DHSI

DIGITAL HUMANITIES SUMMER INSTITUTE

Introduction to Digital Approaches to Music Research

Timothy Duguid

ETCL
ELECTRONIC TEXTUAL CULTURES LAB

Social Sciences and Humanities Research Council of Canada

Conseil de recherches en sciences humaines du Canada

Welcome to DHSI 2023!

Thank you for joining the DHSI community!

In this coursepack, you will find essential workshop materials prefaced by some useful general information about DHSI 2023.

Given our community's focus on things computational, it will be a surprise to no one that we might expect additional information and materials online for some of the workshops—which will be made available to you where applicable—or that the most current version of all DHSI-related information may be found on our website at dhsi.org. Do check in there first if you need any information that's not in this coursepack.

Please also note that materials in DHSI's online workshop folders could be updated at any point. We recommend checking back on any DHSI online workshop folder(s) that have been shared with you in case additional materials are added as DHSI approaches and takes place.

And please don't hesitate to be in touch with us at **institut@uvic.ca** or via Twitter at **@AlyssaA_DHSI** or **@DHInstitute** if we can be of any help.

We hope you enjoy your time with us!

Statement of Ethics & Inclusion

Please review the DHSI Statement of Ethics & Inclusion available here: **https://dhsi.org/statement-of-ethics-inclusion/**

DHSI is dedicated to offering a safe, respectful, friendly, and collegial environment for the benefit of everyone who attends and for the advancement of the interests that bring us together. There is no place at DHSI for harassment or intimidation of any kind.

By registering for DHSI, you have agreed to comply with these commitments.

Virtual Sessions

Your registration in DHSI 2023 also includes access to the virtual **institute lecture** sessions. Access details for these talks will be shared as DHSI approaches.

Due to the high volume of attendees, please ensure your DHSI registration name or DHSI preferred name and your Zoom name match so that we know to let you into the virtual sessions.

DHSI Materials

DHSI materials (ex. videos, documents, etc.) are intended for registrant use only. By registering, you have agreed that you will not circulate any DHSI content. If someone asks you for the materials, please invite them to complete the registration form to request access or contact us at **institut@uvic.ca**.

Auditor and participant registration

If you registered to **audit** any workshops, note that auditor involvement is intended to be fully self-directed without active participation in the workshop. The auditor option offers more flexibility regarding pace and time with the workshop content. Your registration as an auditor will include access to some asynchronous workshop materials only and does <u>not</u> include access to live workshop sessions and/or individual/group instruction or consultation. Please direct any questions about DHSI workshop auditing to **institut@uvic.ca**.

If you registered as a **participant** in any workshops, your registration includes access to asynchronous content + active participation in live workshop session(s). The workshop instructor(s) will contact you about the date(s), time(s), and platform(s) of the live workshop session(s).

If you are unsure whether you registered as an auditor or participant, please check your registration confirmation email. Further questions can be directed to **institut@uvic.ca**.

Schedule

The at-a-glance schedule of DHSI 2023 courses, workshops, institute lectures and aligned conferences & events can be found here: **https://dhsi.org/timetable/**

All times are listed in North American **Pacific Time Zone**.

For those who registered as participants in any workshops, live sessions for online workshops are not currently listed on the above-referenced schedule. **Instructors will be in touch with registered participants directly about the exact date(s) and time(s) of their live workshop session(s).**

Acknowledgements

We would like to thank our partners and sponsors (including the Social Sciences and Humanities Research Council), workshop instructors, aligned conference & event organizers, institute lecturers, local facilitators, and beyond for making this possible.

Further information

General DHSI 2023 information: **https://dhsi.org/program/**

Full course listings (in-person): **https://dhsi.org/on-campus-courses/**

Full workshop listings (online): **https://dhsi.org/online-workshops/**

Aligned conferences & events (in-person): **https://dhsi.org/on-campus-aligned-conferences-events/**

Aligned conferences & events (online): **https://dhsi.org/online-aligned-conferences-events/**

Institute lectures: **https://dhsi.org/institute-lectures/**

Frequently asked questions: **https://dhsi.org/faq/**

Any questions not addressed in the above pages? Please email us at **institut@uvic.ca**!

Intro to Digital Approaches to Music Research

DHSI 2023

Timothy Duguid - tim.duguid@glasgow.ac.uk

Contents

The Coursepack

Welcome to digital approaches to music research! I am excited to have you on this course and hope it will allow you to expand your research workflows to include digital editing and analyses.

This document includes an annotated syllabus for the course, instructions for obtaining software that will be helpful if you wish to start your own digital work, and a number of optional readings to contextualize music encoding and its applications.

This course will take place online and will not require you to install any software. However, installation instructions have been included here for your reference so that you can create a digital research environment on your own machine.

The readings and tutorials are also provided for your reference.

Syllabus

The workshop will run on Friday, June 10 and will consist of three parts.

Part One: Digital Basics
We will begin by introducing motivations and methodologies for text encoding and how they apply to music notation. Together, we will explore some existing examples of encoded notational music to obtain a better understanding of XML and the main components of MEI files.

Part Two: Generating and using music encoding
We will move on to consider the common workflows employed for generating encoded music files and then consider how metadata can be used to augment and assist with encoded music data. We will then bring all this together by exploring recent research projects that incorporate these practices into their workflows.

Part Three: Analyzing digitized scores
The workshop will conclude by exploring some of the more commonly used tools for analyzing encoded music. In particular, we will explore the different types of digital analyses - close and distant - and consider how the tools help and hinder these types of analytical tasks.

Software

Sibelius

Sibelius is music notation software. It is used by professional typesetters around the world, therefore, obtaining a license is expensive. Nonetheless, some academic music departments provide copies of Sibelius to faculty and students; it is worth checking with the music school at your institution.

Thankfully, Sibelius offers a 30-day free trial that we can use for this course. Even after the trial is expired, you will still be able to use Sibelius to export files to MEI, but you won't be able to save files in the Sibelius format.

We will be using Sibelius in this course because of a plug-in that is able to export Sibelius music scores directly to MEI. Currently, Sibelius is the only music notation software able to do this.

Download Sibelius

Get the Sibelius for Windows or Mac from https://connect.avid.com/Sibelius-Trial.html. If you are using a Linux machine, see the section on open source alternatives below and please get in touch with the instructors before DHSI.

