic annotation standards and is capable of expressing a variety of challenging analytical cases in a wide range of styles including Western classical, late-Romantic,1 Jazz,1 and contemporary post-tonal music. While the model may not be exhaustive, its general and flexible nature ensures its extensibility: Its only requirement is that 'chord' in the sense of a collection of pitches is a meaningful concept in this style. The proposed model is therefore well-suited for music encoding standards that seek to consistently represent a broad variety of annotation systems, potentially allowing for conversion between them. Since the landscape of formal representations for harmony is so broad, we argue for a move towards a generalized standard for virtually all harmonic phenomena, which flexibly combines the utilities of each approach when possible. Our contribution should be understood as a first step towards this goal.

Acknowledgments

This research was supported by the Swiss National Science Foundation within the project “Distant Listening – The Development of Harmony over Three Centuries (1700–2000)” (Grant no. 182811). This project is being conducted at the Latour Chair in Digital and Cognitive Musicology, generously funded by Mr. Claude Latour.

Works Cited

Towards a Unified Model of Chords in Western Harmony

Johannes Hentschel, Fabian C. Moss, Andrew McLeod, Markus Neuwirth, Martin Rohrmeier

Abstract

Chord-based harmony is an important aspect of many types of Western music, across genres, regions, and historical periods. However, consistent representation and comparison of harmony across a wide range of styles (e.g., classical music, Jazz, Rock) is a challenging task. Even within a single musical style, multiple theories of harmony may exist, for example, the annotations in Fig. 1, each relying on its own (possibly implicit) assumptions and leading to different harmonic analyses, with a distinct focus (e.g., on the root of a chord, its bass note, or representation as an enharmonic pitch class). Cross-stylistic comparisons require a common underlying representation. The figures above exemplify this challenge through four stylistically distinct musical contexts with various harmonic annotations.

Fig. 1A. Symphony no. 9 (1893), I. Largo, mm. 1–5, score reduction, annotated with Riemannian and Tonfeld functions (C).

Fig. 1B. Tonfeld functions (C) and enharmonic (D) for the D major example above. From occurring notes derived.

Elements of the Unified Chord Model

Our proposed model represents chords as graphs, where the modeled chord is a position in a piece from a central node, and its properties are labeled with attached nodes representing values. Each node is represented as a black diamond in the figures above.

Pitch Classes (PCs, red circles). Fundamentally, the model is based on viewing a chord as a selection of pitch classes (PCs), with a chord's root or bass as the fixed reference point (e.g., in Fig. 2A, the G is the root of the chord). Each of a chord's PCs can also be represented as the interval above some reference PC (e.g., the root or bass; see for example Fig. 2B).

Chord-Based Harmony is an Important Component of Many Types of Western Music, Across Genres, Regions, and Historical Periods. However, Consistent Representation and Comparison of Harmony Across a Wide Range of Styles (e.g., Classical Music, Jazz, Rock) Is a Challenging Task. Even Within a Single Musical Style, Multiple Theories of Harmony May Exist, for Example, the Annotations in Fig. 1, Each Relying on Its Own (Possibly Implicit) Assumptions and Leading to Different Harmonic Analyses, with a Distinct Focus (e.g., on the Root of a Chord, Its Bass Note, or Representation as an Enharmonic Pitch Class). Cross-Stylistic Comparisons Require a Common Underlying Representation. The Figures Above Exemplify This Challenge Through Four Stylistically Distinct Musical Contexts with Various Harmonic Annotations.
Visualizing Harmony Using Chordal Glyphs and Color Mapping

Justin Lind
Stanford University, USA
justinmh@stanford.edu

Abstract
Musical scores are frequently annotated with harmonic information, but widely used text-based methods rely on a limited number of visual channels. Though glyph-based methods exploit more channels, existing systems often violate perceptual design principles when employing color and rarely capture the frequency of chordal changes or their harmonic function. In this work, we introduce a new design idiom for augmenting sheet music through chordal glyphs embedded directly within musical staves. Harmonic concepts, weighted by saliency and categorized by data type, are mapped to visual channels ranked by discriminability. Preattentive processing is leveraged to support various user tasks, alongside redundant encodings of foundational harmonic elements to improve overall perceptual effectiveness. Key names and chord roots are displayed using parallel hue-based 12-step categorical colormaps. We then distill several design implications inherent in assigning colors to musical pitches regarding perceptual and linguistic effectiveness. Following this discussion, we outline open research directions.

Introduction
While the most salient aspects of musical harmony are immediately evident when music is heard, harmonic effects typically remain obscured when a musical score is merely seen. We outline a new visual design idiom to bridge this gap using a visual vocabulary comprehensible to both expert and novice users. Using the framework of a four-level nested design scheme [29], we begin at the domain level for harmonic representation in Common Music Notation (CMN). Section 1 outlines current annotation practices and then shifts to the next design level to discuss harmonic abstraction and visualization practices. Going one design level lower, sections 2 and 3 explain the creation of the visual encoding idiom itself. The fourth nested vis-design level is the algorithm layer, where detailed procedures are created for a computer to automatically encode the harmonic information visually...this is discussed in the conclusion.

Traditional annotation methods do not sufficiently convey, visually, the breadth and richness of music that one perceives aurally. Given the information density of CMN, the insertion of any additional visual encodings faces several hurdles from a design perspective. Although numerous mark and channel types are not typically used in CMN [26], the way in which musical dimensions (particularly spatial and temporal) are encoded significantly constrain the available design space. We navigate this complexity by using a familiar concept, bar lines, as building blocks for encoding harmonic annotation via bar-line-like glyphs. While this restricts the available tools to a very particular range of marks and channels, we argue that the benefits of directly embedding information related to harmony and chordal movement within the musical staff are worth the effort.

1 Motivations and Related Work
This paper presents a new glyph-based method for visualizing harmony in CMN following basic tenets of visualization and color design. Three issues that are not fully addressed in the current literature motivate our design.

1. We would like a design that maximizes the effectiveness of how elements are encoded visually as glyphs within their musical context.
2. We would like to visualize the most salient harmonic concepts and ensure that the most prominent harmonic aspects are linked to the most impactful visual channels.

3. We would like to use color in a way that maximizes visual discernibility by using color values that are strongly associated with familiar color names.

Harmonic annotation is usually written in a space commonly used for lyrics and other instructional features, typically below the musical staves or sometimes above them. Similarly, glyphs are often placed above musical lines to avoid occluding critical material, a frequent requirement due to their size. Guidance around glyph usage consistently emphasizes the relevant task and data characteristics [2, 12, 17, 32], and related work on multiple views highlights the adverse effects of context switching. It underscores the importance of consistency and appropriate perceptual cues [33]. Discussion of glyph placement in complex situation-dependent situations, such as CMN, remains rare despite the overall volume of research [4, 10].

Although the use of glyphs specifically for augmenting CMN is less common, recent work has successfully demonstrated using glyphs for visualizing rhythmic patterns [11] and aggregated harmonic patterns [25]. However, these methods use glyphs that are too large for repetition at a high enough frequency to be repeated once per chord change. Star glyphs [15, 27] are similarly too large for insertion within or overlay upon musical lines. While placing glyphs outside of the musical staff utilizes the available space in a familiar way, externally placed glyphs are often too large to fit in a small enough space to keep up with the frequency of chord changes. Though adequate for information that will be aggregated at the grain of once per measure, glyphs as currently used are a poor match for representing swiftly-changing harmonic patterns. Although not glyph-based, basic visual representations have been shown to support simplified harmonic analysis [24]. Our first requirement, effective visual encoding, must therefore find a better format for augmenting CMN.

Secondly, successfully visualizing musical harmony’s complexity requires prioritizing the most important musical aspects to be encoded. Ideally, a project such as this would begin with a pre-existing hierarchy of tonal harmony that unambiguously informs a taxonomy-centered design process [23]. Though unable to find such guidance, we have used the work of Krumhansl regarding tonal hierarchies [19, 20, 21], functional harmony, and tonal harmony’s statistical characteristics more generally as a starting point for our prioritization [3, 8, 28].

Finally, for color to represent 12 categorically distinct keys successfully, numerous pitfalls must be avoided. While designers of qualitative color palettes should strive to assign colors that provide each color the same perceptual weight [5, 35], their use for encoding harmonic categories requires that memorable, robust, and easily discernible names be prioritized over consistency. This paper’s considerations of assigning color names are limited to English, but work on naming and perceptual differences across languages [1, 16, 30, 34] suggests that speakers of languages other than English might benefit from targeted adjustments to the selected colors (both color names and the hues themselves). Overall, the use of categorical color names that are distinct both linguistically and perceptually is recommended [6, 7, 13, 18, 22, 31].

2 Methodology

2.1 Glyph Design

To satisfy our goal of creating a design that maximizes the effectiveness of how elements are encoded visually as glyphs within their musical context, we begin at the level of the visual encoding idiom. Just as bar lines establish regularity and discretize the element of time into measures, we will use bar-line-like glyphs to discretize harmonic shifts at the grain of one glyph per chordal change. As the harmonic function and effect of any given chord depend on its surrounding context, we will seek a means of visualizing supportive elements through background effects rather than within the glyphs themselves. Table 1 outlines this general process: Each visual channel is assigned a rank of how effectively it functions with both categorical and magnitudinal variables, the Channel Bins Available column provides a rough number of how many variables each channel can reasonably encode, and the Summary Visual Salience column provides an overall rank across both categories.
2.2 Harmonic Encoding Targets

Having created a suitable carrier of additional information, we return to the data abstraction level to identify and prioritize the harmonic targets to encode. In the absence of a definitive hierarchy (the creation of a comprehensive taxonomy being beyond the scope of this project), we created a proof-of-concept catalog for illustrative purposes. Table 2 shows several of the selected aspects alongside some of their relevant attributes: Each harmonic concept is assigned a rank based on its importance as either categorical or magnitudinal variables, the Harmonic Bins Required column provides a rough number of how many values the concept contains, and the Total Harmonic Salience column provides an overall rank of each concept across both categories.

3 Visual Encoding

3.1 Mapping Harmonic Aspects to Visual Channels

In order to visualize the most salient harmonic concepts and ensure that the most prominent harmonic aspects are linked to the most impactful visual channels, our next task is to map the available visual channels to the set of harmonic targets. Assignments balance harmonic importance, channel appropriateness, channel salience, and metaphorical fit. Several background effects provide the required environmental context: Key modal brightness is mapped to page lightness, key pitch to background hue, and key mode and modal brightness values to background pattern. Glyphs themselves convey the remaining information, beginning with harmonic motion. Chord function is mapped to plumbness (tonic chords as vertical and at rest, dominant chords...
as decelerating and tilting backward, weak predominants tilting slightly forward, and strong predominants accelerating further forward still). Cadences receive three encodings: additional weighting via bolding, depth via a drop shadow, and increased width proportional to their cadential strength. The four chord triad types are mapped to orientation. Modulations are mapped to a second glyph, according to the new key, overlaid directly upon the glyph from the previous key. Harmonic distance is mapped to increasing chromatic distance between glyph hues and the key itself. Seventh chords (and their type) are mapped to glyph orientation and curvature, and chord inversions to vertical displacement from a shared baseline. Finally, chord names as chord root pitch are mapped to glyph hue and saturation/lightness.

3.2 Color as Pitch

Our third goal was to use color in a way that maximizes visual discernibility by using color values that are strongly associated with familiar color names. Color values were selected based on overall discernibility (a property of the geometric distance between any two points in color space) in combination with the saliency and familiarity of the color names themselves. Included are 8 out of the 11 most common, or basic, color names in English – pink, red, brown, orange, yellow, green, blue, and purple (white and black were skipped for obvious reasons, as was grey based on expected confusion given the complexity of CMN). Olive, teal, pale blue, and lilac are added to these eight to subdivide the hue space further.

We select one shade that is relatively bright alongside another that is darker or more muted within each color. This serves to metaphorically encode modal brightness for keys (such as major or minor) and as a redundant encoding of a chord's triad type (augmented or major as brighter, minor or diminished as darker). An additional advantage of this technique is that it provides a more robust namespace to differences in color naming – users can always fall back to a meaningful color term that encompasses two different color values if needed. Selection of the specific shades was an iterative process that balanced several goals: ensuring the fidelity of the chosen color value to the color set's name, maintaining the spirit of darker or lighter shades for two parallel palettes, and maximizing the geometrical distance between neighboring values (see Figure 1). Colors also cannot be too dark or too light, as either would undermine accessibility based on the required contrast against a very light page background. Given the relatively small size of glyphs, contrast ratios of 3.0 were targeted (an accepted benchmark in web accessibility for large text, graphical objects, and user interface components).

Conclusion & Future Work

In this manuscript, we presented a new method for visualizing harmony using bar-line like glyphs. Encoding decisions were guided by channel attribute types, both harmonic and visual, while color guidelines emphasized categorical color geometry. Although we have not yet fully automated this process (the ‘algorithm’ layer of Munzner's design scheme), Figure 2 provides a proof-of-concept illustration of this new design idiom in use. We plan to further refine these initial steps alongside creating integrations with Verovio and the broader ecosystem of MEI encoding tools.

Acknowledgments

The author thanks Craig Sapp, whose early work inspired this project, Tamara Munzner for creating a design framework suitable to this project's breadth, and Jeffrey Heer and Maureen Stone for the bread crumbs related to categorical colors.
Figure 1: Name-based comparison of qualitative color palettes. The color names shown in this figure are generated automatically by algorithm [14] hence the slightly darker-than-lilac color top-left tagged as purple. Color-name distances shown via matrices; salience scores with bar charts. Salience scores below 0.2 suggest likely naming confusion. The proposed palettes provide better saliency with less name overlap than “ColorBrewer-Q12” and are comparable to the 10-color “Tableau Classic 10” while encoding more values (these are two popular categorical palettes commonly used in visual design).

Figure 2: Beethoven’s ‘32 Variations on an Original Theme in C minor’, WoO 80, mm. 1–16. Harmonic and functional analysis sourced from the TAVERN data set [9]. Score rendered using Verovio Humdrum Viewer. Proof-of-Concept chordal glyphs were added manually alongside key pitch and key modal brightness (the red background indicating C minor) using Inkscape vector graphics software.
Works Cited


Visualizing Harmony using Chordal Glyphs and Color Mapping

1. Harmonic Concepts and Visual Channels
   ranked by salience

2. Effectiveness-weighted

3. Combined for use in with Chordal Glyphs

4. Circle of Fifths-arranged
   Familiar Color Names

Abstract

Musical scores are frequently annotated with harmonic information, but widely used text-based methods rely on a limited number of visual channels. Though glyph-based methods exploit more channels, existing systems often violate perceptual design principles when employing color and rarely capture the frequency of chordal changes or their harmonic function. In this work, we introduce a new design ideology for augmenting sheet music through chordal glyphs embedded directly within musical scores. Harmonic concepts, weighted by saliency and categorized by data type, are mapped to visual channels ranked by discriminability. Parameters processing is leveraged to support various user needs, alongside redundant renditions of foundational harmonic elements to improve overall perceptual effectiveness. Key names and chord roots are displayed using palette-based 12-tone categorical color maps. We then distill several design implications inherent in assigning colors to musical pitches regarding perceptual and linguistic effectiveness. Following this discussion, we outline open-source directions.

Proof-of-Concept

An example of an alternative chordal glyph design. Here, the scale degree of the root of each chord is mapped to its binary letter using parametrized alphabetical characters. Chord function continues to be shown by glyph color: major triads forward, minor forward, leading inversion, and straight up and down when serving as a Tonic function. Deep shadows again indicate a FMN.

Future Work

There is still a great deal of work to be done refining this design, from the mapping of harmonic concepts to the breadth of visual channels used. This design ideology may also be suited to encoding semantic musical information, something that we would like to explore. The color space, in terms of 12 distinct families of color for the 12 western pitch classes, seems surprisingly robust. This suggests numerous feasible routes for employing consistent color mapping to keys/chords across musical domains, from pedagogy, to guitar tablature, to more abstract real-time musical visualization.

The dominant chords of C Major are shown here, this time with their corresponding Roman numerals sized roughly based on their overall likelihood of appearance in Western Art Music between 1750-1900.
Diagrammatic Analysis of J.S. Bach's *The Well-Tempered Clavier* Fugues, BWV 846–851

Anna Matuszewska ¹,²  
¹ The Institute of Literary Research  
of the Polish Academy of Sciences, Poland  
an.ma.matuszewska@gmail.com  
² Institute for Music Informatics and Musicology  
University of Music, Karlsruhe, Germany  
christoph.seibert@hfm-karlsruhe.de

Abstract

The field of musicology is constantly being enriched with digital, searchable music data. This trend opens new research possibilities; conversely, it requires new abilities to work with numerous data sets efficiently. Digital tools facilitate searching large music corpora and serve music analysis well. Nevertheless, there is still a potential to better harmonize research perspectives from musicology and computer science to make computational analysis outcomes more explicit, comprehensible, and flexible.

The aim of this paper is to present new ways of handling, displaying, and considering musicological data. Music information from fugues BWV 846–851 composed by J.S. Bach, retrieved with Humdrum Tools and the Music Processing Suite (MPS) software, was processed and translated into a relational database. The visual display of the retrieved information was accomplished with dashboards using the data visualization software Tableau Public. The possibility of comparing each fugue’s voices makes it easier to comprehend the knowledge hidden behind music data. Additional options enable further visual exploration of the analyses and ensure conditions for abduction under assumptions of diagrammatic reasoning as proposed by Charles Sanders Peirce.

Introduction

Over the past decade, digital scholarly editions of music collections have flourished around the globe, and this trend continues. Digital tools for music analysis can visualize music features such as interval distribution, rhythmic patterns, and other statistics in ready-to-use charts. As a result, the analysis of large corpora has been simplified, but establishing relations between both subsets of a corpus and particular pieces of music can still be perplexing. Therefore, the visual exploration of analytical data should be regarded as a critical part of the research process.

Symbolic music data can be presented on different kinds of diagrams, thus becoming part of the epistemic process. Sybille Krämer argues that hybrid forms of word and image trigger our creativity, and it is one of the reasons why thinking with diagrams as hybrid representations “form the fundamental semiotic basis of human cognition” [7, p. 15]. According to Valeria Giardino, diagrams are computationally efficient because they externalize information in space, and therefore the user may act on them. They are very good inferential shortcuts in problem-solving. By spatially displaying their content, they affect memory and reasoning. Thanks to their nature of images, they can target issues concerning the relationship between perception, cognition, truth, and knowledge. Diagrams are not only spatial tools; they are ‘scaffoldings’ enhancing the ability to reason and infer. The space of the diagrams is a space interpreted along with textual and background knowledge [4].

The outcomes of the music analysis of Bach’s fugues, BWV 846–851, were visualized on dashboards with interactive charts published on the platform Tableau Public. Chart combinations are filtered jointly. The output may be narrowed down to any number of voices, a range of measures, or a specified musical form element such as subject, countersubject, and episode. Explanatory descriptions support visual information presented on dashboards, and it contains supplementary textual information encoded in tooltips that may be displayed by hovering over an element.

¹  Link to the project: https://tabsoft.co/3qfeYnt (accessed January 12, 2022).
© Published in 2022 with Humanities Commons under a CC BY-NC-ND 4.0 (https://creativecommons.org/licenses/by-nc-nd/4.0/) license.
The concept of diagrammatics inspires this project. Its main goal is to afford inference by the spatial display of symbolic music data. The encoded information is presented in diverse representational formats on a group of combined interactive charts. This approach tends to exploit their illustrative capabilities amplifying dynamic analysis. The current implementation of the project is adapted for two stages: a detailed analysis of a single piece of music and a comparative analysis of music collections on the macro level.

1 Diagrammatics as an Approach to Music Analysis

The concept of diagrammatics refers to the heritage of the American pragmatist and semiotician Charles S. Peirce. It addresses the design and processing of knowledge in diagrams and can be understood in two ways. First, the diagrammatic project may be developed from the manifestations gained in concrete diagrams (bottom-up analysis). Second, however, the consideration of concrete diagrams may also be derived from the theoretical premises of a general diagram (top-down analysis). The first approach is a theory of properties linked to diagrammatic structures, while the latter puts in the spotlight an epistemological process primarily associated with mentally realized diagrammatic conclusions [1, p. 16–17]. The diagrammatic reasoning proposed by Peirce is in harmony with the latter approach [1, p. 21].

Peirce stated that conclusive thinking is performed diagrammatically and interlocked diagrammatic reasoning with abduction [1, p. 64]. The abduction theory evolved from the early syllogistic theory where deduction, induction and abduction were treated as three different classes of reasoning [8] to the inferential theory where they were seen as three stadiums of one research method, in which abduction triggered the whole process [13, p. 15–25].

Abduction is the process of forming an explanatory hypothesis. [...] Deduction proves that something must be [emphasis changed to italics]; Induction shows that something actually is operative [emphasis changed to italics]; Abduction merely suggests that something may be [emphasis changed to italics]. Its only justification is that from its suggestion deduction can draw a prediction which can be tested by induction, and that, if we are ever to learn anything or to understand phenomena at all, it must be by abduction that this is to be brought about [9].

The concept of diagrammatics is based on the belief that the iconic character class of the diagram is a metaphorical pattern (and in so far, a diagram itself) of inferential thought [1, p. 64]. Diagrams convey abstract information by depicting a physical situation from which the abstractions can be inferred [2, p. 5–22].

To sum up, the overall picture of the diagrammatic reasoning process is that it forms a formal deductive reasoning core, embedded, on each side, in the trial-and-error of abductive trials and inductive tests [12, p. 105].

Abduction is inextricably linked with diagrammatic thinking, and this project aims to construct a combination of charts enabling the user to interact with them in accordance with this concept. Basic music information anchored on proposed interactive charts can be viewed broadly or within a narrowly defined scope. It can be reshaped and enriched with available additional information displayed both visually and numerically on linked charts. As a collection of charts, each dashboard comprises different representational formats presenting chosen elements of musical analysis. The process of abduction can start with any chart that draws the attention or curiosity of the viewer. The interaction with data, the possibility to access it differently, freely combine, and filter it, empowers creative reasoning. The information from each chart complements each other, providing the core of deductive reasoning. Accessing the data from different angles and multiple changes in its configuration make it possible to test an explanatory hypothesis inductively.
2 Data Preparation and Data Analysis

A computational analysis of six fugues from *The Well-Tempered Clavier* by J.S. Bach (BWV 846–851) was conducted using Humdrum Tools\(^3\) and the Music Processing Suite software [5, 6]. The choice of tools was dictated by the fact that none of them requires any knowledge of programming languages; mastering them is possible for scholars not advanced in technology and both are available as free software. In addition, the flexibility of Humdrum **kern data and the ability to attribute information to each note is valuable.** Generating data in Linux command line is not troublesome, and the data representation format is intuitive. Music Processing Suite does not require any significant technical preparation; after the installation, an analysis report may be generated with a set of built-in charts describing various musical features. The data generated in .csv format can be accessed and transformed easily.

All the statistical calculations are based on musical scores from the KernScores library [10] encoded in the Humdrum **kern data format.** Humdrum Tools were used to calculate a number of notes and rests, the highest and the lowest note, the shortest and the longest note duration (command: *census* -k), melodic and rhythmic progressions (command: *extractx*), statistics about melodic intervals in each piece (command: *mint* -s) and dissonant types (command: *dissonant* -u). In addition, Music Processing Suite software was applied for harmonic analysis. Because coordination between Tableau Public, music score, and recording is not possible, there is an integrated link to Verovio Humdrum Viewer [11] in the navigation bar where the selected sheet of music can be simultaneously listened to.

3 Visual Representation and Interactive Elements for Visual Exploration

The main assumption of our project was to examine possibilities of displaying analysis results, to enable interaction with data, and ensure conditions for diagrammatic reasoning, not a thorough analysis itself. The presented project is in the development phase, and the adequacy of chart types for each of the music element representations are to be evaluated by the target audience. This version should indicate what impact on research practice is possible with the proposed data representation. The intent is to test the influence of diagrammatic reasoning on the research practice, not the choice of single diagrammatic elements.

The project is designed on three dashboards: two dashboards with the focus on the melodic aspect (Figure 1) or the rhythmic aspect (Figure 2), in which one fugue from a collection is displayed, and a complementary dashboard (Figure 3) on which multiple chosen pieces can be compared on a macro level. Tableau operates on data from Excel worksheets. Aspects of a musical piece are stored in separate Excel files (e.g., melodic sequences, harmony, dissonance, intervals). Each file comprises worksheets for each fugue, which are merged in Tableau using the union function. The Excel files are joined using mutual columns of data (number of fugues, voice, onset).

All the charts on each dashboard are interlinked, and the display changes dynamically according to the filters applied. Furthermore, each chart is supplied with an information button icon with a detailed description of the axes, the aim of the chart and the tool for data extraction is named.

3.1 Main Dashboard

Navigation through the main dashboard provides the possibility to narrow down the scope of the analysis to voices, measures, particular segments such as subjects or countersubjects, their unrestricted combination or to a single note. In addition to filtering, music information may be highlighted according to its attribute. It also displays cadences and dissonances.

*Subject and Countersubject Analysis.** A Gantt chart placed on the left top side of the main dashboard shows the occurrences of subjects and countersubjects over time in different voices in parallel. It contains all the appearances of the subject, countersubject, their variations, and cadences spread over time (x-axis). These may be a basis for data filtration. Voices (y-axis) are defined as in Humdrum music representation, where they are represented by spines and counted from the lower one upwards. The depicted analysis of fugues BWV 846–851 is

---

based on algorithmic calculations. The output was compared with a traditional analysis conducted by Siglind Bruhn [3] to eliminate false-positive results.

**Interval Analysis.** On the top right side of the dashboard, a column chart calculates the number of melodic intervals in a chosen voice, voice combination, or fragment. On the y-axis, the direction of the interval is marked with a color. Additional information available by hovering over a column contains the percentage of the respective interval in a chosen voice and a complete composition. By clicking on a specified interval, its occurrences are shown on corresponding charts.

**Piano Roll.** Another Gantt chart on the bottom of the main dashboard with x-axis representing onset and y-axis representing voices. It allows for tracking the melodic progression along with its pitch, notes duration, context harmony. Additionally, chords or dissonance types can be highlighted using the lists displayed in the Dashboard Navigation section. A description of dissonances is provided in the legend.

The following passage describes a possible application of this dashboard for music analysis using the example of fugue BWV 847 (Figure 1):

*After looking at the Subject and Countersubject Analysis chart we learn that there are six episodes. By observing the number of melodic intervals separately for subjects and episodes, we see that the proportion of intervals is different; we hypothesize that episodes’ material is not derived from the subjects’ material. From the theory, we know that there is a need to test rhythmic similarity as well. The Rhythmic Activity dashboard (see 3.2) suggests that rhythmic patterns in subject and episodes are similar. We go back to Piano Roll, zoom into the episode bars separately, and learn that although the jump in the head motif of the subject was changed, episodes 1, 2, 4 and 5 are bonded with subjects’ head motif, episodes 3 and 6 are built on another material.*

---

4 Algorithmic detection of subjects and countersubjects is part of a broader-scope project that is not finished yet and will not be discussed in this paper.
3.2 Rhythmic Activity Dashboard

This dashboard contains three interactive charts devoted to rhythm analysis. The Rhythmic Activity chart constitutes a central place, and it may be a starting point for abductive reasoning. Its explicit visual outlay suggests possible fragments of the piece of music that might be worth leaning over. Note Duration Distribution and Note Duration Distribution per Beat charts may trigger the generation of abductive hypotheses. Deductive steps should be based on the analysis of all proposed charts.

Figure 2: A view of the Rhythmic Activity dashboard in the Fugue Analysis project with data visualization of J.S. Bach's fugue No. 1, C major, BWV 846.

Rhythmic Activity is designed as a heat map. The x-axis depicts measures, y-axis voices. By hovering over a field on the chart, the number of attacks in a measure is displayed. The darker a colour of a field, the more attacks are in that measure.

Note Duration Distribution is a tree chart that gives information about the number of occurrences of each rhythmic value in a chosen piece of music or one of its parts (when filtered). The information is displayed as a number but also as the size of dedicated fields.

Note Duration Distribution per Beat is a bubble plot in which the x-axis is a mathematical division of one measure, and the y-axis presents the number of occurrences of each note duration. Thus, the chart indicates what rhythmical structures prevail on each beat. An additional tooltip shows specified information about the exact number of occurrences of a rhythmic class and its percentage in a fugue.

The following passage describes a possible application of this dashboard for music analysis using the example of fugue BWV 846 (Figure 2):

We begin the exploration of this dashboard by looking at the Note Duration Distribution chart. It informs us about the proportion of each note duration in the fugue. Surprisingly, the number of occurrences of dotted quavers, demisemiquavers and crotchets is quite similar. The hypothesis is that there is a rhythmic pattern built on these three note durations. From the Note Duration Distribution per Beat, we infer a bond between the two of them, and the density of the notes’ appearances on beat shows that dotted quavers always precede two
demisemiquavers. The heat map shows that the pattern repeats itself in all the voices, suggesting that this is a fixed motif. We may presume that it is a part of the subject. We test the hypothesis by filtering by subject in the main dashboard’s Piano Roll chart (see 3.1).

3.3 Comparative Analysis Dashboard

This dashboard is designed for the overall comparison of the musical form of different pieces in a corpus. It contains statistical information regarding ambitus, rhythm, number of notes and pauses, and rhythmic distribution. In the bottom half, there is a possibility to choose between three interactive charts: The first representation shows the number of intervals as a column chart where the mode of intervals is differentiated, the second contains summarized information about the relative frequency of intervals, and the last one is a rhythmic tree map.

![Figure 3: A view of the Comparative Analysis dashboard in the Fugue Analysis project showing the data visualization of main characteristics of J.S. Bach's fugues No. 1–6, BWV 846–851.](image)

Conclusion

The application of diagrams in research creates cognitive advantages and enhances inferring. The main goal of this project was to adapt the diagrammatic thinking concept to music analysis and explore its potential to enrich research practice. Different representational formats of data presentation incorporated into each dashboard and their dynamic transformations creatively steer our minds to find a path to possible conclusion. The combination of visual and semantic data display makes diagrammatic representation efficient. Furthermore, the explanatory hypotheses generated for the chosen scope may be deductively proved, since exploring all proposed charts gives a detailed overview of chosen aspects; the change of the scope of analysis asserts conditions for inductive testing of the hypotheses.
The proposed approach of symbolic music data processing and visualization facilitates the comprehension of a single piece of music and music collections. Additional options give insight into the musical form from different angles and provide a space for exploration in accordance with abductive reasoning. Explanatory hypothesis inferred for one piece of music may be applied to other pieces from the corpora, and outcomes may be compared with the integrated music score.

In the light of the first experiences within this project, further developments towards tools for an automated process of data preparation and visualization are desirable to enable a fluid integration of diagrammatic reasoning and visually driven explorative processes into musicological research. The further development of visual analysis will be based on empirical observations of the users interacting with dashboards. Owing to this approach, our analysis method should be pragmatic and easy to implement in everyday practice.

Acknowledgments

This project is part of a Bi-nationally Supervised Doctoral project funded by the DAAD (German Academic Exchange Service).

Works Cited

Diagrammatic analysis of J.S. Bach’s
The Well-Tempered Clavier fugues BWV 846-851

Anna Matuszewska | The Institute of Literary Research of the Polish Academy of Sciences | an.ma.matuszewska@gmail.com
Christoph Seibert | Institute for Music Informatics and Musicology University of Music Karlsruhe | christoph.seibert@hfm-karlsruhe.de

INTRODUCTION

The textural analysis of fugues through large music corpora and music analysis software is a tool for generating ideas. As a result, large corpora have been used to develop a method for visualizing music scores. In order to make computational analysis accessible, scalable, and flexible, the visualization of an example score should be regarded as an initial part of the research process.

SYMBOLIC MUSIC DATA

Symbolic music data can be presented in different kinds of diagrams, thus becoming part of the analytics process. Thinning with diagrams as hybrid forms of text and image lends the fundamental notion of human cognition [2]. Diagrams are used to visualize and enhance the ability to reason and infer [2].

DIAGRAMMATICS

The concept of diagrammatics, which is the heritage of the American pragmatist and semiotician Charles Sanders Peirce, addresses the design and processing of knowledge in diagrams. In essence, understanding is put in the spotlight on epistemological processes primarily associated with visually mediated epistemological conclusions. The concept of diagrammatology is written diagrammatically and interested in diagrammatic reasoning with abduction [7].

Abduction is the process of building an explanatory hypothesis. Abduction is the process of building an explanatory hypothesis. Deduction proves that something must be true; induction shows that something actually works; abduction only suggests that something can be true. The process of abduction can turn its assumptions into a prediction which can then be tested, and thus can even learn anything or understand phenomena at all. We can only achieve through abduction, for example: Fugue BWV 846 in C Major.

THE AIM OF THE PROJECT

The main aim of our project is to examine possibilities of displaying analysis results in an interactive way with data and ensure conditions for abduction under assumptions of diagrammatic reasoning as proposed by Charles Sanders Peirce, not a thorough analysis itself.

METHODOLOGY

- Music information from fugues BWV 846-851 composed by J.S. Bach was retrieved with Humdrum Tools and Music Processing Suite. MPS is a framework that supports music analysis and generation.
- Analytical information was processed and translated into a relational database.
- The visual display of the analytical information was accomplished with dashboards using the data visualization software Tableau. Tableau provides a platform for the visualization of the results.
- The outcomes of the research analysis were presented in various representational formats on a group of combined interaction sheets that are linked jointly.

Interactive Elements for Visual Exploration

Detailed analysis of a fugue: Example: Fugue BWV 846 in C Major

Comparative analysis: Example: fugues BWV 846, 848, 851

CONCLUSION

The application of diagrams in research creates cognitive advantages and enhances the ability to reason and infer [2]. The visual display of the retrieved information was accomplished with dashboards using the data visualization software Tableau. Tableau provides a platform for the visualization of the results.

FURTHER INFO

2. https://github.com/AnnaMariaMat/MEC-2021-Bach
1. https://public.tableau.com/app/profile/anna.m8322/viz/poster-v3/1-MAIN

BIBLIOGRAPHY

ODD Structures and Where to Find Them

Benjamin W. Bohl
Johann Wolfgang Goethe-Universität Frankfurt, Germany
bohl@em.uni-frankfurt.de

Johannes Kepper
Paderborn University, Germany
kepper@edirom.de

Abstract

In the past twenty years, the technical setup of the Music Encoding Initiative (MEI) data framework has been adjusted several times. Each of those transitions was motivated by the wish to improve the ways in which MEI could be integrated with other formats, to simplify the maintenance of MEI, and to encourage more people to actively contribute to the development of MEI. Some of those objectives are contradictory, and accordingly, there is no single right answer for all times about the best possible technical setup for MEI.

The main purpose of this poster is to give a historical overview of the technical setups that MEI has gone through in the 20 or so years of its existence, and to illustrate the current workflows. Ideally, this empowers wider parts of the community to contribute to the continued development of both the MEI specification and documentation. Eventually, it will explain the steps necessary to set up a local working environment to participate in these developments.1

Historical Background

The first versions of MEI were specified using Document Type Definition (.dtd) files, starting around 2001. While not being officially published back then, the first version of MEI to be made available as a set of XSD (XML Schema Definition) files was version 1.7b, released in September 2006.2 This version introduced the possibility to use an XML namespace for MEI, turning it into a much more serious option for storing scholarly research data. While many structures available in current MEI sources were already introduced back then,3 the lack of easily accessible documentation was surely a major obstacle for the wider dissemination of the format at that time. Actually, there was some very good documentation, but it was embedded into the MEI sources as XML comments, and it mixed user documentation with explanations of the underlying technical structures.

As a wider community started to adopt MEI, that community published a first official release of the format in May 2010, which is still available on GitHub.4 This was the first version of MEI expressed as a set of RelaxNG (RNG) files. More importantly, the release included a PDF file called the “MEI Tag Library”, which provided proper end-user documentation about the elements available.

The next important change to the technical organization of MEI was introduced with MEI v2.0.0, released in August 2012. This was the first version to adopt TEI’s ODD language as the underlying technical framework.5 The eponymous one document does it all (ODD) system is based on the idea of writing the technical format specifications and the corresponding documentation in the same place, and generating both schema and docs from there. The intention is to help avoid discrepancies between the two, and to ensure full coverage of the documentation. For MEI, this transition allowed the use of tools and workflows developed by the TEI community and simplified the combination of both TEI and MEI data in a mixed environment. Another benefit is that ODD has a built-in mechanism for customizations and extensions of the standard, which are again

1 This poster will assume a certain level of familiarity with technical concepts both from the XML world and around GitHub. We decided to not bloat this poster with (references to) explanations of these terms, and not give links to authoritative introductions. In our experience, there is no single ‘best’ explanation, but depending on the individual learning curve, different materials will be considered as appropriate starting points. However, all these terms and technologies are very common, so that Wikipedia or a simple search on Google will bring up tutorials and introductions easily. In order to follow this poster, it is not required to have expert knowledge in those technologies.


3 For instance, the organization of attributes in classes organized by musical domains still in use today was already implemented in those versions of MEI, utilizing quite different technologies than today.