Add the MEI plug-in

Download the plug-in from https://github.com/music-encoding/sibmei/releases/download/v2.1.0/sibmei-2.1.0.zip.
To install this plugin, copy or symlink the .plg files in the Sibelius Plugin directory on your machine. The specific location depends on your OS and version of Sibelius, see this page for instructions: http://www.sibelius.com/download/plugins/index.html?help=install.

Open Source alternatives

MuseScore is among the best open source music notation software options, and it can be downloaded from https://musescore.org. Like many music notation programs, MuseScore can export to the MusicXML format, an interchange music format (we will talk about this in class, too). MusicXML can be converted into MEI using the Verovio toolkit, something we will discuss in the course.

Text editing software

When working with encoded music files, many find that a dedicated text editor can provide helpful tools for parsing the files that can often include thousands of lines of information.

Visual Studio Code (VS Code) is a text editor that is developed by Microsoft but is available to be downloaded and used for free. Its features for debugging, syntax highlighting, code completion and embedding of Git have made it a frequent choice for coders working with XML, JavaScript, and many other encoding and programming languages. VS Code can be downloaded from https://code.visualstudio.com/.

Atom is a text editor with support for a number of programming and markup languages, including XML. It is free and open source. Through a plug-in, it can be used to validate XML files against a schema—for example to make sure the file being edited follows MEI rules. The same plug-in also offers autocomplete suggestions, which makes it easier to figure out which MEI elements and attributes to use. A very helpful resource for setting up an Atom-based environment is provided by Moacir P. de Sá Pereira in his online course, "The JavaScripting English Major" (https://the-javascripting-english-major.org/v1/1-environment). The following will work through some of the highlights of that tutorial, tweaking a few things to support the functions that will be needed for music encoding.

This section will guide you through a number of steps to install and configure Atom, but if you choose to use VS Code, you may find the same or similar plugins will be helpful.

Download Atom

Atom can be downloaded at https://atom.io/. Versions are available for Windows, Mac, and Linux. Select and install the appropriate version for your operating platform, as you would any other application.

Install Java Development Kit (JDK)

The plug-in to validate XML requires Java code, a very common programming language. The JDK can be downloaded here: http://www.oracle.com/technetwork/java/javase/downloads/jdk9-downloads-3848520.html. Make sure to select the correct platform (Windows, Mac OS, etc.) and follow the instructions to install it.

Add XML schema plug-in to Atom

- Open Atom and access its settings by hitting Ctrl + (Windows), or ⌘+ (Mac)
- Select "Install"
- In the search bar, enter "linter-autocomplete-jing"
- When the corresponding result appears, click on "Install"
- If prompted to "install dependencies," click "Yes"
- When the installation is complete, quit and restart Atom
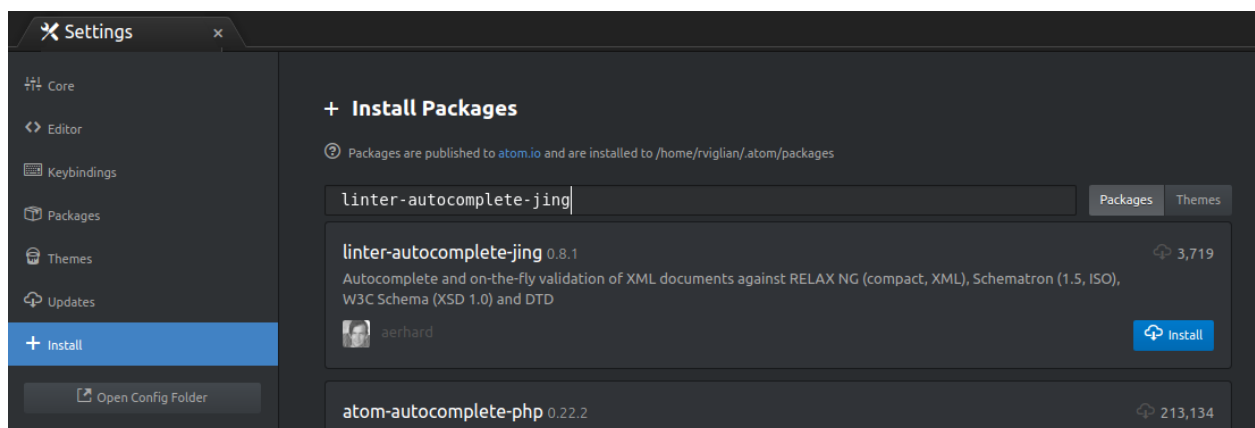
Test XML schema plug-in

- In Atom, open a new file from the main menu bar: "File" → "New File."

- Copy and paste the MEI template from this URL:
  https://gist.github.com/raffazizzi/322536448f140b0f2f9a5b7b629bb327.
  The text is also included below, but copying from this PDF may cause issues;
  we recommend copy-pasting from the URL above.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<?xml-model
href="https://raw.githubusercontent.com/music-encoding/music-enc
oding/develop/schemata/mei-CMN.rng" type="application/xml"
schematypens="http://relaxng.org/ns/structure/1.0"?>
<?xml-model
href="https://raw.githubusercontent.com/music-encoding/music-enc
oding/develop/schemata/mei-CMN.rng" type="application/xml"
schematypens="http://purl.oclc.org/dsdl/schematron"?>
<mei xmlns="http://www.music-encoding.org/ns/mei">
    <meiHead>
    <fileDesc>
        <titleStmt>
            <title></title>
        </titleStmt>
        <pubStmt></pubStmt>
    </fileDesc>
    </meiHead>
    <music></music>
</mei>
```

*The first lines (in grey above) (1) indicate that this is an XML file, and (2)
associate this file with the latest MEI schema available online. In class, we will
use a local schema for better speed.*

- Navigate to "File" → "Save As…" to save the file with an ".xml" extension to
  activate the plugin (e.g. "*test.xml*"). Perform the operations below to make
  sure the plug-in is working properly.
- Check for error messages (indicated by a red-colored circle along the
  left-hand side) when the XML is not "well formed", e.g. by removing a closing
  tag.

```
 4      <mei xmlns="http://www.music-encoding.org/ns/mei">
 5          <meiHead>
 6              <fileDesc>
 7                  <titleStmt>
 8                      <title></title>
 9                  </titleStmt>
10                  <pubStmt></pubStmt>
11              </fileDesc>
12          </meiHead>
13          <music>
14    ●</mei>
15          The element type "music" must be terminated by the matching end-tag "</music>".
```

- Start typing a new attribute or element to get autocompletion suggestions.