5 ODD was introduced as a meta-schema-language by the Text Encoding Initiative during the 1990’s. See [1].

© Published in 2022 with Humanities Commons under a CC BY-NC-ND 4.0 (https://creativecommons.org/licenses/by-nc-nd/4.0/) license.
self-documenting. The MEI documentation, now for the first time called the “MEI Guidelines”, was significantly extended by more than 250 pages of prose description of how to use MEI, and also for the first time included documentation for attribute classes, model classes, and other technical structures used to specify MEI.

However, the perceived complexity of ODD led to very few contributions to the MEI sources from other than a small group of community members that were actively involved in the transition to ODD in the first place. Hoping to lower technical barriers and thus increasing community involvement in the maintenance of the MEI Guidelines, it was decided in 2017 to once again separate the documentation from the MEI specification. While the specs continued to be expressed in MEI, the Guidelines were converted to Markdown and published through GitHub Pages, making it relatively easy to contribute directly online with automatic pull requests.

In the years since that last transition, however, it became clear that this Markdown approach did not increase participation to the extent expected. At the same time, it became evident that the separation of specs and docs also led to significant friction loss when trying to keep both aligned. As both parts were stored in different repositories, pull requests had to be made independently against two separate repositories, which led to inconsistencies more often than necessary and also facilitated the development of the specs without keeping the docs in line. Accordingly, in a great community effort over the past year, the specs and docs of MEI have been reintegrated into a single codebase, which is now again compliant with the concepts of ODD. However, this return to ODD will also inevitably bring back the former challenges. Accordingly, the community is currently seeking to address this situation in multiple ways. First, better documentation for the ODD setup itself is provided. This includes online tutorials on how to use ODD in the context of MEI, but also workshop events that offer in-person training on ODD. There are also monthly so-called ODD meetings. Parallel to this, the community offers tools like the MEI Profile Drafter, which facilitates the generation of project-specific ODD customizations, or the ODD-API Web Service, which allows software developers to query ODD structures programmatically. We anticipate that over time these efforts will increase ODD literacy within the MEI community, lead to a better use of the available concepts, and ultimately help to improve project-specific documentation.

Repository Structure

Currently, the MEI source code is maintained in a single repository on GitHub. The main assets are located in the two folders /customizations and /source within that repository. The first folder holds ODD customizations for the official MEI profiles mei-all, mei-CMN, mei-Mensural, mei-Neumes, mei-all_anyStart, and mei-basic. Those profiles are pre-built versions of MEI, tailored to specific common use cases. The second folder holds the MEI source code. The main entry point there is the mei-source.xml file, which serves as an overarching container for all other files containing the MEI docs and specs. It uses XInclude pointers to pull in the contents of these additional files. For an XML parser, this linking is fully transparent – it looks as if the content of those other files would occupy the position of the Xinclude statements. This allows the specs to be split up into separate files according to MEI modules, while the docs are split up to match the first-level chapters. The main motivation for this separation is an improved maintainability of MEI. The resulting files for each component seem to offer a good compromise between having sufficient context and a manageable size.

Formerly, the RelaxNG schemata compiled from the MEI sources were also stored in the /music-encoding/music-encoding source repository. Today, this is organized in a different way. On a commit to the source repository, both docs and specs are compiled automatically through a so-called continuous integration (CI) workflow running on GitHub Actions. The resulting Guidelines files are then automatically pushed to the separate /music-encoding/guidelines repository and served to the MEI website from there. The gen-

---

7 There was a workshop on ODD during Music Encoding Conference 2021. Other opportunities include the annual MEI developer workshop in fall or the annual Edirom Summer School in Paderborn (https://ess.upb.de; accessed January 12, 2022).
8 There are also monthly so-called ODD meetings. Parallel to this, the community offers tools like the MEI Profile Drafter, which facilitates the generation of project-specific ODD customizations, or the ODD-API Web Service, which allows software developers to query ODD structures programmatically. We anticipate that over time these efforts will increase ODD literacy within the MEI community, lead to a better use of the available concepts, and ultimately help to improve project-specific documentation.
12 See https://www.w3.org/TR/xinclude/ (accessed February 14, 2022).
erated schemata are automatically transferred to the /music-encoding/schema repository. With this new workflow, both repositories are primarily considered read-only access relays of the compiled build artifacts. Although technically edits are, of course, possible, they are highly discouraged and reserved for bug fixing purposes, e.g., of older MEI versions.

**Contributing**

With this new source code structure, the ‘official’ procedure to contribute to MEI is roughly as follows:

- An MEI Interest Group, an individual member of the MEI community, or just anyone willing to contribute to MEI prepares an idea on how to improve MEI.
- After discussing it, this idea is fleshed out as proper ODD customization. If necessary, members of the Technical Team of MEI will assist with this task.
- The customization is expected to not only change the MEI specifications, but also to cover the changes in the Guidelines.
- A Pull Request (PR) against the /music-encoding/music-encoding repository with the proposed changes is made.
- Technical aspects and interdependencies with other parts of MEI are discussed openly on GitHub.
- As soon as the discussion seems to be resolved, the PR will be added to the agenda of the next upcoming ODD meeting. Ideally, someone involved in the preparation of the PR should join that meeting in order to accommodate potential questions.
- During the ODD meeting, the community members present will have a final check on the PR and see if there are no major objections concerning the proposed changes. If approved and all questions could be resolved, the PR will be accepted and merged into the development version of MEI. If there are still questions, the topic is tabled to the next ODD meeting.

For all of these steps, support is always available, and feedback and questions are strongly encouraged at any time.

Besides this official workflow, it is of course possible to compile customizations locally for testing purposes, etc. This can be done using the MEI Customization Service, but also in a local working copy of the /music-encoding/music-encoding repository. The latter requires the following steps on the command line:

1. If you do not have a local copy (clone) on your local machine yet, run the following from your command line:
   
   ```
git clone https://github.com/music-encoding/music-encoding.git --recursive
   ```

2. If you already have a clone on your system, you still might have to initialize the submodules by running the following commands from the command line:
   a. Switch to your clone’s directory:
      ```
cd [YOUR-CLONE-LOCATION]
   ```
   b. Initialize the submodules:
      ```
git submodules init
   ```
   c. Update the submodules:
      ```
git submodules update
   ```

---

15 According to the MEI By-laws, membership in the community is bound to the subscription of the MEI Mailing list, which is open to anyone interested in MEI (see [https://music-encoding.org/community/mei-by-laws.html#3-community-membership](https://music-encoding.org/community/mei-by-laws.html#3-community-membership); accessed January 12, 2022).
16 See [https://meigarage.edirom.de/customization](https://meigarage.edirom.de/customization) (accessed January 12, 2022).
3. Check if your system meets the build requisites:\footnote{17}
   a. Is Java 8 or above available on your machine?
      
      \texttt{java -version}
      
      This should return something similar to:
      
      \texttt{openjdk version "11.0.9" 2020-10-20}
      \texttt{OpenJDK Runtime Environment (build 11.0.9+11)}
      \texttt{OpenJDK 64-Bit Server VM (build 11.0.9+11, mixed mode)}

   b. Is Apache Ant installed?
      
      Apache Ant\footnote{18} is a library for building software projects and drives the creation of MEI schemata and guidelines from the ODD sources. Run the following command to see if it is available on your system:
      
      \texttt{ant -version}
      
      This should return something similar to:
      
      Apache Ant(TM) version 1.10.9 compiled on September 27 2020

4. Initialize the build process:
   a. Switch to your clone’s directory:
      
      \texttt{cd [YOUR-CLONE-LOCATION]}

   b. Call the Apache Ant init task:
      
      \texttt{ant init}

5. Run the build process, either for specific parts of MEI (a., b.) or all possible build artifacts (c.):
   a. Build guidelines HTML:
      
      \texttt{ant -lib lib/saxon/saxon-he-10.5.jar build-guidelines-html}

   b. Build a specific customization’s RNG schema:
      
      \texttt{ant -lib lib/saxon/saxon-he-10.5.jar}
      \texttt{-Dcustomization.path=[/PATH/TO/YOUR/CUSTOMIZATION/FILE.xml] build-rng}

   c. Build everything (all customizations shipped with this repository, compiled ODDs for each customization, guidelines HTML):
      
      \texttt{ant -lib lib/saxon/saxon-he-10.5.jar}

Acknowledgments

This poster is based on the work of many active members of the MEI community, specifically those who attended the MEI Developer Workshop in November 2020. Without the contributions of this group, neither this poster nor the described changes to the technical setup of MEI would have been possible.

Works Cited

\begin{enumerate}
\end{enumerate}

\footnote{17} If any of the below commands return an empty string, please update or install Java or Apache Ant according to an installation instruction matching your operating system (to be found on the internet)

Contributing to the MEI Framework

In the past twenty years, the technical setup of the Music Encoding Initiative framework has been adjusted several times. Each of these transitions was motivated by the need to improve the ease with which MEI could be integrated with other formats or to address the needs of the MEI community. Consequently, there is no single right answer for all situations about the best possible technical setup for MEI.

Since MEI v0.8.0, the development of MEI is based on the ODD format as defined by the Text Encoding Initiative (TEI). This format offers better flexibility and it is possible to adjust the MEI framework to many different use cases. However, the flexibility comes at the price of a certain complexity. This poster intends to give an overview of where to find different types of information and how to contribute to MEI.

Main Repository

The main repository for MEI is located at https://github.com/music-encoding/music-encoding. This is where all the sources for MEI can be found. Information on how to interact with the MEI framework can be found in the repository. The repository can also be used to track issues and pull requests, discuss proposals, and express support for new development in MEI. However, the repository is not intended to be a development site for MEI; development is conducted elsewhere.

Guidelines

The guidelines are automatically generated from the main repository and uploaded to the https://github.com/music-encoding/guidelines repository. This development of the guidelines has been conducted in the repository; however, any updates to them are also discussed on the MEI mailing list or on Slack. The guidelines are also available on the MEI website and can be downloaded as a PDF.

Procedures for Changes

Once per month on the last Friday in odd months and the last Thursday in even months, everyone is invited to join the so-called ODD Fridays. During these meetings, open issues and pull requests for ODD are discussed and voted on. On a dedicated Slack channel, everyone interested can participate in discussions leading to consensus. The proposal will then be merged into the development version of ODD, from which the next release will be based. This is a semiannual schedule; however, in special cases, urgent issues can be discussed and addressed as a special meeting. The MEI mailing list and/or Slack channels are used for these discussions.

The MEI guidelines are automatically generated from the main repository and uploaded to the https://github.com/music-encoding/guidelines repository. While development of the guidelines was formerly conducted in this repository, now it serves only as an archive for generated content, and one must not edit the files there. The guidelines embedded into the MEI website are also drawn from this repository. Since these guidelines are made available again, it is available on the website, but will not be updated in the repository as well.
Font Design of Psaltic (Byzantine) Notation for Greek Musical Repertoires

Nikolaos Siklafidis  
Aristotle University of Thessaloniki, Greece  
siklafidis@gmail.com

Maria Alexandru  
Aristotle University of Thessaloniki, Greece  
malexand@mus.auth.gr

Abstract
The possibility of rendering scores of Greek Chant repertoires from the 3rd to the 21st century A.D. with the use of the computer opens new horizons in musicological research. In this poster a synoptic overview concerning the historical development of notational types used for Greek chants is given. This is followed by a record of various fonts and software for Byzantine neumes created since 1989. The goal of the poster is to present a new font for psaltic notations of the first and second Christian millennium, displaying the great variety of signs, neume families and neume combinations which are encountered in musical manuscripts and theoretical treatises of the Psaltic Art. These by far exceed the Byzantine neumes found today in the unicode system.

The future development of suitable software can facilitate interdisciplinary studies with other traditions and contribute to the communication between musicologists and musicians belonging to different areas of expertise.

Introduction
This poster aims at presenting a new font for writing Byzantine neumes of various periods. Although different fonts for Byzantine notations already exist, there are still many signs and sign combinations of the older layers of Byzantine neumations which have not yet been designed. This is why we would like to proceed to a more thorough inventory of the extant neumes in their historical evolution and to the creation of a more complete font, in order to allow for a faithful transcription of pieces in various notational types, for their edition, analysis, and interpretation.

1 Overview of the History of the Notation of Psaltic Art
Christian hymnography in Greek language has been foreseen with different kinds of musical notations sporadically during the first millennium A.D. and at large scale from the 10th century onwards. The various notational types expose highly interesting interconnections to each other and, as ‘living art tools’, they have undergone a complex history of evolution (see Table 1) [22].

Some of the most important types of notation used in connection with Greek hymns are the ancient Greek alphabetical vocal notation occurring in the Papyrus Oxyrrhynchus 1786 (2nd half of the 3rd century A.D.), the theta and diple notations, the Sinaitic notation (ca. 9th–12th century), and the two great types of main-stream Byzantine notations: ekphonetic and melodic.

The latter has undergone an impressive development history since the 10th century until today. It comprises the Palaeobyzantine notations (10th–12th century, Coislin/Hagiopolite and Chartres/Athonite, and with mixed forms), the Middle Byzantine notation (ca. middle of the 12th to middle of the 19th century), and the notation of the so-called New System (since 1814/15 until today). Besides these, the 14th-century manuscript Kastoria 8 with its famous red hyperstases (extra large signs referring to entire formulas and their cheironomy) also features a highly interesting notation based on older layers of notations for melismatic repertoires.

The neumes of the various notational types exceed the number of 600. If we take into account also the most characteristic neume combinations (the so-called theses, around 80 for the Middle Byzantine notation) as well as the different graphic variants of the neumes, a very impressive corpus of music signs emerges. This corpus still waits, in many aspects, for thorough apprehension and codification.
<table>
<thead>
<tr>
<th>Notation</th>
<th>Examples</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ancient Greek alphabetic vocal notation</td>
<td>(Papyrus Oxyrrhynchus 1786, 2nd half of 3rd century A.D.)</td>
<td>[22, p. 228]</td>
</tr>
<tr>
<td>Notation in Papyrus Berolinensis 21319 (6th–8th century)</td>
<td></td>
<td>Sarischouli, <em>apud</em> [22, p. 232]</td>
</tr>
<tr>
<td>Hermoupolis notation (ca. 8th century)</td>
<td></td>
<td>Papathanasiou &amp; Boukas, <em>apud</em> [22, p. 233]</td>
</tr>
<tr>
<td>Sinaitic notation</td>
<td>(Codex Liturgicus Sinaiticus, 9th century)</td>
<td>[1, tables 14 and 17]</td>
</tr>
<tr>
<td>Theta notation (10th century)</td>
<td></td>
<td>[22, p. 138]</td>
</tr>
<tr>
<td>Protoekphonetic notation</td>
<td>(Uspenskij Gospel, A.D. 835)</td>
<td>[22, p. 130]</td>
</tr>
<tr>
<td>Palaeobyzantine notation</td>
<td>(Laurensi Γ 67, f. 101r, beginning of 11th century, mixture of Coislin and Chartres [Floros])</td>
<td>[22, p. 266]</td>
</tr>
<tr>
<td></td>
<td>(Vatopedensis 1488, f. 117v, 2nd half of 11th century, Coislin V [Floros])</td>
<td>[22, p. 267]</td>
</tr>
<tr>
<td>Early Middle Byzantine notation: <em>symphona</em> and signatures</td>
<td>(Theoretical text, Petropolitanus gr. 495, f. 2r, 1st half of 13th century [Gertsman])</td>
<td>[22, p. 313]</td>
</tr>
<tr>
<td>Late Middle Byzantine notation</td>
<td>(Atheniensis 2458, f. 11r, A.D. 1336)</td>
<td>[22, p. 308]</td>
</tr>
<tr>
<td>Late Middle Byzantine notation: <em>emphona</em>, i.e., interval signs, in various combinations and modulation signs (<em>phthorai</em>)</td>
<td>(Theoretical text, Docheiariou 338, f. 2r, A.D. 1767, and Vaticanus Barberinus graecus 300, f. 3r, 15th century [Floros])</td>
<td>[22, p. 321 and 336]</td>
</tr>
<tr>
<td>Asmatikon: hyperstases (signs showing entire formulas and their cheironomy)</td>
<td>(Kastoria 8, f. 58v, 14th century)</td>
<td>[22, p. 526]</td>
</tr>
</tbody>
</table>

*Table 1:* Some examples of neumes belonging to different notational systems and categories.
2 Brief History of the Written Record of Psaltic Notation in the Computer Age (Fonts and Software)

The first use of the computer in research about Byzantine music was made by the musicologist Nana Schiødt in Copenhagen in the 1970's. In this case, no font was used, as the computer served to find basic principles in melody construction of Byzantine chant with an algorithm [15].

The creation or existence of a font that contains the signs of psaltic notation is essential in order to show and encode the signs through the computer. An exception is the use of coloring apps, where the user can directly draw the signs using the mouse, his/her hand, or an e-pencil on touch screens. In the case of using a font, it is necessary to use also another software, so that the user can write a digital score with the musical and poetic text in an environment with several visualized commands. The first efforts in this direction were usually related to the creation of fonts and at the same time to the development of software for writing Byzantine music.

The first font for Byzantine Music was prepared by Dimitrios Gianellos in 1989, with the parallel creation of a software called “Byzantinus” [25]. Between 1994 and 2002 Byzantine writing software was implemented by Vellissarios Gezerlis, who created the “Byzwriter” (Byzantinographos), of which we have the second version today [24]. The program supports writing musical texts in the New Method and a small selection of signs belonging to the older notation.

An important step in the evolution of fonts with historical scope and including the signs of older layers of Byzantine musical notation is due to Gregorios Stathis. He started in 1995, and his font was directly related to the standard of ELOT 1373 (Hellenic Organization for Standardization), which was later included in the unicode system [28, 29, 30, 31, 32, 33, 34]. There were 246 characters in this proposal, and the last documentation was produced by Nick Nicholas in 2006.

Ioannis Arvanitis implemented his own fonts with the extended notation of the New Method, which he used in his book The Akathist Hymn (1997) [23]. The typeset comprises two fonts with a total of 121 signs. Between 1999 and 2002, a similar font was created by the rev. Konstantinos Terzopoulos of blessed memory. This font, with the name “Ephesios”, was inspired by Petros Ephesios’ historical editions, issued 1820 in Bucharest, and contains also some signs of the older neumatic layer.

The NEUMES Project is an acronym for Neumed & Ekphonetic Universal Manuscript Encoding Standard. The project had two phases: October 2001 – March 2003 and March 2005 – August 2007 (extended to February 2008). The main goal was to design and build a software infrastructure for digital transcription and description of medieval chant manuscripts (including psaltic notation) [4]. This represents a very important approach for the encoding of neumes belonging to various traditions by using XML formats. In this context, the contribution of Annalisa Doneda should be mentioned, which focuses on the hyperstaseis of the Manuscript Kastoria 8 [6].

In 2005, Ioannis Vamvakas implemented his own open source package for writing Byzantine music in the TeX program language and, assisted by Panagiotis Kotopoulis, created a special font [21]. Another effort, which is also related to missionary action in America, was made in 2006 by Stefanos Souldatos and father Ephraim Arizonitis with the ED, EZ, and BZ packages, basically for the neumes of the New Method, working in the context of the OfficeSuite. Roughly in the same period, P. Bakalis developed his “Byzantini Kalamos”, a software programme for the neumes of the New Method [3].

In 2008, Savvas Papadopoulos implemented a program of Byzantine music writing with the name “Melodos”. The latest version of this software is “Melodos 2013”. This program refers to the New Method and contains one font with 151 signs, which, however, allows for the rendering of many sign combinations. It also offers, among other features, a facility for optical music recognition and a midi player for scores with Byzantine notation. In 2009, Dimitrios Manousiakis released the software “Virtual Psaltica”; two years later, Petros Moustakas

implemented the program “Pandouris”. In 2012, Erik Ferguson developed another software called “Byzscribe: A program for scribing Byzantine chant”, written in the Racket programming language. One year later, Demetrios Papadopoulos released version 4.3.3 of his open source tool “Musical texts”, a very user-friendly program, which is on version 6.5 today.

In 2016, Charalampos Cornaros presented the third edition of his TeX package for Byzantine music writing. As part of this package, a graphical user interface (GUI) by Zisis Tsiatsikas was implemented to facilitate its use. It is important to mention that there are examples in old notation created with this package which are very close to the notation found in the musical manuscripts themselves.

Another font called “Chourmouzios”, containing neumes of the New Method, has been used in a series of editions issued by the Society United Romiosyne [26]. Most recently (2020), George Douros prepared documentation for the rendering of Byzantine music with the standards of the Unicode system and OpenType, including fonts such as “Symbola” (prepared by the aforementioned author) or “Garamond-free” (by Daniel Benjamin Miller and Bob Tennent †), and the open source software LibreOffice. John Daly created the open source software “Kassia” on Github in 2016, which was then improved by Trevor Bullock and others in 2020. This program uses Python and XML.

Finally, Petru Dimitriu and Vasile-Ion Manta developed a scorewriter application in 2018 that allows, among others, to produce transcriptions from the Byzantine notation of the New Method into staff notation [5].

The fonts and software for Byzantine chant we have seen above could be categorized into three groups: fonts and software with a) neumes of the New Method, b) neumes of the New Method and of older notational layers of Byzantine Chant, and c) Byzantine neumes in a wider context of various neumatic notations. While today the first group presents a wide gamut of choices, the second one still needs joint efforts in order to provide – as far as possible – a complete inventory of the extant neumes of the different psaltic notations across the centuries. The present poster is a step in this direction. It is a desideratum that Byzantine notation, both in its older and more current layers, be incorporated in the format of the Music Encoding Initiative (MEI) in the future, and we hope that this paper can contribute to that end.

3 Attempt to Systematically Record and Design the Neumes

Continuing previous work in the field, our attempt is to carefully record the neumes of the various types and stages of neumatic notation occurring with Greek hymns and troparia of the 3rd–21st century. This is done with the help of music theoretical writings (neume lists and treatises), the existing palaeography manuals, and on the base of representative musical sources for each type of notation.

To this end, the simple neumes will be collected as basic digits, along with their main graphical variants. Afterwards, the various combinations made of simple neumes, both as neume constellations and ligatures, will be systematized, again together with their main graphical variants. The systematic mapping of the ‘neumatic landscape’ will include all categories of musical signs, such as the neumes for sounds, musical rendition (rhythm, ornaments, etc.), modal signatures, and modulation signs. For the creation of the font, the different types of historical classifications of neumes will be taken into account in order to match the different signs with their appropriate names according to the notational type to which they belong. Table 2 shows an example of

12 See, for example, [2, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 27, 35].
neume family, along with different graphic variants of the signs, in order to illustrate the immense richness of
neumatic shapes and some basic challenges for the creation of a complete psaltic font.

<table>
<thead>
<tr>
<th>Neumes</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parakalesma</td>
<td>The family of the <em>parakalesma</em> in the fully developed, late and exegetic Middle Byzantine notation:</td>
</tr>
<tr>
<td>Heteron parakalesma</td>
<td>(the first shape is the most frequent one)</td>
</tr>
<tr>
<td>Allon</td>
<td>(another <em>parakalesma</em>)</td>
</tr>
<tr>
<td>etktstrepton</td>
<td>(inverted <em>parakalesma</em>)</td>
</tr>
<tr>
<td>diploparakalesma</td>
<td>(double <em>parakalesma</em>)</td>
</tr>
<tr>
<td>homoion</td>
<td>(and a similar double <em>parakalesma</em>).</td>
</tr>
</tbody>
</table>

[22, p. 538]

Table 2: Complex neume families and multiple graphic shapes: challenges for font design.

**Conclusion**

The immense wealth of musical signs found in sources containing Greek texts from the 3rd to the 21st century is still waiting to be thoroughly systematized in a machine-processable language by notational types, categories of neumes, neume families, and graphical variants in historical perspective. The production of a comprehensive neume inventory will allow for an efficient completion of the digital neume forms and the creation of a font that attempts to cover all previously known signs from manuscripts and prints containing Greek liturgical chant through the centuries. Further steps will include the creation of a rendering software for psaltic neumes. This can lead to the preparation of a proposal for an MEI Schema. In turn, this will promote editorial work, transcriptions, and analyses of Byzantine chant as well as many new interdisciplinary studies.

**Acknowledgments**

We would like to thank Prof. Eleanor Selfridge-Field for valuable bibliographical advice.

**Works Cited**


Works in Greek
[26] Perisinakis, Dimitrios E. (Περισσάκης, Δημήτρης Ε.), ed. Κοινωνικά παλαιά διδασκαλίας: ήτοι κοινωνικά μελετήθηκαν υπό παλαιώς διδασκαλίας και εξηγήθηκαν εκ της παλαιάς εις την νέαν γραφή para Χαραλάμπου Κωστοφάλακος [Koinonika by Old Masters, that is Koinonika Composed by Old Teachers and Transcribed from the Old Method to the New One by Chourmouzios the Archivist]. Thessalonikion: Enomeni Romiosinis, 2014.
[28] Stathis, Gregorios (Σταθής, Γρηγόριος). "Γρηγόριος Κολλέζιος - της 1ης καθηγητός της ψαλτικής τίμης και της νέας ακαδημαϊκής μεθόδου [Graphic Arrangement of the Neumes of the Byzantine Middle Notation and of the New Analytical Method] in "...
[31] Stathis, Gregorios (Σταθής, Γρηγόριος). "Η σημειογραφία της βυζαντινής και μεταβυζαντινής μουσικής ήτοι της ελληνικής ψαλτικής τύχας, είναι το ελληνικό μουσικό αλφάβητο (C. The Notation of Byzantine and post Byzantine Music, that is of the Greek Psaltic Alphabet) in "...

178


Font Design of Psaltic (Byzantine) Notation for Greek Musical Repertories

Nikolaos Sikelakis and Maria Alexandru, Aristotle University of Thessaloniki

<table>
<thead>
<tr>
<th>Notation type and sources</th>
<th>Neume samples from the process of designing the neumes font Κωνσταντίνα: a work in progress</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ancient Greek alphabetic vocal (2nd half of 3rd cent.)</td>
<td></td>
<td>ΟΝΤΟΤΕΡΑ ΧΟΙΡΙΟΝ ΧΑ ΠΙΟΝ ΦΙΝΕΥΜΑ</td>
</tr>
<tr>
<td>Sinaitic notation (9th – 10th cent.)</td>
<td></td>
<td>ΧΑΙΡΕ, Η ΚΙΒΟΤΟΣ, Η ΝΟΕΡΑ</td>
</tr>
<tr>
<td>Elephonic notation (10th–12th cent.)</td>
<td></td>
<td>Και ιδιον εγώ μεθ' όμοιον ελμή πάσας τὰς ἡμέρας τὰς ἡμέρας έως τῆς συντελεσίας τοῦ αἰώνος, ἀμην</td>
</tr>
<tr>
<td>Palaeobyzantine notation (10th–12th cent.)</td>
<td></td>
<td>ΒΑΣΙΛΕΥ ΣΟΦΡΑ ΝΕ Ε ΧΡΙΣΤΟΣ ΓΕΝΝΑΤΗ ΣΣ ΣΟΧΤΑΤΕ</td>
</tr>
<tr>
<td>Middle Byzantine notation (12th–19th cent.)</td>
<td></td>
<td>ΔΕΥ τε προ σκυ νη σωμεν και προ στε ε σωμεν τω βασι i λει</td>
</tr>
</tbody>
</table>

Annotation of Medieval Music Facsimiles Using ‘Good Enough’ OMR

Joshua Stutter
University of Glasgow, UK
j.stutter.1@research.gla.ac.uk

Abstract

The Clausula Archive of the Notre Dame Repertory (CANDR) is an in-progress PhD project with the aim of cataloguing, transcribing and analysing digital facsimiles of the thirteenth-century repertory commonly termed Notre Dame polyphony, and a secondary aim of providing new datasets and analytical tools for studying medieval polyphony. This poster highlights the use in the project of (a) a new methodology for de-skewing facsimile images, and (b) average symbol masks in an OMR–enhanced workflow with an emphasis on creating an OMR workflow that is ‘good enough’ to accelerate the annotation of an image dataset of particularly transitional notation.

1 Introduction

As far as medieval repertories go, it is difficult to locate any that is more significant to the development of ‘Western music’ than the Notre Dame (ND) school, a thirteenth-century collection of hundreds of pieces of polyphony on Gregorian chant. These settings of polyphony were likely initially intended for performance during services at Notre Dame de Paris, but it is now recognised that this particular style of polyphony spread throughout Western Europe in the mid to late thirteenth century [4].

This repertory is vitally important to the understanding of the nature of performance, rite, notation and the concept of ‘composition’ in the medieval period. However, those that do not study it in any detail may have only heard of this repertory historiographically: an important juncture in the history of the ‘Western canon’, where the quantum leap from monophony to polyphony, and from oral to literate was made, but often only seen as a fleeting reference before moving onto more compositionally stable ground.

A picture of the ND repertory as a period of medieval musical revolution of composers over improvisers is obviously a simplified one, but it is a view frequently taken even by recently–written history and reference books. For example, Edward Roesner’s article for Grove characterises ND polyphony as “composition in the modern sense” [9, p. 202–203]. The ND school therefore becomes a useful jumping–off point to introduce the concept of written rather than oral practice, as well as the idea of named composers rather than anonymous improvisations. As a result, study of the ND school has been burdened with the anachronistic requirement to consider the ramifications of the first ‘composers’, ‘compositions’ and written process in Western music, responsibilities somewhat undeserving of a repertory which is continually being shown to be influenced by oral processes [1, 13].

2 Research Context

Rather than laying the blame for the elevation of the ND school as the ‘crucible’ of Western art music squarely at the feet of music historians, the construction of the ND repertory to be a progression towards classical music in fact emerges from the field’s overreliance on the ideas of Friedrich Ludwig, concerning the creation and dissemination of the ND repertory. It is almost impossible to consider the topic of Notre Dame polyphony without Ludwig’s Repertorium being mentioned [7].

Now over a century old, Ludwig’s monumental work on the ND repertory dominates current thought as the key text to understanding the creation and dissemination of the repertory in manuscript. Repertorium catalogues the settings of polyphony in the central manuscripts and draws concordances between settings and between the genres of organum, clausula and motet, purely through textual apparata and an array of sigla geared towards canonicalisation.
Although *Repertorium* is a striking piece of scholarship that has deservedly stood the test of time for its attention to detail and scrutability, Anna Maria Busse Berger, among others, has criticised Ludwig's biases, adherence to written composition and tendency towards *Urtext* [2]. Busse Berger rightly finds fault in Ludwig's crucial role in constructing the repertory from an unknown set of jumbled manuscripts and turning it into the missing link in the ‘progression’ from monophony and improvisation to polyphony and written composition, which would in Ludwig's view ultimately culminate in Palestrina. Although Ludwig's biases are now well known, Ludwig's conception of the ND repertory as fixed, written composition pervades scholarship in this area and, perhaps unsurprisingly, his flawed ideas are still often taken as central tenets of the repertory. Work must be done to unpick Ludwig's ideas as objectively as possible and re-evaluate the ND repertory outside of the modern Western canon.

3 **CANDR**

The *Clausula Archive of the Notre Dame Repertory* (CANDR) aims to deconstruct Ludwig's concept of the ND repertory and begin the process of concordance once again from the source manuscripts as they appear in facsimile. CANDR's aim is to untangle some of these interrelationships not as Ludwig did by hand and sheer force of repertorial knowledge, but by quantifiable methods.

The central manuscripts of the ND repertory, F (476 folios), W1 (214 folios), and W2 (253 folios), have been digitised and released online under permissive licences by their host institutions. This project takes advantage of these facsimile images as the base from which to generate a suitable dataset for calculating concordances. In order to extract the notational content from these images and analyse their interrelationships, the project uses a bespoke tool for automatic transcription from tagged annotations of staves of music. This proceeds in three phases:

1. Creation of an online database and annotation-transcription tool to aid the input of a dataset of thirteenth-century polyphony (complete).
2. Input of the dataset using these tools (ongoing).
3. Design and creation of tools to analyse this dataset dynamically (not yet begun).

Most of these tools had to be created particularly for this project, as many preexisting tools have no architectural support for the extremely contextual, transitional and inherently pluralistic notation of the ND school [11], which depends on performance practice, written and oral transmission, scribal practice, and was neither written for nor is suitable for performance. In creating a more flexible framework for ND notation, execution of these first two stages have presented two broad issues that have had to be solved. Firstly, the three-dimensional irregularity of the facsimile images caused difficulties during the annotation process, causing the automatic transcription to be inaccurate and affine de-skewing to be unsatisfactory. Secondly, the effort that would be required to transcribe the manuscripts (943 folios total) by hand was not feasible in any reasonable period of time.

4 **Regularisation**

By far the two most common issues in the annotation of medieval facsimiles are (a) curved parchment from the binding not allowing the folio to rest flat during photographing; and (b) perspective effects caused by the camera not being placed directly above the surface. These issues make it impossible to, for example, trace a straight, horizontal staff line across the staff as the line may be curved, and one side of the staff may appear larger than the other as it is marginally closer to the lens.

Previous work in this area takes two divergent paths: either to attempt to de-skew staves using shearing and other affine transforms on the facsimile image [5], or to take the image as-is, and consider skewed elements such as staff lines to be a property of the image, electing to model them as smoothed and reconstructed curves [12]. CANDR takes a new approach, using a perspective transform in multiple rounds to de-skew images to yield straightened staves. To be precise, ‘de-skew’ is not the correct terminology here, as ‘skew’ relates particularly to two-dimensional affine transformations (translation, scale, rotation, shear). The perspective transform used in
CANDR is a three-dimensional projective transformation, capable of not only transforming any parallelogram, but any arbitrary quadrilateral.

Figure 1: A fully-annotated stave: horizontal stafflines (dark blue), C clef (light blue), vertical divisiones (yellow), notes and ligatures (orange and red respectively).  

First, the bounding quadrilateral of a system is manually annotated on a facsimile image, and an inverse perspective transform matrix is calculated that can transform that quadrilateral into a straightened rectangle. A second bounding quadrilateral for each stave on that system can be manually annotated upon the straightened system, and the same process applied to create individual straightened and cropped stave images. This process is integrated into the annotation workflow of CANDR and is calculated on-the-fly: annotating a system or stave creates a new object in the database along with its bounding quadrilateral. Viewing a system or stave in the visual editor is to view that item transformed into a straight rectangle using this transform. This method (a) mitigates the effects of parchment bend significantly, and (b) can eliminate not only rotation and skew but also perspective effects in the facsimile. There is also the added advantage of presenting to the user a clear and straightened image.

Figure 2: A real-world example: Before transform of four-line staff with staff boundaries and target rectangle marked (top), and after transform (bottom).
5 ‘Good Enough’ OMR-enhanced Transcription

Regularising the staff images made OMR approaches to annotation much more viable. To this end, CANDR uses a standard pixelwise classification using a convolutional neural network (CNN) and a sliding context window [3]. However, where CANDR differs from other approaches is in the purposefully limited scope of its OMR methodology. Often, approaches using neural network methodologies aim for pixel-perfect accuracy for ground truth at a variety of sizes over numerous manuscripts as one small step towards creating accurate reader- or researcher-friendly transcriptions.

Conversely, CANDR’s primary input method is manual annotation. As such, creation of OMR-enhanced transcription was desired simply to speed up manual annotation by automating repetitive actions (such as note and staff line positions) which can then be corrected manually. As a result, it is necessary to acknowledge the limitations of both the data and the methodology. Pixel-perfect accuracy requires a pixel-perfect input which is time consuming to create and maintain even with modern pixel classification systems [10]. Such accuracy is not feasible nor necessary with CANDR, which aims more narrowly for ‘good enough’ automatic annotation, and a transcriber-centric approach.

CANDR models staff items instead as simplified geometric shapes: clefs, accidentals and syllables are modelled as rectangles, notes are modelled as single points, and staff lines and divisiones are modelled as lines. This approach is similar to the methodology of using convex hulls as ground truth, but with the further simplification of those convex hulls constrained to simple 1D and 2D polygons [6]. These items are traced directly onto the straightened staves using the visual editor both as annotation and ground truth for OMR. Then for each stave, average symbol masks are generated for staff lines and other items. CANDR conceives of staff lines as a separate layer, onto which other items are placed. Two almost identical CNNs were created: one to classify pixels as staff lines, and another to classify pixels as other items. The separation of these items into two models allows pixels to have two simultaneous and independent classifications: a pixel can exist as a staff line and staff item, and each model can be tuned to better extract its features.

The software that generates these average symbol masks contains parameters for altering its output, such as staff line/divisione width and note radius, which were found by hyperparameter tuning but also differ depending on the window size of the classifier. These symbol masks are not a simple binarisation, but the edges are ‘feathered’ to better reflect the average matching of mask to item using supersampling.