```
13          <music>
14            <body>
15              <mdiv>
16                <score>
17    ●             <|
18                <>  add
19             <
20          </b  <>  anchoredText
21        </mus  <>  annot
22   </mei>
23             <>  app

             <>  cb

             <>  choice

             <>  colLayout

             <>  corr

             <>  curve

             <>  damage

             (addition) – Marks an addition to the
             text.
```

Add HTML preview plug-in to Atom

- Open Atom and access its settings by hitting Ctrl + (Windows), or ⌘ + (Mac)
- Select "Install"
- In the search bar, enter "atom-html-preview"
- When the corresponding result appears, click on "Install"

Test XML schema plug-in

- In Atom, open a new file from the main menu bar: "File" → "New File."
- Copy and paste the HTML template below and save it as index.html.

```html
<!doctype html>
<html lang="en">

  <head>
    <title>My site</title>
    <meta charset="utf-8">
    <meta name="description" content="A site for my MEI">
    <meta name="author" content="Me">
  </head>

  <body>
    <p>Hello world.</p>
  </body>
</html>
```

- Hit `Ctrl + Shift + H` (also on Mac) to toggle the preview. You should see "Hello World." appearing on a blank page on the right hand side.
  A word of caution: this plug-in occasionally will display the file opened in the first tab instead of the one currently opened. Where this to happen, drag the tab of your HTML file to the first position on the left.

Python and Music21

Music21 is a powerful tool that can allow you to analyze single pieces and large corpora of music. However, it runs on Python. Therefore you will need to install Python, a few dependencies for Atom, and then the music21 library (preferably in that order).

1. Python installation
   MacOS
   Go to https://www.python.org/downloads/mac-osx/ to download the latest release of Python. Once downloaded, run the installer.

Windows

Go to https://www.python.org/downloads/ to download the latest release of Python. Once downloaded, run the executable file.

2. Python IDE installation for Atom

- Open Atom and access its settings by hitting Ctrl + (Windows), or ⌘ + (Mac)
- Select "Install"
- In the search bar, enter "ide-python"
- When the corresponding result appears, click on "Install"
- If prompted to "install dependencies," click "Yes"
- Back in the search bar, enter "atom-python-run"
- When the corresponding result appears, click on "Install"
- If prompted to "install dependencies," click "Yes"
- When the installation is complete, quit and restart Atom

3. Install music21 library

MacOS:

- Open up "Terminal" in "Applications -> Utilities"
- In the prompt, type: `sudo pip3 install music21`
- It will prompt you for your password

Windows

- Open up a command prompt and type: `pip install music21`
- With Python installed, launch IDLE by clicking on the start menu and clicking Run (Windows XP or older) and typing "IDLE" on the Search Programs list (Windows Vista and newer)
- Make sure the Python version begins with 3.4 or higher (if it is older, download an updated version using the instructions above)
- Type: `import music21`
- If your system waits a few seconds and displays ">>>", you have installed music21 properly. There may be a few warning messages about missing packages, but that is alright. If, however, the system cannot find music21, you may have more than one version of Python on

your system. Try uninstalling them along with music21 and restart the process

4. Configure music21

MacOS
   - When you downloaded and installed music21, a new folder should have been created on your desktop. In that folder, double-click the file "installer.command"
   - The system will likely warn you about running it, but click on "Open"
   - A Python window will then open and the Configuration Assistant will start.
   - Follow the instructions under "All OS" to complete the configuration

Windows
   - Configure music21 by typing the following in IDLE: `import music21 music21.configure.run()`
   - Follow the instructions under "All OS" to complete the configuration

All OS
   - When asked to install music21 in the standard location, enter "y" or "yes" and press return.
   - You may be asked for your Mac password
   - Installation will begin, and a large amount of text will display files being copied.
   - Music21 will then ensure you have a MusicXML reader.
   - If you have not installed Sibelius or Finale, you can enter "y" or "yes" to be taken to a website to download and install Finale Reader.
   - If you have installed Sibelius or Finale installed, enter "n" or "no" and music21 will show a list of compatible readers. Select the program you wish to use.
   - The remaining questions will ask if you wish to allow music21 to access the internet (up to you) and if you agree to the license for music21 and the included scores (answer 'yes').

GitHub and GitHub Pages

GitHub is an online service for versioning code, but can also be used to host simple websites through GitHub Pages. In this course, we will learn how to store MEI data on GitHub and publish a digital score website with GitHub Pages.

Getting Started

- Navigate to https://github.com and create an account. Follow the prompts to create a free GitHub account. Be sure to make the email address associated with your account public, so that you'll later be able to publish a static site.
- Navigate to https://desktop.github.com to download GitHub Desktop. Then, open the application and sign in.

Git / GitHub operations

- **REPOSITORY:** code and data in one place
- **FORK:** make a copy of someone else's repository to your account. We will provide an example repository that you will be able to fork
- **CLONE:** copy a repository to your machine
- **ADD**: add a file to be tracked by Git
- **COMMIT**: save local changes to your local clone
- **PUSH**: send your committed changes to your online repository on GitHub
- **PULL**: get changes from the online repository to your cloned repository
- **SEND PULL REQUEST**: contribute changes to an original repository that you forked

Publishing with GitHub pages

If your repository contains a website, you can publish it on GitHub pages. In your GitHub account (in your internet browser), navigate to your repository containing a website. Click on the "Settings" tab. Scroll down to "GitHub Pages" and pop-out "none" to select "Master branch." Click Save. The page may refresh before offering you a link to your page.

MEI Tutorials

The following MEI tutorials are lightly adapted from "MEI 1st", which used to be hosted on the MEI website. They are written by Maja Hartwik and Kristina Richts. It should be noted that this relies on an older version of MEI. Most of the basic structures and commonly used elements remain unchanged. Some minor changes have occurred since the release of version 4.0, so you should reference the current guidelines whenever you encounter errors in your encoding: https://music-encoding.org/guidelines/v4/content/

You may also want to consult the various pedagogical tools available from the Music Encoding website: https://music-encoding.org/resources/pedagogy.html

General MEI Document Structure

MEI provides an XML-based schema that describes music notation, using both XML elements and attributes. An element, often referred to as XML tag, is notated in angle brackets:

```
<elementName></elementName>
```

The name of an element is case sensitive, and every element consists of an opening and closing tag. The closing tag contains a solidus character (forward slash), which precedes the element name. An element may contain one or more other elements, XML comments, or text. Content is nested within the opening and closing tag of the element.