The CNNs were implemented as binary and categorical classifiers for staff lines and staff items respectively. However, rather than binarising the output, the confidence of the prediction for each pixel was multiplied over a radial blur with shape equal to the size of the classifier window, followed by a round of despeckling and Otsu thresholding [8]. The dimensions of most 2D items were detected by calculating the bounding boxes of a connected component analysis, but the 1D staff lines and divisiones were instead defined as the coordinates of the furthest extents of each component. To reduce false positives, two further constraints were added: Staff lines must have an angle approaching the horizontal, and must have a width larger than half the width of the image, and, secondly, divisiones must approach vertical and must exceed a minimum length.

Notes, however, could not be constructed in the same way as they are modelled as single points, and are often connected into ligatures. Ligatures often resulted in single components on the heatmap. After thresholding, an iterative process of binary erosion was applied to the note heatmap, with the aim of creating as many connected components as possible that were larger than the despeckle parameter. This erosion parameter was found using a binary search.

By not aiming for pixel-perfect accuracy, timing tests indicate that CANDR's use of average symbol masks decreases the time taken for annotation of staves of polyphony threefold over purely manual effort. The fundamental outcome is that, rather than having to annotate staves by tracing items from scratch, transcribing staves of ND polyphony using CANDR is now, much more simply and quickly, a process of checking and correcting OMR.

CANDR is released under AGPLv3 at: https://gitlab.com/candr1.
Works Cited


Annotation of Medieval Music Facsimiles Using “Good Enough” OMR

Joshua Stutter
University of Glasgow

Abstract

The Clausula Archive of the Notre Dame Repertory (CANDR) is an in-progress PhD project with the aim of cataloguing, transcribing and analysing digital facsimiles of the thirteenth-century repertory commonly termed Notre Dame polyphony, and a secondary aim of providing new datasets and analytical tools for studying medieval polyphony. This poster highlights the use in the project of (a) a new methodology for de-skewing facsimile images, and (b) average symbol masks in an OMR-enhanced workflow with an emphasis on creating an OMR workflow that is “good enough” to accelerate the annotation of an image dataset of particularly transitional notation.

Context

• Notre Dame (ND) polyphony is unfairly elevated & portrayed as the “crucible” of Western art music.
• The field currently relies heavily on the outdated ideas of Friedrich Ludwig’s Repertorium (1910).
• CANDR aims to reconstruct ND concordances from scratch.
• Usual affine de-skewing methods are unsatisfactory.
• 943 folios of music must be annotated by hand.

1. Regularisation

CANDR’s regularisation process was designed to remove not only stave shears and rotations but also perspective effects caused by parchment bend and/or offset camera placement. Previous work either: (a) attempts to de-skew staves using shearing and other affine transforms on the image, or (b) considers skewed elements such as stave lines as a property of the image, electing to model them as smoothed and reconstructed curves. CANDR takes a new approach, using a perspective transform in multiple rounds to de-skew images to yield straightened staves.

2. “Good enough” OMR

CANDR uses a standard pixelwise classification using a convolutional neural network (CNN) and a sliding context window. However, the primary input method of the 943 folios is manual annotation and as such, creation of OMR-enhanced transcription was desired simply to speed up manual annotation by automating repetitive actions which can then be corrected manually. Rather than aiming for pixel-perfect accuracy, CANDR models staff items as simplified geometric shapes: clefs, accidentals and syllables as rectangles, notes are modelled as single points, and staff lines and divisiones are modelled as lines. This approach is similar to the methodology of using convex hulls as ground truth, but with the further simplification of those hulls constrained to simple 1D and 2D polygons. For each stave, average symbol masks are generated for staff items. The software that generates these masks contains parameters for altering its output, such as linewidth and note radius, which were found by hyperparameter tuning. These symbol masks are not a simple binarisation, but the edges are “feathered” to better reflect the average matching of mask to image using supersampling. The dimensions of most 2D items were detected by calculating the bounding boxes of a connected component analysis, but the 1D items were instead defined as the coordinates of the furthest extents of each component.

Notes however, could not be constructed in the same way as they are modelled as single points, and are often connected into ligatures. Ligatures often resulted in single components for multiple notes. To ameliorate this issue, after thresholding and applying an iterative process of binary erosion to the analysis with the aim of creating as many connected components as possible that were larger than the despacce parameter, this was found using a binary search. Timing tests indicate that CANDR’s use of average symbol masks decreases the time taken for annotation of staves of polyphony three-fold over purely manual effort. The fundamental outcome is that, rather than having to annotate staves by tracing items from scratch, transcribing staves of ND polyphony using CANDR is now a process of checking and correcting OMR.

First the bounding quadrilateral of a system is manually annotated on a facsimile image and an inverse perspective transform matrix is calculated that can transform that quadrilateral into a straightened rectangle. A second bounding quadrilateral for each stave image is created an OMR work flow of CANDR and is calculated on-the-fly: annotating a system or stave creates a new object in the database along with its bounding quadrilateral. Viewing a system or stave in the visual editor is to view that item transformed into a straight rectangle using this transform. This method (a) mitigates the effects of bend and skew significantly and (b) can eliminate not only rotation and skew but also perspective effects in the facsimile. There is also the added advantage of presenting to the user a clear and straightened image.

A contrived example for demonstration (original on top, transformed on bottom): Note particularly how the page is straightened but none of the image edges are parallel or perpendicular.

A real-world example: Before transform of four-line staff with boundary rectangles marked (top), and after transform (bottom).

A fully-annotated stave: horizontal stafflines (purple), C-clef (teal), vertical “divisiones” (blue), notes and ligatures (orange and red respectively).
Encoding Scores for Electronic Music

Florian Zwißler
Osnabrück University, Institute for Musicology and Music Pedagogy, Germany
florian.zwissler@uni-osnabrueck.de

Philip Schwarzbauer
Osnabrück University, Institute for Musicology and Music Pedagogy, Germany
philip.schwarzbauer@uni-osnabrueck.de

Michael Oehler
Osnabrück University, Institute for Musicology and Music Pedagogy, Germany
michael.oehler@uni-osnabrueck.de

Abstract

A perspective on the specific issues of music encoding dealing with Electronic Music is presented. In many cases the works to be discussed exist in a fixed media format and hence no prescriptive score is necessary to facilitate a ‘valid’ performance. While there are a number of descriptive scores for pieces of Electronic Music, these are to be treated differently, as they are purely aimed at analysis and therefore contain a certain information bias. Data that is more comparable to instrumental scores is contained in rare examples of so-called realization scores. It is argued that these realization scores can be identified as the main subject for encoding of Electronic Music works. For this we will discuss an example from one such score by Karlheinz Stockhausen. For his piece KONTAKTE, Stockhausen released a realization score that unfolds a very detailed documentation of all steps made within the studio production of that work, including the complex patching of studio devices and the specific transformation processes achieved by the use of tape machines. The paper presents an approach to formalize and encode all these steps within the framework of a semantic database. Using technology like the semantic web standard, Linked Data and the corresponding RDF/OWL framework, an Electronic Music production setup and its usage can be encoded, stored, and analyzed.

1 Introduction

In the field of digital music encoding, there are increasingly different research perspectives, methods, and topics in recent years. Frequently, phenomena related to Common Western Notation [17, 18] or notation systems of the Renaissance and Medieval periods [2, 23] are addressed. However, examples of Electronic Music are not very strongly represented. (Here and in the following, the term Electronic Music is intended to refer generally to electronically produced, composed new music and to explicitly include genres such as musique concrète.) Only a few examples can be found, and most of them are oriented more towards the field of Music Information Retrieval [13, 19] and/or analysis [7, 20]. The main reason for this could be found in the nature of Electronic Music that is – if not realized within a live-electronic setup – predominantly existent within a fixed media format, be it on magnetic tape in the earlier days of the genre or any type of digital data storage today. Instead of having to be realized within a performance situation that implies musicians reading and playing or singing the notes of a score, the ‘tape’ (we will stick to this term, including all digital variants) is realized within a production situation in a studio for Electronic Music. (This studio may today of course also be realized virtually within a software framework – our examinations and results will be valid for either type of situation.)

The absence of a prescriptive score implies a problem in tracking informational resources for any of these pieces, as basically all there is to be analyzed is the resulting audio material that can, e.g., be dealt with by classic methods of visualization. The well-known examples here would be the amplitude-depiction of a waveform display or the frequency-based sonogram. Both follow the reading logic of a score, offering a time axis to read along and in the case of the sonogram even an equivalent vertical tonal dimension with the analogous assignment of ‘low’ and ‘high’. In addition, a considerable amount of work has been put on methods of transcription to generate scores that enable a listener to follow the piece with a corresponding visual counterpart [4, 28]. Evidently, the status of such a descriptive score differs significantly from that of a prescriptive one: In spite of the
various attempts to gain access to an ideally unified language for such transcriptions, the *a posteriori* nature of it will always imply a bias imposed upon the written result by the inevitable pre-assumptions of each transcriptor. This is, of course, an analogy to the margin of interpretation that is at the disposal of each instrumentalist playing the notes of an instrumental score. It is evident that these transcribed scores are therefore not an apt material to be taken into consideration as a database. The question naturally arises whether this inherent lack of *a priori* information in Electronic Music is to be taken for granted, that is, to basically just rely on empirical methods of information retrieval or, if there are ways to do so, on a piece under examination.

2 Scores for Electronic Music

2.1 Background: Early Scores of Stockhausen's Electronic Pieces

The early period of Electronic Music saw a multitude of approaches to this new genre, ranging from the empirical techniques of the *musique concrète* to the more pre-determined approach of the works from the Studio for Electronic Music of the West German Radio (WDR) in Cologne. The latter, among others represented by Karlheinz Stockhausen, provides a few interesting examples that prove the composers' reflections on the relevance of scores for their new creations.

An early example of a full-fledged score for a piece of Electronic Music is Stockhausen's *STUDIE II* from 1954 [25]. This score not only provides enough information to follow the piece by reading, it is also one of the rare cases where enough information is offered to actually re-realize the piece. As the introductory notes of the score suggest, this score was meant to be used to create further versions of the piece. This clearly indicates that the concept of a piece existing solely in one finished version – usually realized by the composer – on fixed media was not quite decided yet within the then context of Electronic Music. Due to this, a variety of new interpretations of the piece exists, for example, as a realtime patch running on the Max/MSP platform [8] or in more hardware-oriented approaches [29]. For this paper, we will focus on a slightly more complex example of a score that opens up many facets of the production process.

2.2 The Score of Karlheinz Stockhausen's *KONTAKTE*

From 1958 to 1960, Stockhausen realized, together with Gottfried Michael Koenig, the Electronic Music for his piece *KONTAKTE*, which turned out to become one of his best known pieces. Within the five years that had passed since *STUDIE II* (which of course incorporated the composition and realization of *GESANG DER JÜNGLINGE*), the working processes in the studio of the WDR had become much more complex: Many more facets of possible usages of all devices as well as the manifold possibilities of manipulation of sound structures by the means of tape had been made accessible [6, 26].

Given this background, Stockhausen decided to fully document the working processes undertaken to realize the material for *KONTAKTE* in a so-called realization score [27]. All the crucial processes of sound generation and processing are written down precisely, as the examples in Figures 1 and 2 illustrate.

The graphic examples display how the single devices in use (whose individual symbols are thoroughly explained in an introductory section) are set up and connected. Specific production dates are identified and thereby indicate individual stages of the production. It should be noted that at this point every aspect of electronic sound generation (with exception of very simple instruments such as electric organs) had to be individually planned and performed. This situation would change within the 1960s that brought along the development of commercial music electronics and the concept of the synthesizer. This process encapsulated a lot of the processes explicitly shown here back into the body of a new type of 'instrument'. *KONTAKTE* exists in two versions: One that only consists of the Electronic Music on four-channel tape,
and one that combines this with two soloists on piano and percussion. Interestingly, Stockhausen also created a transcription of the tape as he needed to communicate the Electronic Music to the instrumentalists. The index numbers of patches and processes can be picked up from the timeline, giving the reader the opportunity to switch to a ‘second layer’ of detailed information. Comparing the graphic part of the performance score with the transcribed graphical representation and the more documentary realization score, it becomes quite clear that the latter will be the more rewarding database for an encoding process that is likely to produce valid and productive results. Evidently, the realization score is a close relative to the classic instrumental score, providing the information necessary to produce the acoustic result intended by the composer. We take this as evidence that those detailed notes should serve as our database. The samples from Stockhausen’s realization score give an impression of the type of information offered. For this article, just these very simple instances were chosen, the score of course also offers much more complex settings on its more than 60 pages. The first example shows a typical setup to generate sounds with filtered impulses. In the second, a unique technical idea to realize rotations of sound that could be put down on a four-track tape machine is depicted. On the one hand, the examples show the potential relevance of a semantic interface to electronic processes of sound generation; on the other hand, they illustrate the challenges that can arise when encoding score data of electronic music. In the following chapter, we propose an approach for this based on linked semantic data encoding. For this article, we choose to encode the second example, as the rotation apparatus is probably furthest from a classic understanding of musical or instrumental parameters. This qualifies it as a suitable object to test our encoding concept.

3 Encoding the Rotation Table from Karlheinz Stockhausen’s KONTAKTE

The ‘rotation table’ presented a simple and convincing possibility to render spatial movements into a fixed form on a multi-channel tape and thus at the same time to integrate them as a new parameter into the process of composition. A mono signal is sent to a directional loudspeaker mounted on a rotating table (Figure 3). Using manual movements, the signal can thus be made to actually rotate in the studio space, and this movement is recorded with four microphones mounted symmetrically at an angle of 90 degrees. By recording their signals on a four-channel tape and reproducing them in an equally symmetrical quadraphonic square of loudspeakers, the effect of the rotation is reproduced quite convincingly.

In the lack of normalized standards and information processing formats, a comprehensive encoding and storage of this realization score resulting from the complex development process is hardly possible. The general lack of an established documentation format for Electronic Music finds a shining example at this point. To close this methodological gap, a data structure is needed that does adequate service to the hybrid of technical and artistic-aesthetic nature of Electronic Music production. In fact, this results in an encoding framework for Electronic Music, which will be described and evaluated in this paper.

3.1 Ontological and Semantic Approach

The networked structure and the connected, interacting technical setup of an Electronic Music production places a high demand on the Linked Data [24] structure. Realization scores for Electronic Music, as shown in the previous chapter, call for a technical orientation of the stock. In addition, a complex and heterogeneous
data set must be expected, since there are no standards for scores in the analog context either. These requirements can be met with the help of a semantic database according to the RDF standard\(^1\) based on a specialized ontology. Such a semantic database models the data fed to it as a knowledge graph, as is done, for example, in the music databases of DBtune.org.\(^2\) Semantic databases and ontologies also have a certain legacy in musicology as exemplified by the Music Ontology approach\(^{[22]}\) and its extending frameworks.

For the creation of a semantic database, a specific ontology must first be defined as a basic structuring order that describes the nomenclature and data type of the occurring data sets or objects. In the case of Electronic Music scores, fundamental preliminary work is required here, since terminologies are lacking in many areas. The need for systematization has been addressed in various places\(^{[3, 9, 12, 16]}\). A categorization, for example, of the electronic devices in the studio fundus of the time, as comparably made by the Hornbostel-Sachs classification for acoustic instruments\(^{[10]}\), does not exist here or is poorly elaborated\(^{[15]}\).

However, an ontological framework for the representation of conventional western notation exists in the form of MusicOWL\(^{[11]}\). It extends the basic Music Ontology with data types for musical notation and thus allows semantic storage of musical notation, similar to the MEI format, using XML. Other extending ontologies like the Studio Ontology\(^{[5]}\) or the Audio Features Ontology\(^{[1]}\) provide capabilities for representing music technology functionalities. Other ontological approaches like the Device Ontology or the Connectivity Ontology offer basic technical functionalities, whereas the Timeline Ontology\(^{[21]}\) depicts a time dimension, in real-time historical or in inner-work runtime aspects. An ontological approach to Electronic Music encoding therefore works at the intersection of these technologies (Figure 4). Using these technologies and adding some new functionalities, an Electronic Music production setup can be implemented by means of a replication of the signal paths and devices. Here, the terminology and functionality is chiefly delivered by the Device Ontology and the Connectivity Ontology\(^{[5]}\). In order to avoid a documentation of purely technical structures and to assign musical meaning to the information, a concept of function, which assigns an acoustic or music technological task to sections or individual elements of the apparatus, is being delivered by this work. The function concept is multidimensional in order to be able to represent both overall and partial functions of the production devices. This results in a multi-dimensional ontological structure of the production setup, which can be functionally documented on all levels. This allows encoding of Electronic Music scores and realization descriptions and can even be linked to various other ontologies.

### 3.2 Encoding

Semantic Encoding of Stockhausen’s rotation table is quite straightforward with the help of the Device and Connectivity Ontology. With the exception of the acoustic signal between the loudspeaker and microphones, all signal flows can be traced and the rotation angle of the rotation table can be realized with the help of a status term. But the use of these two existing ontologies involves the problem that no statement about the

---

1  https://www.w3.org/RDF/ (accessed January 12, 2022).
function and use of the apparatus is possible. The physical-acoustic effect of the rotation is also not recognizable from the technical setup, since there is no possibility to locate the microphones to the loudspeaker. This results in a lack of expression for the composer’s intentions documented in the score.

By introducing the function concept, this missing functionality can be documented. In order to reference the effect of four-channel symmetric rotation, which can be characterized quite unambiguously by acoustic or room acoustic parameters, the above function description concept is used. Defining an input and an output module, the function type “4-channel rotation” is stretched over the whole apparatus. In the overall structure of all devices (e.g., pulse generator, band pass filter), which were used to realize KONTAKTE, a specific section can now be found as a rotation element (Figure 5).

In order to realize a multi-dimensional function description that can thus be applied to different levels of detail, the term can be used in an overlapping manner: First, “4-channel rotation” is the function of the overall circuit. Furthermore, the attribution of the function “Recording-Channel-1” for the recording process at one of the four microphones describes the sub-task of a circuit section and thus a sub-function of the overall function of the rotation table. The same procedure can be used with superordinate functions for the definition of the procedure in the production apparatus. The semantic representation of time dimensional data (like changes in device settings within the production process, e.g., the table’s rotation angle) can be realized by using the Timeline Ontology. For means of conciseness, this complex semantic encoding is not shown in the given example.

4 Conclusion
Using a semantic data representation provides new possibilities of Electronic Music encoding. Intended functionality and composers’ thoughts on the apparatus can be documented, even extending the possibilities of classical music notation. The actual target of Semantic Web and Linked Data technologies, a worldwide and cross-disciplinary connected knowledge graph [24], allows various use cases for the encoded information. As, for example, RDF datasets become readable by AI systems, a great potential beyond the mere encoding of Electronic Music scores can also be expected.
Scores of Electronic Music deserve to be added to the catalogue of encoded musical notation as a valuable expansion. They confront us with new problems as to how to encode specific processes of electronic sound generation. These processes are not only relevant within a limited focus on production processes from a very specific area of music history. Rather, through the vast development of music electronics in the 20th century and within the contexts of digitization of all levels of music production, they have become interwoven with a multitude of aspects within present day music theory and practice. Therefore, the aim of encoding these processes is clearly justified.

The presented approach of semantic encoding by means of an ontology has proven to be a sufficiently powerful tool to deal with the information available. A resulting database on the foundations of this ontology could in a future scenario even provide possibilities for automated remodelling of the device setups shown. This could (as in the example cited in chapter 2.1) for example be realized on a platform like Max/MSP. Sound generation processes that were previously difficult to comprehend could thus be exemplified within the experimental setting itself.

Acknowledgments

This work was supported by the German Research Foundation (DFG) within the PRESET project (grant number 424734041).

Works Cited


Encoding Scores for Electronic Music

Background: The Score of Stockhausen’s KONTAKTE

In the field of digital music coding, there are increasingly different research perspectives, methods and topics in recent years. However, examples of Electronic Music are not very strongly represented. The main reason for this could lie in the nature of this music that is predominantly existent within a fixed media format. Instead of having to be realized within a performance situation that implies musicians reading and playing or singing the notes of a score, the “tape” is realized within a production situation in a studio for Electronic Music. This absence of prescriptive scores implies a problem in tracking informational resources for any of these pieces, as basically all there is to be analyzed is the resulting audio material. The question naturally arises whether this inherent lack of a priori information in Electronic Music is to be taken for granted, that is, to basically just rely on empirical methods of information retrieval or if there are ways to do so on a piece under examination.

The early period of Electronic Music saw a multitude of approaches to the new genre, ranging from the empirical techniques of the musique concrète to more pre-determined approach of the works from the studio for Electronic Music in Cologne. The latter, among others represented by Karlheinz Stockhausen, provides a few interesting examples that prove the composers’ reflections on the relevance of scores for their new creations. From 1958 to 1960 Stockhausen, together with Gottfried Michael Koenig, realized the Electronic Music for his piece KONTAKTE. Stockhausen decided to fully document the working processes undertaken to realize the material for KONTAKTE in a so-called realization score [Stockhausen, 1968]. All the crucial processes of sound generation and processing are written down precisely. Graphic depictions of device setups illustrate individual stages of the production. Evidently, the realization score is a close relative to the classical score, providing the information necessary to produce the acoustic result intended by the composer. For this example we choose to encode the rotation apparatus used in KONTAKTE as this is probably most remote from a classic understanding of musical or instrumental parameters.

Semantic Encoding of the KONTAKTE Rotation Table

The rotation table presented a simple and convincing possibility to render spatial movements on a multi-channel tape by sending a mono signal to a directional loudspeaker mounted on a rotating table (Figure 1). Using manual movements, the signal is picked up by four microphones and recorded on tape, recreating a rotation effect when played back over a quadraphonic loudspeaker setup (Figure 2).

In the lack of normalized standards for scores of Electronic Music, a linked data (Sakr, 2018) structure is being used for the encoding. These semantic databases, according to the RDF standard and based on a specialized ontology, also have a certain legacy in musicology, as exemplified by the Music Ontology approach (Raimond, 2007) and its extending frameworks e.g. the Studio Ontology or the Device Ontology (Pazekas, 2011). According to this interconnected data representation system, the Electronic Music Ontology (EMON) was developed for our purpose.

Semantic Encoding of Stockhausen’s rotation table is quite straightforward with the help of the Device and Connectivity Ontology. Signal flows can be traced and the rotation angle of the rotation table can be realized with the help of a status term. But the use of these two existing ontologies involves the problem that no statement about the function and use of the apparatus is possible. The physical- acoustic effect of the rotation is not recognizable from the technical setup, due to the lack of a possibility to locate the microphones in relation to the loudspeaker. This results in the absence of any means of expression for the composer’s intentions documented in the score.

By introducing the function concept from EMON, this misinformation can be documented. In order to reference the effect of 4-channel symmetric rotation, an input and an output module for the functionality are defined and signed with the function type “4-channel rotation”. In the overall structure of all devices which were used to realize KONTAKTE, a specific function can now be found as a rotation element (Figure 3).

In order to realize a multi-dimensional function description that can thus be applied to different levels of detail, the term can be used in an overlapping manner: First, “4-channel rotation” is the function of the overall circuit. Furthermore, the attribution of the function “Recording-Channel-1” for the recording process at one of the four microphones describes the sub-task of a circuit section and thus a sub-function. The same procedure can be used with superordinate functions for the attribution of the function “Recording-Channel-1” for the recording process at one of the four microphones describes the sub-task of a circuit section and thus a sub-function. The same procedure can be used with superordinate functions for the attribution of the function “Recording-Channel-1” for the recording process at one of the four microphones describes the sub-task of a circuit section and thus a sub-function. The same procedure can be used with superordinate functions for the attribution of the function “Recording-Channel-1” for the recording process at one of the four microphones describes the sub-task of a circuit section and thus a sub-function. The same procedure can be used with superordinate functions for the attribution of the function “Recording-Channel-1” for the recording process at one of the four microphones describes the sub-task of a circuit section and thus a sub-function. The same procedure can be used with superordinate functions for the attribution of the function “Recording-Channel-1” for the recording process at one of the four microphones describes the sub-task of a circuit section and thus a sub-function. The same procedure can be used with superordinate functions for the attribution of the function “Recording-Channel-1” for the recording process at one of the four microphones describes the sub-task of a circuit section and thus a sub-function.

Fig. 3: Proposal semantic encoding of the rotation speaker using EMON and related ontologies

Conclusion

Scores of Electronic Music confront us with new problems as to how to encode specific processes of electronic sound generation. These processes are not only relevant within a limited scope on production processes from a very specific area of music history. Through the vast development of music electronics of the 20th century and within the contexts of digitization of all levels of music production, they rather have become interwoven with a multitude of aspects within present day music theory and practice. Therefore, the benefit of encoding these processes using a semantic data representation is straightforward. Intended functionality and composer’s thoughts on the apparatus can be documented. The actual target of Semantic Web and Linked Data technologies, a worldwide and cross-disciplinary connected knowledge database (Sakr, 2018), allows various use cases for the encoded information. For example, as RDF datasets become readable by AI systems, it can be expected that they will be used beyond the mere encoding of Electronic Music scores. A resulting database on the foundations of this ontology could in a future scenario even provide possibilities for automated remodelling of device setups, for example on a platform like Max/MSP. Sound generation processes that were previously difficult to comprehend could thus be exemplified within the experimental setting itself.

References

Encoding Polyphony from Medieval Manuscripts Notated in Mensural Notation

Karen Desmond  
Brandeis University, USA  
kdesmond@brandeis.edu

Laurent Pugin  
RISM Digital Center, Switzerland  
laurent.pugin@rism-ch.org

Juliette Regimbal  
McGill University, Canada  
juliette.regimbal@mail.mcgill.ca

David Rizo  
Universidad de Alicante, ISEA.CV, Spain  
drizo@dlsi.ua.es

Craig Sapp  
Stanford University, USA  
craigsapp@gmail.com

Martha E. Thomae  
McGill University, Canada  
martha.thomaelias@mail.mcgill.ca

Abstract

This panel submission for the 2021 Music Encoding Conference brings together five short papers that focus on the making of computer-readable encodings of polyphony in the notational style – mensural notation – in which it was originally copied. Mensural notation was used in the medieval West to encode polyphony from the late thirteenth to sixteenth centuries. The Measuring Polyphony (MP) Online Editor, funded by an NEH Digital Humanities Advancement Grant, is a software that enables non-technical users to make Humdrum and MEI encodings of mensural notation, and links these encodings to digital images of the manuscripts in which these compositions were first notated. Topics explored by the authors include: the processes of, and the goals informing, the linking of manuscript images to music encodings; choices and compromises made in the development process of the MP Editor in order to facilitate its rapid deployment; and the implications of capturing dual encodings – a parts-based encoding that reflects the layout of the original source, and a score-based encoding. Having two encodings of the music data is useful for a variety of activities, including performance and analysis, but also within the editorial process, and for sharing data with other applications. The authors present two case studies that document the possibilities and potential in the interchange of music data between the MP Editor and other applications, specifically, MuRET, an optical music recognition (OMR) tool, and Humdrum analysis tools.

Introduction

The availability of high-quality digital images of music manuscripts has fundamentally changed the way we encounter the music of the western Middle Ages. Instead of consulting modern printed editions, those interested can go directly to images of the original manuscript sources. Nonetheless, for both human and computer readers (such as OMR applications), issues with deciphering and understanding the manuscript sources persist – legibility, scribal copying quirks, and outright mistakes. While the nature of the interaction has changed due to the access to these images, the production of human- and computer-readable versions of what medieval scribes encoded in these manuscripts is still useful and often requires (human) editorial interventions.

What maybe is no longer needed, however, is the translation of medieval notations into modern notation, which does a poor job of communicating the nuance and simplicity of the notation in which the music was originally conceived. This panel presentation brings together five short papers that focus on the making of computer-readable encodings of polyphony in a notation used in the medieval West from the late thirteenth century.
to sixteenth centuries called mensural notation. The Measuring Polyphony (MP) Online Editor, funded by an NEH Digital Humanities Advancement Grant (Principal Investigator: Karen Desmond), is a software that enables non-technical users to make Humdrum and MEI encodings of mensural notation. Capitalizing on the availability of high-quality images documented by the International Image Interoperability Framework (IIIF) standards, the encodings are linked directly to ‘zones’ within IIIF-documented manuscripts, allowing the simultaneous viewing and study of both the manuscript image and its encoding, but with the images stored and served by the manuscript’s host institution. The link between image and encoding also facilitates the editing process: In §1, Juliette Regimbal introduces the data model design for the MP Editor, while in §2, Karen Desmond examines some of the early choices made in the development process, focusing on the intersection between the project’s requirements and choices as to what elements get encoded.

One key feature of these encodings is that they encode both the content of the voice parts as they were copied in the original manuscripts, and encode the voice parts realized into score format. Laurent Pugin (in §3) focuses on one ubiquitous feature of mensural notation – the ligature, a glyph that ligates (that is, ties together) several note shapes – and how these ligatures are flexibly handled in the Verovio project (see §3), whether represented in parts-based or score-based encodings. In §4, David Rizo, Martha E. Thomae, and Regimbal examine the possibilities of interchange between computer applications – between MuRET, an optical music recognition (OMR) tool, and the MP Editor – so that the MP Editor’s score-editing features can be used to refine MuRET’s OMR transcriptions and produce digital editions. Finally, in the last section (§5), Craig Sapp, along with Regimbal and Thomae, continues this theme of interchange with a presentation of conversion tools between MEI-mensural and Humdrum **mens digital formats, which means Humdrum analysis tools can be used on MEI-mensural encodings.

1 Introducing the Data Model Design for the Measuring Polyphony Editor (Regimbal)

1.1 Overview of the Data Model and Its Requirements

All software is designed with decisions made about how its data are structured and transmitted across different components and, in many cases, outside of the software. The MP Editor is not an exception and primarily stores musical content, the locations of this content, and metadata. It has a modular design composed of three stages – the Input Editor, the Score Editor, and the Editorial Corrections mode – that structure data differently to better fit the tasks performed in each stage. Many specifics of these tasks only became clear later on, and the data model constructed for the MP Editor could not take them into account before development. Instead, a flexible, ‘prototype-friendly’ model was designed using MEI as a common interchange format to support the development team in implementing the different features of the editor. This model also was required to input and output MEI encodings to provide for interoperability with other tools and projects. An overview of this structure can be seen in Figure 1.1 below.

Each of the three stages uses a different format for storing and manipulating data in response to the scale of its primary editing task. The Input Editor stage either starts with no preexisting data or loads data from a parts-based mensural MEI file (on the structure of parts-based MEI see section 4.2.3 below). Data is formatted in an internal data format that stores data within a system (a single stave) in Humdrum. This permits Humdrum to be used as an intermediary to convert this internal data format, when needed, into a parts-based MEI file for the Score Editor stage. This parts file can be edited, exported, or ‘scored up’ to produce a score-based MEI file. The score-based MEI file is used to store data for the third and final stage of editorial corrections. Even here, the MEI encoding may pass through Humdrum again for additional processing before being exported.

1 For good introductions to mensural notation see [5, pp. 85–134] and [4].
1.2 Stages of the MP Editor

The Input Editor stage uses its own internal format based on Humdrum. Each musical element that can be entered in the MP Editor is stored in a manner directly based on Humdrum’s **mens format [9]. This internal input data format distinguishes itself in how it handles systems, the base unit of the Input Editor. A system is the first element that must be entered, preceding both the assignment of parts and the entry of notes, and defines the structure of the music across different pages. This structure does not necessarily emerge in order. A user can enter systems in any order they find most convenient and set the association of the systems to parts after all other content is entered. This means that the actual structure as it would be expressed as a single Humdrum or MEI document is only fixed once the user finishes this Input Editor pass.

To match the user’s possible actions without risking the data being transformed into an invalid state, links between the system, its part, and its location within the manuscript and within a page are kept dynamic in the input data format. When the user signals the end of the Input Editor stage, the systems are then sorted with the links to other data resolved so they can be encoded in a parts-based MEI file as shown in Figure 1.2.

This parts-based MEI file is then transferred to the Score Editor stage. The user is presented with a view as a score, but data is edited at this stage in the parts format and converted to a score on each change. The Score Editor interface presents a pared down level of control: Changes can be made within systems and parts but the higher level actions that impact the structure of the document are not available. This ensures that maintaining valid data, in this case a valid MEI document, is straightforward. As such, changes are made directly to the parts-based MEI file using standard browser functions to manipulate the Document Object Model (DOM) on the web and XML standards like XPath queries [2]. This minimizes the overhead involved in data conversion and still allows for easily reusable outputs in the form of parts- and score-based MEI documents.

The Editorial Corrections stage keeps the same interface as the Score Editor, however the underlying task changes from fixing problems with entry in the Input Editor stage to fixing problems with how the piece was written on the original page. While this change is important to the user, the scale of the edits remains at the level of elements within systems. The same DOM and XML tools are used to enter edits directly into the score-based MEI document with the original and corrected data wrapped in <choice>, <sic>, and <corr> tags.
1.3 Impact of the Data Model

The data model designed for the MP Editor focuses on matching the scale of user edits through the web interface. While converting between formats adds some overhead and room for error, this choice allows for the rapid implementation of features. Now, this flexibility in the data model makes the decision easy to add other features that require conversion to a different format, rather than recreating the process ourselves. An example of this is discussed in more detail in section 5.

At this time, it appears that this approach to the data model worked well for a project like the MP Editor. Rather than spending our limited time focusing on a workable monolithic data structure, we were able to move quickly and reach more serious design challenges, like those discussed in the next section. Keeping the inputs and outputs of the editor as standard MEI also permitted its use in workflows involving other projects, like MuRET (see section 4).

2 What Gets Encoded: Early Choices, Decisions, Compromises (Desmond)

2.1 Fundamental Requirements of the Measuring Polyphony Editor

One of the fundamental requirements of the MP Editor is that it be simple: an easy-to-use tool for someone with no little to no experience in computer coding that allows them to rapidly enter pitches and rhythms through ASCII keystrokes. The user sees, in real-time, a visual representation of how the note shapes generated by their keystroke entry match the shapes in the medieval manuscript. The end goals are twofold: (1) the creation of encodings of medieval music repertoires that can be read by computers, but also (2) the generation of transcriptions and/or editions in score format that can be read by twenty-first-century musicians and scholars. A crucial by-product of the encoding process is that users of the MP Editor become intimately familiar with the palaeography and layout of medieval manuscripts: with how medieval notations work, and with how medieval scribes efficiently and elegantly encoded their music repertoires.

This requirement of simplicity influenced decisions regarding which aspects of the music must be encoded versus those whose encoding might add unnecessary complexity to the interface. One of our first decisions, which was intended to simplify and ensure the speed of the user input, but which had a significant impact on
both the design of the interface and the encoding process, was that we chose the staff as the fundamental unit for the input interface. A compromise of this decision is that the resulting encodings do not have a one-to-one correspondence between the notational glyph and the manuscript ‘zone’ (as you might have in encodings generated through OMR), but rather the correspondence is between the individual staff (and its contents) and the manuscript ‘zone’.

Another fundamental requirement that had a significant impact on the data model design was that we wanted to create two encodings: one that encodes the musical content as notated in parts in the original manuscript source, and a second encoding that scores up the voice parts. As part of the encoding process, the MP Score Editor allows the transcriber to quickly catch transcription mistakes because they can toggle views to see the alignment of the parts in the midst of transcribing. As described in section 1, the MP Editor was designed to accommodate these iterative processes of input, proofreading, and editing. In the following, I briefly outline some consequences that these early decisions – the staff-based encoding and the encoding of two versions of each composition (a parts-based version and a score-based version) – had on how certain structural aspects of late medieval repertoires are encoded with the MP Editor. Specifically, I outline how medieval encodings of formal structure intersect with manuscript page layout (the mise-en-page), and how a modern score-based layout has to realize musical content based on instructions often only implicitly conveyed in the parts-based layout.\(^5\)

### 2.2 Tenor Repetitions in Motets

In the later medieval period – that is, from about the third quarter of thirteenth century – most polyphonic repertoires were copied in a parts-based layout. This was a change from earlier polyphony: The Aquitanian and Notre Dame organum repertoires, for example, were encoded in score. The fundamental difference between organa and medieval motets is that for the duration of an entire motet, each voice part usually moves at a different rhythmic rate, and often have different texts. The top of the page shown in Figure 2.1, for example, shows a discant section from the organum Alleluia. Nativitas gloriose: The three parts, tenor, duplum, and triplum have roughly the same number of pitches, all moving at roughly the same rhythmic rate, and are copied in score. By contrast, Figure 2.2 shows a motet based on this same passage: Now the two upper voices (the triplum and motetus) have texts underlaid throughout, and the tenor takes up much less space on the page than either of the two upper parts. Later in the thirteenth century, and into the fourteenth, the uppermost voice frequently took up even more space on the page. Practically speaking, in terms of production costs – parchment was the most expensive cost for makers of music manuscripts since the skin of a single animal might make just two or four folios – scribes needed to conserve manuscript space. Copying medieval motets laid out in parts instead of the older score-based layout saved a lot of parchment.

In their desire to save time and parchment, medieval scribes became experts in techniques of abbreviation. Depending on the eventual readers and purpose of the manuscript, the degree of formality of text script and abbreviation differs. In medieval music manuscripts, motet tenors were frequently copied in an abbreviated manner. Most motet tenors were composed from repeated segments of preexistent melodies, taken from plainchants or songs. In Figure 2.3a, the melodic segment is written out once, and four short vertical strokes indicate the tenor sings it four times; a variant on this has dots surrounding the three strokes (Figure 2.3b). In Figure 2.3c, the repeat of the tenor part is indicated by a rubric or canon (literally an instruction to the performer) that is an abbreviation of the Latin word iterum ("it.") which means ‘again’. Sometimes both the sign and the rubric are found together, as shown in Figure 2.3d. Most often the entire tenor is repeated, but sometimes the sign and/or rubric can be found in the middle of the tenor part: What comes before the sign is repeated and what comes after is sung at the very end (this is the case in the tenor of Figure 2.3d).

In development meetings for the MP Editor, we discussed at length how to encode these tenor repetitions, debating the applicability of attributes such as <corrres>, <sameas>, and <copyof>, and the elements <multi-Rpt>, <rptSign>, <dir>, and <expansion>. We wanted our encoding to do two things: (1) in the parts-based file, encode the pitches of the tenor exactly as they appear in the manuscript; (2) in the score-based file, actually repeat the tenor segment the number of times indicated. We also wanted to avoid adding new MEI elements or attributes, even though the above-mentioned elements and attributes were conceived with Common Western Music Notation (CWMN) in mind. But, in addition, ideally we wanted to include, within the parts-based file, an

\(^5\) On the manuscript layout of late medieval polyphony, see [6, 7] and several essays collected in the volume [10].
Figure 2.2: The beginning of the three voice motet *Ex semine rosa/Ex semine habrahe/EX SEMINE*, copied in parts. Staatsbibliothek Bamberg, Msc. Lit. 115, fol. 15v. Photo: Gerald Raab, [http://digital.bib-bvb.de/webclient/DeliveryManager?custom_att_2=-simple_view-er&pid=2957869&childpid=2957903 (CC BY-SA 4.0)].
Figure 2.3: (a), (c), (d) are from Paris, Bibliothèque nationale de France, fr. 146, fols. 9v, 1v, and 3v (details), https://gallica.bnf.fr/ark:/12148/btv1b8454675g; (b) is from Paris, Bibliothèque nationale de France, fr. 1586, fol. 213v (detail), https://gallica.bnf.fr/ark:/12148/btv1b8449043q/f432.item. © gallica.bnf.fr / Bibliothèque nationale de France (license).

Figure 2.4a: MP Editor Interface to mark tenor repetitions.

Figure 2.4b: Detail of Tenor Repetition Entry Screen.

If the piece has repetitions of the tenor that are not written out in full, please specify here. See "Music Input Help" for details.
encoding of the actual symbols or text that the scribe used to indicate the \textit{tenor} repetition. In the end, we opted to encode \textit{tenor} repetitions using the \texttt{<dir>} element in the parts-based file, with the number of repetitions of the \textit{tenor} indicated by \texttt{@n}, and the segment marked by including the \texttt{@xml:id}s of the start and end of the segment in the \texttt{@plist} and \texttt{@follows} attributes (Listing 2.1). In the score-based file, the pitches generated following the scribe's instructions are encoded by including a \texttt{@copyof} attribute with the \texttt{@xml:id} value of the pitch it repeats. The interface allows a user to mark the portions of the \textit{tenor} that are repeated, and to indicate the number of statements of this melodic segment (Figures 2.4a and 2.4b). Future discussions with the MEI Mensural Interest Group of the Music Encoding Initiative\footnote{https://music-encoding.org/community/interest-groups.html (accessed January 12, 2022).} will look at the options for encoding the symbols and rubrics that indicate the repetitions in the medieval manuscript.

\begin{verbatim}
<part>
  <scoreDef>
    <staffGrp>
      <staffDef n="1" lines="5" notatiotype="mensural.black"
               notationsubtype="Ars_antiqua" label="tenor" xml:id="tenor">
        <mensur modusminor="3" tempus="3"/>
      </staffDef>
    </staffGrp>
  </scoreDef>
  <section>
    <staff n="1">
      <layer n="1">
        <pb facs="#m-7211a079-a7e2-4016-b3b9-74c5e4ce03bd"
              xml:id="m-53315e6e-c978-42f0-9133-915ac68d3378"/>
        <sb facs="#mc47c8bc2-ee92-4c2c-93a1-6762131aa821"
            xml:id="m-8f6c12cb-e4da-4382-86bb-57779ddblace"/>
        <clef xml:id="m-f89d3444-4758-42e8-af7f-c8970c60b32b" shape="C" line="3"/>
        <note xml:id="m-3a009f48-f43d-43f6-85ab-48383077d93f" dur="longa" oct="3"
              pname="f"/>
        <note xml:id="m-47c737f7-5848-4a20-a045-96ca90957426" dur="brevis" oct="3"
              pname="g"/>
        <note xml:id="m-64042039-24d3-4265-8427-77247745af22" dur="longa" oct="3"
              pname="a"/>
        <note xml:id="m-029clae5-9374-46d9-9e6b-52965d345b65" dur="longa" oct="3"
              pname="g"/>
        <note xml:id="m-c472a4cc-0c41-47ec-8e75-34807cf4652b" dur="longa" oct="3"
              pname="f"/>
        <rest xml:id="m-5e364b62-11ae-402b-aa39-ceccfc510ab3f" dur="brevis"/>
      </layer>
      <!-- The @n attribute on <dir> is used to represent the number of repetitions in a machine-readable format -->
      <dir n="3" layer="1" plist="#m-3a009f48-f43d-43f6-85ab-48383077d93f
                   #m-5e364b62-11ae-402b-aa39-ceccfc510ab3f"
           follows="#m-5e364b62-11ae-402b-aa39-ceccfc510ab3f"/>
    </staff>
  </section>
</part>
\end{verbatim}

\textbf{Listing 2.1:} Code from the parts-based file that uses \texttt{<dir>} to mark \textit{tenor} repetitions.
2.3 First- and Second-time Endings in Song Forms

Another standard abbreviation in manuscripts of medieval polyphony is in the notation of certain song forms: specifically those that have repeated sections, but with different first- and second-time endings. Figure 2.5 shows one manuscript source of Machaut’s ballade *Esperance qui* where the first statement of the ‘A’ section has an open ending, and the second statement has a closed ending. Generally, medieval scribes included no indication in the manuscripts (no symbols or rubrics) that specified which pitches comprised the open and closed endings of the same section: Scribes and performers were aware of this copying/performance convention. In this case, the encoding of the parts-based and score-based files does not have to differ, since modern musicians are also usually aware of the conventions of first- and second-time endings. Currently, encoders using the MP Editor to encode medieval songs simply encode exactly what they see on the page, with the pitches for the second-time ending following directly on from the pitches of the first-time ending (see Figure 2.6). In the future, we plan to add functionality to the interface whereby the encoder could select which notes comprise the first- and second-time endings, so that these endings would be marked-up in the score-based encoding. Again, the manner of encoding will be discussed within the Mensural Interest Group of the Music Encoding Initiative.

![Figure 2.5: Boxes and arrows show the first- and second-time endings in Machaut's ballade *Esperance qui*, copied in Paris, Bibliothèque nationale de France, fr. 1584, fol. 461r (detail), https://gallica.bnf.fr/ark:/12148/btv1b84490444/f943. © gallica.bnf.fr / Bibliothèque nationale de France (license).](image)
Another fundamental way that medieval scribes could save space on the page was to use ligatures to notate pitches that were sung to a single syllable of text, although in this case this copying convention is an inheritance of neume-based plainchant notations. The next section focuses on the problems of encoding and representing pitches notated with ligatures in mensural notation.

3 Bridging CWMN and Mensural Notations: Implementing the Encoding of Ligatures (Pugin)

Mensural notation available in separated parts includes several features that do not fit well with the concept of a score where all voices are vertically aligned and represented together. Nonetheless, the score representation provides the reader with a view of the music that remains essential in many situations. This is particularly true when editing mensural notation music from original sources. The score makes it much more straightforward to find the places in the transcription that need to be corrected or amended than with each voice left transcribed separately. Amongst the features of mensural notation that are problematic in score representation are ligatures. Ligatures can be represented as separated individual notes in a score, but it is often desirable to preserve in the encoding the original aspect of the ligatures in the source.

As part of the development of the MP Editor, some improvements have been made to the Verovio engraving library. Verovio is a music notation engraving library for rendering MEI scores into various digital environments. Its main focus is Common Western Music Notation (CWMN) but it also supports some features of mensural notation. One of the key features of Verovio for mensural notation is its handling of triple and duple divisions. It allows for the voices to be properly aligned without having to encode in each note whether the}

---

7 In addition, in mensural notation, the form of the ligature had specific meaning for the rhythmic duration of each note contained within it; in modal notation, patterns of ligatures (how many notes contained within each ligature and in what order) indicated specific rhythmic patterns.

time division is triple or duple since Verovio infers this itself from the encoding of the mensuration sign(s). This proved to be very useful for the MP Editor. On the other hand, ligatures were lacking a robust implementation, and we describe in this section what had to be considered for supporting them in Verovio and the current status of the implementation. We look in particular at what the implementation of ligatures meant in terms of the underlying encoding and in terms of usability of the MP Editor.

3.1 Automatic Layout

One of the main design principles at the root of the development of the Verovio engraving library is automatic layout. In addition to producing excellent quality engraving results, Verovio aims to do so without extensive engraving instructions indicating how a music notation element needs to look or how it needs to be laid out – i.e., how it needs to be arranged on the page. An important point behind this design principle is that Verovio is meant to be usable in dynamic environments featuring interactivity. The MP Editor is a typical use case where user interactions require dynamic rendering. In such a context, reducing the encoding content to the minimal information needed provides a maximal level of interactivity and flexibility.

What does this mean for ligatures in mensural notation? What is the minimal information needed for the rendering of ligatures? Let us consider the ligature in Figure 3.1 with a possible corresponding MEI encoding. In addition to the ligature and the notes elements with their duration and pitch information, the encoding also includes some attributes indicating the placement of the stem on the ligature, for example, that there is a stem on the left side of the first note, and that its direction is downwards. The encoding in Figure 3.1 also includes an attribute on the ligature indicating that the ligature is drawn as an oblique ligature (i.e., it has an angled shape). This is obviously all correct and appropriate. Such an encoding also has the advantage that it will make its rendering quite straightforward. From a notational point of view, however, the information is not reduced to the minimum. The fact that the ligature features a stem down on the first note and that it is oblique is in fact already given by the duration of the notes (two breves) and the direction of the interval their pitches yield (downwards). It means that the information needed for this ligature can be reduced. Figure 3.2 shows the MEI encoding with the minimal information Verovio expects for rendering this ligature.

What are the implications of this in the context of the MP Editor – or any other interactive applications? Let us imagine that the pitch of the ligature is changed so that the interval is now a third upwards as illustrated in Figure 3.3. Since attributes for the stem and about the form of the ligature were not included means that the rendering can be updated directly without having to change or remove that extraneous information. In this particular case, it means rendering the ligature as a recta form (proper form, i.e., not oblique) and without a
stem. Similarly, if the duration of both notes is changed from *brevis* to *semibrevis*, the automatic layout implementation of Verovio will add the upward stem to the ligature.

![Image](image1.png)

**Figure 3.3:** A ligature of two *breves* yielding an ascending interval.

There are cases, however, where the form of the ligature must be encoded when a particular shape is desired. This is the case for the *cum opposita proprietate* that can be *recta* or *obliqua*. In this case, adding an attribute to indicate the form of the ligature cannot be avoided (Figure 3.4). Similarly, for *obliqua* pairs within a ligature of more than two notes, an additional attribute must be added to indicate this.

![Image](image2.png)

**Figure 3.4:** *Cum opposita proprietate* is one case where the ligature form has to be explicitly encoded for distinguishing *recta* and *obliqua*.

Another layout feature that Verovio implements for ligatures is the stacking of the *longa* at the end of a ligature. Here, a difference is to be made between black and white mensural notation, and Verovio looks at the notation type for distinguishing the two cases. With black mensural notation, the last *longa* is stacked above the previous note when the interval is a third upwards or more (Figure 3.5).

![Image](image3.png)

**Figure 3.5:** Verovio distinguishes black and white mensural notation and stacks up the *longa* at the end of the ligature when appropriate in black mensural notation.
3.2 Ligatures in a Score

When scoring up parts and aligning the voices, ligatures are problematic because they cannot reasonably be stretched to fit the score layout. One possible solution is to preserve them and to align their global duration as illustrated in Figure 3.6. This is the default behavior of Verovio when ligatures are encountered in scored-up parts.

![Figure 3.6](image)

Figure 3.6: When scored up, the context around ligatures can be aligned, but not the ligatures themselves.

However, this is clearly not satisfactory because the reading of the intervals remains difficult. In order to improve it, we introduced a new option in Verovio that allows displaying ligatures with a bracket above the staff with the notes separated and aligned in the score (Figure 3.7). While the rendering with a bracket is not new and simply adopts a standard editorial practice, the originality of the approach here is that the underlying encoding of the ligature remains unchanged.

![Figure 3.7](image)

Figure 3.7: Verovio has an option to render ligatures in score as separated and aligned notes together with a bracket above the staff without having to modify the underlying encoding.

3.3 Open Questions

With the improvements made to Verovio for the MP Editor, ligatures can now be properly rendered with a minimal encoding, both in parts and in score. However, some open questions remain. One of them is what to do when the encoding is contradictory to the expected rendering rules. For example, what should happen if a ligature is marked as *obliqua* when the rendering rule would expect it to be *recta*? Is this something that can be improved in the validation of MEI through additional Schematron rules? If that is possible, do we need a way to bypass them whenever necessary, for example, where there is a mistake in a source?

Another open question is how best to deal with the problems arising when ‘moving’ towards a more CWMN-like score. For example, an editorial next step can be the insertion of some additional bar lines. Some of them can potentially occur within a ligature. As it stands, this can be made possible and properly rendered with Verovio as long as no system break is expected to occur within the ligature. In other words, there is currently no way to split a ligature over two systems. Such limitations, which are not unique to ligatures, can be problematic with dynamic and interactive rendering and will need to be addressed in the future.
4 From OMR to the Measuring Polyphony Editor: Possibilities of Interchange (Thomae, Rizo, and Regimbal)

Any encoding format has two aims: the preservation of the contents described by it and the possibility of interchange between computer applications. In this section, we focus on the latter. We describe the decisions made in order to export contents from the OMR research tool MuRET to the MP Editor. The MP Editor is used to curate MuRET's transcriptions by both correctly scoring up the different parts and correcting scribal errors to make digital editions. The goal of interconnecting these tools is to provide a semi-automatic way to move from digital images of mensural music to edited scores in symbolic notation.

Optical Music Recognition (OMR) applications perform automatic recognition of the music symbols on a page and usually provide an interface to correct the symbols misidentified by the automatic recognition process. The results of this process can be encoded in a format such as MEI. In mensural notation, however, correctly identifying the symbols present on the page does not convey all the rhythmic information of a piece. In triple meter, the duration implied by a particular note shape can differ in particular contexts. In addition to OMR's recognition of the note shapes and pitches, two additional encoding steps are required to make digital editions of works notated in mensural notation: (1) encoding the durations of the note shapes (perfecta, imperfecta, altera) based upon these contextual rules; and (2) correcting scribal errors that affect the alignment of the voices when the encodings in the parts-based files are scored up.

Rather than implementing these two extra steps in MuRET's OMR workflow (which are not generalizable to other notation systems), we use an existing technology that already handles these steps within its design, namely the Score Editor interface of the MP Editor. Here, we first present the MuRET OMR framework, and then focus on documenting the encoding decisions made on both sides (MuRET's and the MP Editor's) regarding the MEI file to be used as an interchange format between the two applications. We close with some final remarks about future work and issues that are out of the scope of this interchange process.

4.1 MuRET: Music Recognition Encoding and Transcription

The online version of MuRET (Music Recognition Encoding and Transcription) is a research-oriented OMR framework. It includes machine learning models to recognize the music symbols on each staff, providing two types of information per symbol: the class of symbol and its position in the staff. The encoding of these two graphical characteristics of a symbol is known as agnostic encoding. The user can add, edit, and remove misidentified symbols within MuRET's interface (see Figure 4.1). These graphical symbols are converted from an agnostic encoding into a semantic encoding by interpreting information such as pitch and key signatures from the position of the symbols in the staff. The encoding of the semantics of mensural symbols is provided by **mens (see Figure 4.2), which is the Humdrum format for mensural notation.

![Figure 4.1: Recognition and editing of agnostic symbols in MuRET. The tool incorporates a toolbox for correcting mistakes or even adding them from scratch.](https://muret.dlsi.ua.es/muret/) (accessed January 12, 2022).
4.2 Encoding Decisions for the MEI File

We used MEI as the interchange format between MuRET and the MP Editor. In this section, we present the encoding decisions made regarding the MEI file exported by MuRET so that it worked as input for the MP Editor. The decisions are divided into the following four categories:

4.2.1 Use of IIIF Manifest and Server to Share the Transcribed Document Between Applications

Following the philosophy of the International Image Interoperability Framework (IIIF) [11], rather than implementing ad-hoc image servers for each tool and developing a system for interchanging image between each application, the IIIF approach has been used for storing and serving images that can be accessed not only from MuRET and the MP Editor, but also from any other system that could need them in the future.

When importing new musical works to be transcribed with MuRET, images are uploaded into a IIIF-compliant system named Cantaloupe.10 Currently, MuRET just accesses the images using hardcoded IIIF URIs. When exporting the MEI file obtained from the recognition process, MuRET generates and stores a IIIF manifest whose URI is encoded in the MEI header, inside the <source> child of the <fileDesc> element identified as IIIF by using the @targettype="IIIF" attribute (see Listing 4.1).

Listing 4.1: Reference within the MEI header to the IIIF manifest.