```
<elementName>
  <subElement>Text</subElement>
  <subElement>Text</subElement>
</elementName>
```

Elements must be nested properly, with the element opened last being closed first, in order to comply with the rules of XML.

An element may be described further by adding attributes to it. These attributes are attached to the element's name in the opening tag, separated by a space character.

Attributes can be listed in any order within a particular element. Whereas elements may be specified further by attributes, there is no mechanism for providing additional information about attributes. Therefore, attributes are less versatile than elements. In running text, attributes are often referred to by attaching a preceding @ character.

Elements lacking content between their opening and closing tags (containing nothing, not even spaces or tabs), may be written as empty elements:

`<sb></sb>` or `<sb/>`

In the last case, the backslash at the end of the element's name indicates that the opening tag is also functioning as the closing tag. This mechanism is independent of the use of attributes.

An XML document following these general rules of XML is "well-formed". In addition to being well-formed, an XML document may be "valid" with regard to a schema, such as MEI, meaning that it also reflects the structure defined by that particular schema. The following instructions assume well-formed and valid MEI documents. Every MEI document consists of two main parts:
- the MEI Header and
- the MEI music.

1 The MEI Header

In addition to providing a means of electronically transcribing or encoding the musical content of a given piece of music notation, MEI enables the capture of information about this encoded material (title, source, related works, data about creation, etc.), using the MEI Header, represented by the <meiHead> element. While this part of the schema is modeled on the respective sections of the [Text Encoding Initiative (TEI)](), the [Encoded Archival Description (EAD)](), or other metadata encoding and transmission standard formats, it is specifically adjusted to address the needs of musicologists, music librarians, and others in the field.

Every MEI file requires a minimal header. The header in the example below provides information about the title of the digital resource and details of its publication status. Of course, other details may be specified using a variety of optional elements.

2 Music

2.1 Basic Structure of an MEI Document

The example below shows a typical, simple MEI document, using the **<mei>** document element, which consists of some metadata, represented by the **<meiHead>**, and the **<music>** itself. The **<music>** element is the container for the complete musical text and makes some important constituents available. The following example shows the mei document element with the appropriate namespaces (briefly, the namespace is a definition required for all XML documents), the required **<meiHead>**, and an empty **<music>** element.

```
<mei xmlns="http://www.music-encoding.org/ns/mei">
    <meiHead>
        <fileDesc>
            <titleStmt>
                <title>title of the encoded example</title>
                </titleStmt>
                <pubStmt/>
        </fileDesc>
    </meiHead>
    <music/>
</mei>
```

While the **<music>** element is required in this case, it may be empty. The **<music>** element is simply a placeholder for content that is not yet provided.

2.2 Body and Group

The structure below illustrates a musical text with a **<music>** element and the sub-element **<body>**. The **<body>** element must contain at least one segment, known as a "musical division" (**<mdiv>**).

```
<music>
    <body>
        <mdiv>
            <score>
```

16

```
            <scoreDef meter.count="2" meter.unit="4"
key.sig="3s">
                <staffGrp symbol="line">
                    <staffDef n="1" label="Singstimme." lines="5"
clef.shape="G" clef.line="2"/>
                    <staffGrp symbol="brace" label="Pianoforte.">
                        <staffDef n="2" lines="5" clef.shape="G"
clef.line="2"/>
                        <staffDef n="3" lines="5" clef.shape="F"
clef.line="4"/>
                    </staffGrp>
                </staffGrp>
            </scoreDef>
        </score>
    </mdiv>
  </body>
</music>
```

The **<score>** element contains a score definition (**<scoreDef>** element), which acts as a container for score meta-information (i.e., information related to the whole score, such as meter). The score definition (**<scoreDef>**) may, in turn, include one or more staff definitions (**<staffDef>**, which can be grouped by one or more staff groups (**<staffGrp>**). As shown in the example, the **<staffGrp>** element can be used to specify the symbol used to group a set of staves. It is also possible to encode staff groups within other staff groups. Specific information about individual staves of a score, such as clefs or key signatures, is encoded within staff definitions (**<staffDef>**). Names of single instrument voices may be encoded in staff definitions or in staff groups.

Multiple musical division (**<mdiv>**) may be encoded within a file. In describing the separate movements of a symphony, for example, separate **<mdiv>** elements are used to demarcate the different segments. **<mdiv>** elements may also be nested within another when the structure requires it, as in the following sample of an opera:

```
<music n="opera">
    <body>
        <mdiv n="act_1">
            <mdiv n="scene_1">
                <score/>
            </mdiv>
```

```
        </mdiv>
        <mdiv n="act_2">
            <score/>
        </mdiv>
        <mdiv n="act_3">
            <score/>
        </mdiv>
    </body>
</music>
```

The **\<mdiv>** element is a critically important part of the structure of an MEI file. The content of the divisions is also significant, as that decision will enable one of two possible views: **\<score>** and **\<parts>**. Each of the elements may contain elements and attributes for describing features such as clefs, key signature, meter, etc.

If you would like to encode a single violin part, the main structure could look like this:

```
<music>
    <body>
        <mdiv>
            <parts>
                <part>
                    <staffDef n="1" label="Violin" lines="5"
meter.count="3" meter.unit="4" clef.shape="G" clef.line="2"
key.sig="1s"/>
                </part>
            </parts>
        </mdiv>
    </body>
</music>
```

When encoding performer's parts, i.e. those of a string quartet, a separate **\<part>** element must be created for each of the instrument voices. Each of these may include its own staff definition (**\<staffDef>**).

In contrast to the **\<body>** element, the **\<group>** element enables the encoding of a group of essentially-separate musical texts. If, for example, you are encoding a collection of songs, you can nest one or more **\<music>** elements, along with their respective **\<body>**, **\<mdiv>**, **\<score>**, or other elements, within **\<group>**, as seen in the example below.

```
<music>
    <group>
        <music n="song_1">
            <body>
                <mdiv/>
            </body>
        </music>
        <music n="song_2">
            <body>
                <mdiv/>
            </body>
        </music>
    </group>
</music>
```

As you can see, while the MEI basic framework is very simple, it can be used to model complex situations.

2.3 Front and Back

The elements **<front>** and **<back>** are children of the **<music>** element and are located on the same hierarchical level as the elements **<body>** and **<group>**. They are used to describe extensive introductory texts, for instance, those in critical editions and collected works. Both the **<front>** element and the **<back>** element require a mandatory division (**<div>**) child element, which itself requires a paragraph (**<p>**) child element containing the actual text you are encoding. An example of this hierarchical structure is illustrated below:

```
<music>
    <front>
        <div>
            <p><!-- introductory text here --></p>
        </div>
    </front>
    <body>
        <!-- music notation -->
    </body>
    <back>
        <div>
            <p><!-- "back matter" here --></p>
        </div>
```

```
        </back>
</music>
```

The **<section>** element is a child element of the **<score>** or **<part>** element and may be used to divide a score or parts into further linear segments, so-called sections. Within the **<section>** element, several methods of organization are possible, depending upon the notational style of the source material and the encoder's needs. A section can be described as the container for the actual musical data of an individual segment in a musical work.

```
<music>
    <body>
        <mdiv>
            <score>
                <section>
                        <!-- musical data for this segment of the
musical work -->
                </section>
            </score>
        </mdiv>
    </body>
</music>
```

Basic MEI Document Structure

A minimal MEI file should contain the following:

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-model
href="https://raw.githubusercontent.com/music-encoding/music-enc
oding/develop/schemata/mei-CMN.rng" type="application/xml"
schematypens="http://relaxng.org/ns/structure/1.0"?>
<?xml-model
href="https://raw.githubusercontent.com/music-encoding/music-enc
oding/develop/schemata/mei-CMN.rng" type="application/xml"
schematypens="http://purl.oclc.org/dsdl/schematron"?>
<mei xmlns="http://www.music-encoding.org/ns/mei">
    <meiHead>
```

```xml
        <fileDesc>
            <titleStmt>
                <title/>
            </titleStmt>
            <pubStmt/>
        </fileDesc>
    </meiHead>
    <music>
    <!-- music content goes in here -->
    </music>
</mei>
```

Encoding Notation Basics

(This tutorial is lightly adapted from "MEI 1st", written by Maja Hartwik and Kristina Richts)

Using the Basic MEI Document above, we will now focus on some of the basics of encoding music using the song "Don't Worry, Be Happy".

1. Encoding of preliminary information

It is necessary to encode a number of elements before encoding the notes themselves. You may recall that we covered the child elements from **<body>** to **<section>** and explained the use of the **<section>**element in the chapter "Common Structure and Elements". So you should now add the elements **<body>**, **<mdiv>**, **<score>** and **<section>** (in this sequence) into the **<music>** element of your empty MEI file. Next, insert a **<measure>** element into the **<section>** element (recall that **<section>** contains the actual musical data). A **<measure>** element always requires a **<staff>** child element. Of course, you could encode multiple staves, but for now, let's start with just one. The **<staff>** element requires a **<layer>** child element. Once again, you could encode more than one layer, but to keep things simple, we'll begin with a single layer, which contains a stream of musical events on a staff. So, at this point, your **<measure>** element should contain a **<staff>** and **<layer>** element.

```xml
<?xml version="1.0" encoding="UTF-8"?>
```

```
<?xml-model
href="https://raw.githubusercontent.com/music-encoding/music-enc
oding/develop/schemata/mei-CMN.rng" type="application/xml"
schematypens="http://relaxng.org/ns/structure/1.0"?>
<?xml-model
href="https://raw.githubusercontent.com/music-encoding/music-enc
oding/develop/schemata/mei-CMN.rng" type="application/xml"
schematypens="http://purl.oclc.org/dsdl/schematron"?>
<mei xmlns="http://www.music-encoding.org/ns/mei">
    <meiHead>
        <fileDesc>
            <titleStmt>
                <title>Don't worry, Be Happy</title>
                </titleStmt>
                <pubStmt/>
        </fileDesc>
    </meiHead>
  <music>
    <body>
        <mdiv>
          <score>
            <section>
                <measure>
                 <staff>
                 <layer>
                 </layer>
                 </staff>
                 </measure>
             </section>
            </score>
          </mdiv>
      </body>
    </music>
</mei>
```

Key signatures are encoded by using the **@accid** (accidental) attribute. The "f"
stands for "flat", "s" for "sharp" and "n" for "natural".

2. Notes

A note has to be defined with its elementary attributes: **@pname** (pitch name), **@oct**(octave) and **@dur** (duration). The pitch name is just the typical A, B, C, etc. For the octave, the Acoustical Society of America adopted a system that numbers the octaves from the lowest to the highest. The following octave starting with middle C and extending up to treble-clef, third-line B would be octave 4.
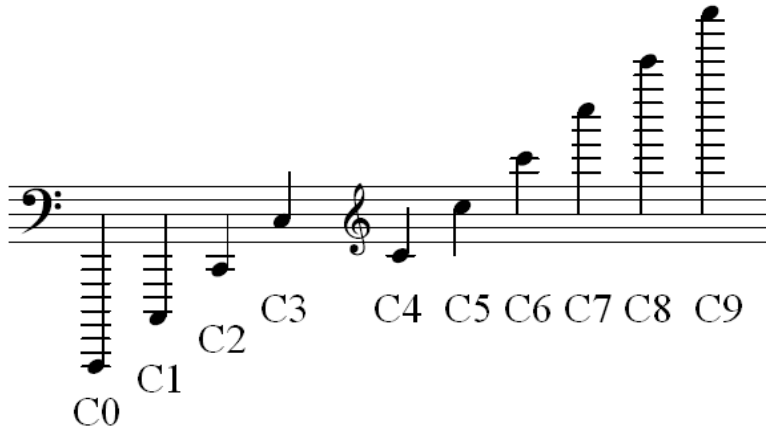

Image courtesy of User:Angr - Own work, Public Domain,
https://commons.wikimedia.org/w/index.php?curid=1451757

Finally, the duration value is the number of notes that would fit in a 4/4 measure. So, quarter notes would be 4; half notes, 2; whole notes, 1; eighth notes, 8; etc. The first note of the famous refrain of our song, a quarter note on E, would therefore be encoded as:

```
<measure>
   <staff>
      <layer>
         <note pname="e" oct="4" dur="4"/>
      </layer>
   </staff>
</measure>
```

3. Beams

The first note of the measure, which you've already encoded, is followed by two eighth notes on C:

```
<measure>
   <staff>
      <layer>
          <note pname="e" oct="4" dur="4"/>
          <note pname="c" oct="4" dur="8"/>
          <note pname="c" oct="4" dur="8"/>
      </layer>
   </staff>
</measure>
```

While this realization works, it creates two separate eighth notes. More commonly, successive eighth notes (along with smaller denominations) are grouped by beams. The **\<beam\>** element is realized as a grouping element around the two notes. Therefore, this is accomplished by encoding the notes and adding a **\<beam\>** element around them:

```
<measure>
   <staff>
      <layer>
          <note pname="e" oct="4" dur="4"/>
          <beam>
              <note pname="c" oct="4" dur="8"/>
              <note pname="c" oct="4" dur="8"/>
          </beam>
      </layer>
   </staff>
</measure>
```