```xml
<m ei>
  <meiHead>
    <fileDesc>
      ...
      <sourceDesc>
        <source target="https://<IIIF-server>/<work-name>/manifest.json"
           targettype="IIIF"/>
    </fileDesc>
  </meiHead>
</mei>
```

4.2.2 Facsimile Encoding Decisions

In order to be able to recognize music in a digitized image, the OMR process in MuRET first divides the image into regions of interest such as pages, staves, notes, clefs, etc., that are delimited by bounding boxes. This graphical information is encoded by using the <facsimile> element of MEI (see Listing 4.2) and includes the URL of the source image and the bounding boxes of those regions of interest.

Source images are linked using a IIIF URI in the @target attribute of the <graphic> child of the <surface> facsimile element. All bounding boxes of regions of interest that have been identified in the image are specified through <zone> tags with their corresponding coordinate attributes (@ulx, @uly, @lrx, @lry).

kinds of zones are used with the `@type` attribute: "page", "region", or "symbol". The former is used to delimit the pages shown in the image (usually just one or two of them). The type "region" is used for all other zones, different from individual symbols, that are now differentiated by using the `@label` attribute with values "staff", "title", "drawing", "author", etc. Finally, zones identified containing symbols such as clefs, mensuration signs, accidentals, dots, or notes are also delimited by bounding boxes with the type "symbol" and a label containing the agnostic string representation of the symbol (e.g., "clef.C:L4").

Some OMR models are not able to recognize the exact symbol bounding box, but an approximate one. In that case, the bounding boxes encode the rectangle defined from the identified horizontal position of the symbol until the next one.