*Note for oXygen users: If you select the note elements and use the shortcut Ctrl + E (or Cmd + E on the Mac), you will get a list of elements which can be legitimately wrapped around them. The **\<beam\>** element will be in that list.*

4. Rests

In order to finish the first measure of our song, we need to add some rests. Like a note, the **<rest>** element will be encoded with specific attributes. In this case, we need to encode the durations of the rests:

```
<measure>
   <staff>
      <layer>
         <note pname="e" oct="4" dur="4"/>
         <beam>
            <note pname="c" oct="4" dur="8"/>
            <note pname="c" oct="4" dur="8"/>
         </beam>
         <rest dur="4"/>
         <rest dur="4"/>
      </layer>
   </staff>
</measure>
```

You may have already noted that the refrain of "Don't Worry, Be Happy" begins with a pick-up note. Or, to be more accurate, the quarter note falls on the fourth beat of one measure and the eighth notes fall on the downbeat of the following measure. We therefore need to be able to place the notes we have already encoded into their correct metrical positions.

```
<measure>
   <staff>
      <layer>
         <rest dur="2"/>
         <rest dur="4"/>
         <note pname="e" oct="4" dur="4"/>
      </layer>
   </staff>
</measure>
<measure>
   <staff>
```

```
            <layer>
                <beam>
                    <note pname="c" oct="4" dur="8"/>
                    <note pname="c" oct="4" dur="8"/>
                </beam>
                <rest dur="4"/>
                <rest dur="4"/>
            </layer>
        </staff>
    </measure>
```

5. Staff Definition (staffDef)

As you might have noticed, we haven't yet encoded the clef and meter information notated at the beginning of the measure. This information is commonly divided among different elements. For the meter, you should add **@meter.count** (top number in time signature) and **@meter.unit** (bottom number in signature) to a new **<scoreDef>** element preceding the **<section>** element. it is recommended that a staff definition, represented by the **<staffDef>** element, be encoded as a child of the **<section>** element. The **<staffDef>** element is briefly introduced in the General MEI Document Tutorial. To add the clef, use **@clef.shape** and you should also make sure the clef is placed on the correct line of the staff using **@clef.line**.

```
<music>
    <body>
        <mdiv>
          <score>
            <scoreDef meter.count="4" meter.unit="4">
              <staffDef clef.shape="G" clef.line="2"/>
            </scoreDef>
            <section>
              <measure>
                <staff>
                    <layer>
                        <rest dur="2"/>
                        <rest dur="4"/>
                        <note pname="e" oct="4" dur="4"/>
```

```
                </layer>
              </staff>
           </measure>
...
```

The default key signature has no flats and sharps, so we do not need to include it. However, if we did, you would use **@key.sig** and indicate the number of flats or sharps ("1f" = one flat, "5s" = five sharps).

6. Name or number (@n)

The next step will be to add the remainder of the refrain, which is just a repetition of the first three notes. To help distinguish what is a growing number of measures, we will use the **@n** attribute to number the measures.

```
<staffDef clef.shape="G" clef.line="2"/>
<section>
    <measure n="1">
        <staff>
            <layer>
                <rest dur="2"/>
                <rest dur="4"/>
                <note pname="e" oct="4" dur="4"/>
            </layer>
        </staff>
    </measure>
    <measure n="2">
        <staff>
            <layer>
                <beam>
                    <note pname="c" oct="4" dur="8"/>
                    <note pname="c" oct="4" dur="8"/>
                </beam>
                <rest dur="4"/>
                <rest dur="4"/>
                <note pname="e" oct="4" dur="4"/>
            </layer>
        </staff>
```

```
        </measure>
        <measure n="3">
            <staff>
                <layer>
                    <beam>
                        <note pname="c" oct="4" dur="8"/>
                        <note pname="c" oct="4" dur="8"/>
                    </beam>
                    <rest dur="4"/>
                    <rest dur="2"/>
                </layer>
            </staff>
        </measure>
...
```

You can add **@n** to any element, and it is particularly helpful for distinguishing parts in a score.

7. Lyrics

So far, so good! But to be able to sing the song, we need some text. So the next step will deal with some lyrics. The best way to encode lyrics is to add **<syl>** elements within **<note>** elements.

```
<measure n="1">
    <staff>
        <layer>
            <rest dur="2"/>
            <rest dur="4"/>
            <note pname="e" oct="4" dur="4">
                <syl>Don't</syl>
            </note>
        </layer>
    </staff>
</measure>
<measure n="2">
    <staff>
        <layer>
```

```
            <beam>
                <note pname="c" oct="4" dur="8">
                    <syl>Wor-</syl>
                </note>
                <note pname="c" oct="4" dur="8">
                    <syl>ry</syl>
                </note>
            </beam>
            <rest dur="4"/>
            <rest dur="4"/>
            <note pname="e" oct="4" dur="4">
                <syl>Be</syl>
            </note>
        </layer>
...
```

8. Chords

Now you have successfully encoded three measures of notes and text in MEI. Congratulations!

Later on in the song, there are chords, accidentals and dotted rhythms. To simplify matters, the first image will show only the notes, without rhythm or accidentals. Chords must be encoded with a **<chord>** element around the **<note>** elements. If there are beamed chords, encode the beam first, then the chords, and finally the notes. Encode the following as measure 10.


```
<measure n="10">
    <staff>
        <layer>
            <rest dur="2"/>
            <beam>
                <chord>
                    <note pname="e" oct="4" dur="8"/>
                    <note pname="g" oct="4" dur="8"/>
                </chord>
```

```
            <chord>
                <note pname="d" oct="4" dur="8"/>
                <note pname="f" oct="4" dur="8"/>
            </chord>
            <chord>
                <note pname="c" oct="4" dur="8"/>
                <note pname="e" oct="4" dur="8"/>
            </chord>
            <chord>
                <note pname="a" oct="3" dur="8"/>
                <note pname="c" oct="4" dur="8"/>
            </chord>
        </beam>
    </layer>
  </staff>
</measure>
```

9. Rhythm

Now let´s encode the dotted rhythms by adding a **@dots** attribute to the **<note>** element. The value represents the number of augmentation dots accompanying the note and must be an integer less than or equal to 4. Note also that the number of beamed groups has changed. Insert the second beam and add a **@dots** attribute to the relevant notes.


```
<measure n="10">
    <staff>
        <layer>
            <rest dur="2"/>
            <beam>
                <chord>
                    <note pname="e" oct="4" dur="8" dots="1"/>
                    <note pname="g" oct="4" dur="8" dots="1"/>
                </chord>
                <chord>
                    <note pname="d" oct="4" dur="16"/>
```

```
                <note pname="f" oct="4" dur="16"/>
            </chord>
        </beam>
        <beam>
            <chord>
                <note pname="c" oct="4" dur="8" dots="1"/>
                <note pname="e" oct="4" dur="8" dots="1"/>
            </chord>
            <chord>
                <note pname="a" oct="3" dur="16"/>
                <note pname="c" oct="4" dur="16"/>
            </chord>
        </beam>
    </layer>
  </staff>
</measure>
```

10. Accidentals

Finally, let's add the accidentals:

MEI offers a lot of choices in the encoding of musical phenomena. In this case, you get to decide whether the accidental should be encoded as an attribute or as an element. Using **@accid**, a "f" stands for "flat", "s" for "sharp" and "n" for "natural". If you prefer to encode the accidental as an element - perhaps if you'd like to specify the location of the accidental by the use of the **@place** attribute - you'll have to encode an **<accid>** child element inside the **<note>** element.

```
<beam>
    <chord>
        <note pname="e" oct="4" dur="8" dots="1" accid="f"/>
        <note pname="g" oct="4" dur="8" dots="1"/>
    </chord>
    <chord>
        <note pname="d" oct="4" dur="16"/>
```

```
            <note pname="f" oct="4" dur="16"/>
        </chord>
    </beam>
    <beam>
        <chord>
            <note pname="c" oct="4" dur="8" dots="1"/>
            <note pname="e" oct="4" dur="8" dots="1" accid="n"/>
        </chord>
...
```

Additional Notation Encoding Practice

For the next example, we'll use the opening section of the famous chorale "Ein' feste Burg ist unser Gott", composed by Johann Sebastian Bach.

1. Score definition

Start your encoding with the creation of a score definition (**<scoreDef>**). As you've learned in the previous exercise, you need to encode a **<scoreDef>** element with its sub-elements and appropriate attributes within the **<score>** element. You might also want to indicate that the two staves are grouped by a brace. To do this, add a staff group (**<staffGrp>** element) as a child element to your score definition (**<scoreDef>**) and use the **@symbol** attribute with the value "brace" to encode this information. You can now insert two separate staff definitions (**<staffDef>** elements) into the staff group (**<staffGrp>**). Each **<staffDef>** should include specific staff information such as information on the shape (**@clef.shape**) and location (**@clef.line**) of its clef.

```
...
<scoreDef meter.count="4" meter.unit="4" meter.sym="common"
key.sig="2s">
   <staffGrp symbol="brace">
      <staffDef n="1" lines="5" clef.shape="G" clef.line="2"/>
      <staffDef n="2" lines="5" clef.shape="F" clef.line="4"/>
   </staffGrp>
</scoreDef>
...
```

2. Staff and layer

After encoding the score definition (**<scoreDef>**), you can start to encode the first
staff. Please ignore the upbeat for a moment (we will encode it a few steps later),
and concentrate on just the first measure. As you can see in the image below, the
first staff contains two individual streams of musical events, requiring the encoding
of two layers here. Start the encoding of the first measure by putting two **<layer>**
elements within the **<staff>** element. As it is the first staff of the chorale, remember
to encode the number using the **@n** attribute. While the **<layer>** element is used to
encode multiple "voices" within a single staff this element is always required to
complete the structure, even if there is only one voice to encode.

```
...
<measure n="1">
   <staff n="1"> <!-- Treble clef stave -->
      <layer n="1"> <!-- Soprano voice -->
      </layer>
      <layer n="2"> <!-- Alto voice -->
      </layer>
   </staff>
   <staff n="2"> <!-- Bass clef stave -->
      <layer n="1"> <!-- Tenor voice -->
      </layer>
      <layer n="2"> <!-- Bass voice -->
      </layer>
    </staff>
 </measure>
```

3. Notes

Next, add each of the notes with their respective durations, pitch names and octaves. Since you are encoding multiple voices on a single stave, you may wish to use the **@stem.dir** attribute of the **<note>** element to ensure that the voices can be differentiated.

```
...
<measure n="1">
   <staff n="1">
      <layer n="1">
         <note pname="d" oct="5" dur="4" stem.dir="up"/>
         <note pname="d" oct="5" dur="4" stem.dir="up"/>
         <beam>
            <note pname="a" oct="4" dur="8" stem.dir="up"/>
            <note pname="b" oct="4" dur="8" stem.dir="up"/>
         </beam>
         <note pname="c" oct="5" dur="4" stem.dir="up"/>
      </layer>
      <layer n="2">
         <note pname="d" oct="4" dur="4" stem.dir="down"/>
         <beam>
            <note pname="d" oct="4" dur="16" stem.dir="down"/>
            <note pname="e" oct="4" dur="16" stem.dir="down"/>
         </beam>
         <note pname="f" oct="4" dur="4" stem.dir="down"/>
         <note pname="g" oct="4" dur="4" stem.dir="down"/>
      </layer>
   </staff>
...
```

4. Individual lyric syllables and lyric verses

Using the **\<syl\>** element for the encoding of individual lyric syllables can be useful if, for instance, you want to provide additional details for the lyric syllables. The **\<syl\>** element has to be encoded as a child element of the **\<note\>** element.

If there are several lyric verses to encode, MEI also provides a **\<verse\>** element, which must also be encoded as a child element of the **\<note\>** element. It should be wrapped around the **\<syl\>** element. You can number the verses, if you like, using the **@n** attribute on **\<verse\>**.

The first section of the chorale has two verses, so you'll want to add **\<verse\>** and **\<syl\>** child elements to the relevant **\<note\>** elements.