```xml
<music>
  <facsimile>
    <surface xml:id="image_2728" ulx="0" uly="0" lrx="1335" lry="2000">
      <graphic target="<IIIF-server>/<work-name>/canvas/f1" xml:id="graphic_21083"/>
      <zone xml:id="page_2964" ulx="47" uly="24" lrx="1128" lry="1682" type="page" label="Page #1"/>
      <zone xml:id="region_25073" ulx="240" uly="123" lrx="1105" lry="267" type="region" label="staff"/>
      <zone xml:id="symbol_116997" ulx="245" uly="173" lrx="282" lry="264" type="symbol" label="clef.C:L4"/>
    </surface>
  </facsimile>
</music>

Listing 4.2: The MEI facsimile element as generated from MuRET.

All musical contents include references to the graphical information in the `<facsimile>` by using the `@facs` attribute. Instead of using the specified bounding boxes, the MP Editor uses the page beginning (<pb>) and system beginning (<sb>) references for graphically delimiting the start of each page and system (see Listing 4.3). More details about the use of <pb> and <sb> are given below.

```xml
  ...  
  <section>
    <staff>
      <layer>
        <pb xml:id="spb_21356" facs="#page_2964"/>
        <sb xml:id="spb_21357" facs="#region_25073"/>
        <clef line="4" shape="C" facs="#symbol_116997" xml:id="clef_21358"/>
      </layer>
    </staff>
  </section>

Listing 4.3: Graphical content linked from musical MEI elements describing musical content.

4.2.3 Structure of the Mensural MEI Parts File

The MEI file used for interchange between MuRET and the MP Editor is a parts-based MEI file. This is an MEI file that encodes the musical content using the `<parts>` element rather than the `<score>` element. This `<parts>` element is intended to encode each of the parts (i.e., voices) individually within a `<part>` element. The parts-based representation reflects exactly what is on the manuscript – most mensural sources are notated in separate parts. This representation is ideal as output for MuRET since, at this point in the workflow, we only have a sequence of music symbols for each part without any note duration information that would allow for lining up the piece as a score – the duration information will be determined by the MP Editor.

The decisions regarding the encoding of the `<part>` elements within `<parts>` may be summarized as follows (note that each `<part>` follows the same structure):

- Each `<part>` has one `<scoreDef>` element containing a single `<staffDef n="1"/>` (see purple area in Listing 4.4).
- Each `<part>` has one `<section>` element containing a single `<staff n="1"/>` (see yellow area in Listing 4.4).
The decisions made regarding the placement of certain elements within a `<part>` are the following:

- **Page-beginning elements** (`<pb>`): The `<pb>` elements do not mark a physical page beginning; instead, they are placed in each voice (i.e., part) to mark the page beginning for that voice. As an example, the encoding below would work for both voices shown in Figure 4.3 (regardless of which voice is placed at the physical beginning of the page). The `<pb>` elements are placed at the very beginning of each voice (as the first child of `<layer>`) and at every page turn. The ‘absolute’ page beginning can be found using encoded facsimile information for each part.

- **Clef** (`<clef>`) and **key signature** (`<keySig>`) elements and their relationship with the system beginning element (`<sb>`): In the original sources, clefs and key signatures are written at the beginning of each system of a voice. Because of this, we decided to encode the `<clef>` and `<keySig>` information following each `<sb>` element within `<layer>` (see Listing in Figure 4.4). The `<clef>` and `<keySig>` elements point to the physical location in the corresponding system in the facsimile through the `@facs` attribute (see blue zones in Figure 4.4).

- **Mensuration element** (`<mensur>`): Contrary to the behavior of clefs and key signatures, mensuration signs are given only at the beginning of the first system of a voice. Because of this, we decided to encode the `<mensur>` element once within the `<staffDef>` element. This `<mensur>` element also points to its location at the beginning of the first system of the piece through its `@facs` attribute (see magenta zone in Figure 4.4). In the case of a change in mensuration, the new `<mensur>` element representing the new mensuration sign is encoded within the stream of notes inside `<layer>`.

To illustrate the encoding of clefs (and key signatures) vs. mensuration signs, see the MEI excerpt in Figure 4.4 used to encode the first voice in Figure 4.3.
Figure 4.3: Manuscript page with two voices shown in red rectangles (left). Page-beginning element (<pb>) marking the “page beginning” of a voice (right).

<part>
  <scoreDef/>
  <section>
    <staff n="1">
      <layer n="1">
        <pb/>
        <sb/>
        <clef/>
        <!-- notes in the 1st system -->
        <sb/>
        <clef/>
        <!-- notes in the 2nd system -->
      </layer>
    </staff>
  </section>
</part>

Figure 4.4: First voice of Figure 4.3, showing the zones for the mensuration sign (magenta box) and the clefs (blue boxes) (left). These zones correspond to the one <mensur> element and the multiple <clef> elements (right). <mensur> is encoded once in <staffDef> and related to the zone where it appears. On the other hand, <clef> is encoded several times, one after each <sb> (within <layer>), and each clef is related to its zone at the beginning of the system where it appears.

<part>
  <scoreDef/>
  <staffGrp>
    <staffDef n="1">
      <mensur/>
    </staffDef>
  </staffGrp>
  <section>
    <staff n="1">
      <layer n="1">
        <pb/>
        <sb/>
        <clef/>
        <!-- notes in the 1st system of the voice -->
        <sb/>
        <clef/>
        <!-- notes in the 2nd system of the voice -->
      </layer>
    </staff>
  </section>
</part>
4.2.4 Encoding of Unqualified Durations and Dots

None of the notes in the parts-based MEI file exported by MuRET provide a @dur.quality attribute indicating the relative duration of a note. The value of this attribute ("perfecta", "imperfecta", or "altera") is meant to be found and encoded within the <note> elements by the MP Editor's scoring-up functionality. In a similar way, all <dot> elements coming out from MuRET do not include a @form attribute clarifying whether the dot is behaving as a dot of division (@form="div") or a dot of augmentation (@form="aug"). The nature of the dots, just as the quality of the notes, is computed and encoded by the MP Editor.

4.3 Future Work and Remaining Issues

In the transfer from MuRET to the MP Editor, a few details get lost: the presence of barlines (<barLine>), the stem direction of notes (@stem.dir), the location of the rests (@Loc), and the bounding boxes (the <zone> elements) of individual music symbols. The first three can be easily fixed since barlines, stem directions, and rest positions are supported in **mens, which is the internal representation used in both MuRET and the MP Editor (as pointed out in sections 4.1 and 1.1, respectively). The only change needed is to add functionality in the MP Editor to parse these attributes and/or elements coming from the uploaded MEI file.

On the other hand, encoding the zones of each symbol is not within the scope of the MP Editor. In the Input Editor interface of the MP Editor, the user draws bounding boxes for each system and enters (types in) the symbols – clefs and notes – belonging to that system. As such, entry of symbol-level workflow is unsupported in the current interface. Despite this, the <zone> elements for each symbol in MuRET's MEI file can easily be recovered by running a post-processing script on the output of the MP Editor.

Another minor issue is that a few elements from the uploaded parts-based MEI file change the @xml:id values in the MP Editor. This has no effect, however, on the quality of the edited score exported by the MP Editor. Part of future work could devise a way to avoid any loss or change in the information provided in the uploaded parts-based MEI file. One proposal would be to directly access the MP Score Editor and avoid passing through the Input Editor part that internally encodes the uploaded information into Humdrum (see Figure 1.1).

Here we have presented the decisions that allow for MuRET and the MP Editor to communicate with each other in order to generate edited scores with OMR assistance. Each tool is responsible for a different task; MuRET deals with symbol recognition and the MP Editor with automatic voice alignment and editorial corrections. Their integration allows us to move from digital images of mensural music to symbolic edited scores in a semi-automatic way. Future work would entail improving the information interchange between MuRET and the MP Editor to support more varied workflows.

5 Interaction between Measuring Polyphony MEI Scores and Humdrum Analysis Tools (Thomae, Sapp, and Regimbal)

The MP Editor is used to generate edited scores encoded in mensural MEI. The Score Editor interface has an editorial mode for correcting scribal errors. The MP Editor facilitates the editorial process through (1) rendering the piece as a score; and (2) allowing the user to bar the transcription by different mensural note values (semibreves, breves, longs). Computational music analysis tools can also assist in the editorial process by providing information about counterpoint. We present the interaction of Humdrum analysis tools with the MP Score Editor to facilitate the detection of voice-alignment errors. The first section introduces the analysis tool used and the goal behind its inclusion into the MP Editor, while the second section focuses on the implementation details.

5.1 Dissonance Analysis in the MP Editor

The score layout of the voices in the MP Score Editor interface can reveal counterpoint errors that are useful in detecting mistakes in voice alignment. These mistakes might be due to scribal errors, user-entry errors, or

---

11 Some of the elements that do change their xml ids are <dot>, <accid>, and <ligature>. These elements represent features that are encoded in **mens within the same token as the note they relate to (e.g., Sg for a note followed by a dot, Lr# for a note with an accidental, and [sg and sa] for the notes at the beginning and ending of a ligature). Other elements that change xml ids are the children of facsimile (<surface>, <graphic>, and <zone>), the page and system beginning elements (<pb> and <sb>), and the lyric-related elements (<verse> and <syl>).
errors in the scoring-up algorithm of the MP Editor. Detecting voice-alignment errors is useful in the process of producing an edited score – where all notes and their interpreted durations are accurate and scribal errors have been corrected.

We have implemented functionality in the Score Editor interface to facilitate the capture of errors in voice alignment by using counterpoint cues, specifically, dissonances. This functionality shows the dissonances in the score (see Figure 5.1), where the definition of ‘dissonance’ follows the generally understood conventions of Renaissance counterpoint. The dissonance labels used come from the dissonant function labels of the Verovio Humdrum Viewer’s (VHV) “dissonant filter” (note that while the fundamentals of contrapuntal writing emerged

Figure 5.1: MP Score Editor interface with the “Add Dissonance Labels” checkbox activated. The dissonance labels are shown below each voice in a dark orange color. The description of these labels can be found at https://doc.verovio.humdrum.org/filter/dissonant/#dissonant-function-labels (accessed January 12, 2022).

Figure 5.2: The Z label in the highlighted note in the superius points to a voice-alignment error. The superius forms a second with the bass (and the altus) in a strong beat, which is an uncommon dissonance for Renaissance style. The voice-alignment error, revealed by this counterpoint error, was due to a user-entry error.
in the fourteenth century, many of these labels and contrapuntal concepts will not apply to music before the fifteenth century).  

The Z/z labels indicate an “unclassified dissonance” according to Renaissance style. Since this type of dissonances rarely occurs in Renaissance music, their presence in a piece might indicate an error in voice alignment, as illustrated in Figure 5.2. The appendix provides a pair of graphs, generated from data in the Josquin Research Project and Tasso in Music Project, showing the frequency distribution of the various dissonance labels during two different periods in the Renaissance.

The VHV “Renaissance dissonance labels” filter only works on **kern files, which is Humdrum format for Common Western Music Notation (CWMN). Since the output of the MP Score Editor is mensural MEI, we converted these MEI scores into **kern through a series of processes in order to have access to the “Renaissance dissonance labels” filter. The following section presents the details about this conversion process and the insertion of the dissonance labels into the MEI file for rendering them within the MP Score Editor.

5.2 Implementation Details

The mensural MEI score of the MP Editor is converted into **mens (Humdrum format for mensural notation), and then into **kern (Humdrum for CWMN). The @xml:id of the notes are preserved in this conversion process. Once in **kern, the dissonant filter is used and both the dissonance labels and @xml:id are retrieved and encoded in a JSON file. These JSON-encoded labels can then be applied back to create a new, annotated MEI file. The saved @xml:id data are used to find the correct notes in the original MEI file using XPath. A new verse is inserted for these notes in a distinct color to display the dissonance labels in a clear way to the user. The process can be seen in Figure 5.3.

5.2.1 Conversion Process

![Conversion process](image)

**Figure 5.3:** Conversion process starting from the MP Editor’s Mensural MEI score to retrieve the dissonance labels to render in the MP Editor.

Appendix

![Dissonant types in JRP scores](image)

**Figure 5.4:** Dissonance label frequencies in Josquin Research Project scores (1480–1520). The Z label has a frequency of 1.08%, and the z label has a frequency of 0.85%.

---

Figure 5.5: Dissonance label frequencies in Tasso in Music Project scores (primarily 1580–1600). The Z label has a frequency of 2.15%, and the z label has a frequency of 4.33%. The high increase in the use of perfect fourths above the bass (z) might indicate that the interval was considered less dissonant in this later period (compare with Figure 5.4).

Acknowledgments

The authors gratefully acknowledge The National Endowment for the Humanities, Fonds de recherche du Québec – Société et culture (FRQSC), Bourse au doctorat en recherche (13D - Musique) 2019-B2Z-261749, Alex Morgan, postdoc for the Josquin Research Project (2017), for his work on the “Renaissance dissonance labels” filter in collaboration with Craig Sapp, the Spanish Ministry HISPAMUS project TIN2017-86576-R, and the MultiScore Project, I+D+i PID2020-118447RA-I00, funded by MCIN/AEI/10.13039/50110001103.

Works Cited

Automatic for the People: Archives and the Future

Pip Willcox
The National Archives, UK
Pip.Willcox@nationalarchives.gov.uk

Abstract

Artificial intelligence will take half our jobs in the next 20 years; people will be left helpless and subjugated to the will of machines. (We’re all doomed.)

Artificial intelligence is a distant dream; our current machine learning methods are overblown hype destined to fail in their promise to improve lives. (We’re all doomed.)

Artificial intelligence has already taken over, and our historical biases are baked into it. (We’re all doomed.)

Between these extreme positions – exaggerated only slightly – that scream at us from headlines and sidebars, is a world of nuance: of ethical, legal, technological, commercial, environmental, cultural, infrastructural, and practical considerations.

For centuries archives have collected, organised, preserved, and enabled access to materials produced by other activities – administrative, social, cultural, musical, legal, and commercial. What this work has entailed has been transformed by our embrace of digital technologies, and we have seen the term ‘archive’ borrowed and repurposed by our friends in computer science. Artificial intelligence is one area where the term’s two usages approach each other, and where the importance of archives and their function of holding people and systems to account is evident.

In the practice of archiving, there may always have been tension between attention to detail, and descriptions of material ‘good enough’ to enable their discoverability and re-use for consultation and research. We are seeing this tension through a new lens and at a new scale as significant born-digital archives join digitised, analogue archives in the collections we curate. At The National Archives (UK)1 we are experimenting with computational approaches to help us in our mission to collect and preserve the record, to connect people with their history, and to support the work of archives.

The National Archives is an essential resource for our democracy, a public good, and an asset for future generations, as we lay out in our vision Archives for Everyone.2 To fulfil this role, we are committed to building trust, and tearing down barriers to access, participation, and understanding. It is frequently noted that while digitising analogue collections enables new knowledge to be created and linked, it creates a further collection that equally requires preservation and curation. As we aim to increase understanding and remain trusted, digitisation also demands contextualisation and explanation, linking across and beyond individual collections to understand, for example, voices and perspectives that are not recorded, reflected and valued. As we look to computational methods to assist us in this, we consider the algorithms we use, their training data, the software we use to understand and create new links, the links themselves, the wider hyperstructure and the uses it enables, the content behind those links, the tools we use to explain the approaches, the people who use our resources, and the standards that underlie all of them.

Drawing on collaborative research at The National Archives, including through the UK Arts and Humanities Research Council’s programme Towards a National Collection, this talk explores computational archival science, artificial intelligence, citizen involvement, and post-custodial approaches to challenge doom-laden technological determinism, and how together we might combine ‘hand-curated’ and ‘at-scale’ approaches to our shared cultural heritage to ensure automation works ‘for the people’.


© Published in 2022 with Humanities Commons. Crown copyright. Licensed under the Open Government Licence v. 3.0 (https://www.nationalarchives.gov.uk/doc/open-government-licence/version/3/).
<table>
<thead>
<tr>
<th>Wednesday, 21 July 2021</th>
<th>Thursday, 22 July 2021 (Un-conference)*</th>
</tr>
</thead>
<tbody>
<tr>
<td>15:00-15:10 Music event El Mister d’Ex, talk by Rubén Pacheco</td>
<td>14:00-15:00 Board meeting</td>
</tr>
<tr>
<td>15:30-15:45 Panel</td>
<td>15:00-16:00 Community meeting (open to all)</td>
</tr>
<tr>
<td>15:45-16:00 Panel</td>
<td>16:00-16:30 Meet the MEC2021 design team</td>
</tr>
<tr>
<td>15:45-16:00 Panel</td>
<td>16:30-17:30 Interest groups (IG) Metadata IG</td>
</tr>
<tr>
<td>16:00-16:45 Keynote #4</td>
<td>17:30-17:40 Short Break</td>
</tr>
<tr>
<td>16:00-16:45 Keynote #4</td>
<td>17:40-18:40 Interest groups (IG) and other meetings</td>
</tr>
<tr>
<td>17:15-18:45 Paper presentations #2</td>
<td>18:00-18:40 Short Break</td>
</tr>
<tr>
<td>18:45-19:15 Break: discussion of papers</td>
<td>18:40-18:50 Short Break</td>
</tr>
<tr>
<td>19:15-19:30 Short Break</td>
<td>19:30-20:45 Keynote #2</td>
</tr>
<tr>
<td>20:45:21:00 Closing</td>
<td>19:30-20:15 Poster slam</td>
</tr>
<tr>
<td>15:00:15:40 Online platforms and on-site venue open</td>
<td>21:00:21:00 Meetings</td>
</tr>
<tr>
<td>15:40-16:00 Opening</td>
<td>21:30 Dinner</td>
</tr>
<tr>
<td>16:00-16:15 Keynote #1 Ávaro Torrente and Ana Lorenz</td>
<td>18:00-18:30 Break</td>
</tr>
<tr>
<td>16:00-17:15 Break</td>
<td>18:30-21:00 Workshop Session #1 ODD Tutorials</td>
</tr>
<tr>
<td>16:30-16:45 Break: discussion of papers</td>
<td>19:15-19:45 Break: discussion of papers</td>
</tr>
<tr>
<td>17:15-17:45 Break</td>
<td>19:45-20:00 Break</td>
</tr>
<tr>
<td>17:45-19:15 Paper presentations #1</td>
<td>20:00-20:15 Poster slam</td>
</tr>
<tr>
<td>19:15-19:45 Break: discussion of papers</td>
<td>20:15-21:00 Social meetings</td>
</tr>
<tr>
<td>19:30-20:45 Keynote #2</td>
<td>21:00:21:00 Meetings</td>
</tr>
</tbody>
</table>

*Un-conference*