The word "feste" contains a dash that acts as a syllable separator and "Burg" has a syllable extension. To encode these, you could add the appropriate **@con** attribute onto the **\<syl\>** element, choosing from the following list: s (space) = word separator, d (dash) = syllable separator, u (underscore) = syllable extension, t (tilde) = syllable elision. You can also define whether a syllable marks the beginning, middle or end of a word by using the **@wordpos** attribute. Approved values include "i" (initial), "m" (medial) or "t" (terminal).

```
...
<measure n="1">
   <staff n="1">
      <layer n="1">
         <note pname="d" oct="5" dur="4" stem.dir="up"/>
         <note pname="d" oct="5" dur="4" stem.dir="up"/>
         <beam>
            <note pname="a" oct="4" dur="8" stem.dir="up"/>
            <note pname="b" oct="4" dur="8" stem.dir="up"/>
         </beam>
         <note pname="c" oct="5" dur="4" stem.dir="up"/>
      </layer>
      <layer n="2">
         <note pname="d" oct="4" dur="4" stem.dir="down">
            <verse n="1">
```

```xml
                    <syl con="d" wordpos="i">fe</syl>
                </verse>
                <verse n="2">
                    <syl wordpos="i">hift</syl>
                </verse>
            </note>
            <beam>
                <note pname="d" oct="4" dur="16" stem.dir="down">
                    <verse n="1">
                        <syl con="u" wordpos="t">ste</syl>
                    </verse>
                    <verse n="2">
                        <syl wordpos="i">uns</syl>
                    </verse>
                </note>
                <note pname="e" oct="4" dur="16" stem.dir="down"/>
            </beam>
            <note pname="f" oct="4" dur="4" stem.dir="down">
                <verse n="1">
                    <syl con="u">Burg</syl>
                </verse>
                <verse n="2">
                    <syl wordpos="i">frei</syl>
                </verse>
            </note>
            <note pname="g" oct="4" dur="4" stem.dir="down"/>
                <verse n="1">
                    <syl>ist</syl>
                </verse>
                <verse n="2">
                    <syl wordpos="i">aus</syl>
                </verse>
            </note>
        </layer>
    </staff>
...
```

5. Upbeat

We now need to encode an upbeat by adding the **@metcon** attribute to the **<measure>** element. This attribute indicates the relationship between the content of a staff or layer and the prevailing meter. A value of "true" for the **@metcon** attribute indicates conformance with the prevailing meter (i.e. a complete measure) while "false" means the opposite, an incomplete measure. Please encode the complete upbeat, using the appropriate **@metcon** attribute.

```
...
<measure metcon="false" n="0">
   <staff n="1">
      <layer n="1">
         <note pname="d" oct="5" dur="4" stem.dir="up"/>
      </layer>
      <layer n="2">
         <note pname="a" oct="4" dur="4" stem.dir="down">
            <verse n="1">
               <syl>Ein'</syl>
            </verse>
         </note>
      </layer>
   </staff>
   <staff n="2">
      <layer n="1">
         <note pname="f" oct="4" dur="4" stem.dir="up"/>
      </layer>
      <layer n="2">
         <beam>
            <note pname="d" oct="4" dur="8" stem.dir="down"/>
            <note pname="c" oct="4" dur="8" stem.dir="down"/>
         </beam>
      </layer>
   </staff>
</measure>
...
```

6. Fermatas

Now we're ready to deal with the fermatas by adding the **@fermata** attribute to the relevant **\<note\>** elements. Values for specifying its location include: "above", "below" or "within" the staff. Choose the appropriate value from the list to encode each fermata.

7. Slurs

The most common approach for encoding slurs is to use a combination of the **@tstamp** (timestamp) attribute and the **@tstamp2** attribute. Note that slurs are encoded at the end of a measure, following notes, chords, and rests. The **@tstamp** attribute is used to encode the onset time in terms of musical time, i.e. beats. Since our chorale is in four-four time, the main timestamps of the measure are tstamp="1", tstamp="2", tstamp="3" and tstamp="4". As an example, we'll look at the slur in the second layer of the first staff, which begins on tstamp="1" and it ends on tstamp="2.5". The duration of a feature is encoded as a timestamp giving its endpoint, using the **@tstamp2** attribute. In our example, you would enter a **@tstamp2** attribute with the value "0m+2.5" (m stands for measure). You may choose to indicate the curve direction of the slurs ("above" or "below") and you can also specify the staff and the layer where they occur.

```
...
  <staff n="2">
    <layer n="1">
      <note pname="e" oct="4" dur="8" stem.dir="up"/>
      <note pname="a" oct="3" dur="4" stem.dir="up"/>
      <note pname="g" oct="3" dur="8" stem.dir="up"/>
      <note pname="f" oct="3" dur="4" stem.dir="up"/>
    </layer>
    <layer n="2">
      <beam>
        <note pname="e" oct="3" dur="8" stem.dir="down"/>
        <note pname="d" oct="3" dur="8" stem.dir="down"/>
      </beam>
      <note pname="a" oct="2" dur="4" stem.dir="down"/>
```

```
          <note pname="d" oct="3" dur="4" stem.dir="down"
fermata="below"/>
      </layer>
   </staff>
   <slur tstamp="1" tstamp2="0m+2.5" curvedir="below" staff="1"
layer="2"/>
   <slur tstamp="1" tstamp2="0m+2.5" curvedir="above" staff="2"
layer="1"/>
   <slur tstamp="1" tstamp2="0m+3" curvedir="below" staff="2"
layer="2"/>
</measure>
...
```

8. Repetition

Finally, you can encode the repetition on the last measure, using the **@right** attribute with the appropriate value from the following list: "*dashed*", "*dotted*", "*dbl*", "*rptstart*", "*rptboth*", "*rptend*". Please add a **@right** attribute to the **<measure>** element of measure 4 and insert the value "*rptend*".

```
...
<measure n="4" right="rptend">
...
```

More Advanced Notation Encoding

Now we can move ahead with some more advanced features of MEI, such as various types of articulation, ornamentation, cross-staff beaming and dynamics. You will also learn about the encoding of tuplets and XML identifiers.

Let's start by encoding the tuplets at the beginning of Schubert's "Erlkönig":

1. Tuplets

There's a flurry of details to encode here! To simplify matters, we'll take it one step at a time. First, let's concentrate on the tuplets.

You'll have to decide whether you want to encode the tuplet as an element or as an attribute. In this encoding, we've chosen to use the **<tuplet>** element, which is wrapped around the notes of the tuplet. MEI offers several attributes for describing the details of the tuplet, among them: **@bracket.visible**, **@num**, **@num.visible** and **@num.place**. For a list of other possible attributes, take a look at chapter 4.2.4 of the **Guidelines** .

Since the dyads in the right-hand of the piano are chords, be sure to wrap each simultaneous pair in a **<chord>** element.

```
...
<measure n="1">
   <layer n="1">
   <tuplet bracket.visible="false" num="3" num.visible="true"
num.place="above">
      <beam>
         <chord stem.dir="up" dur="8">
            <note n="1" pname="g" oct="4"/>
            <note n="2" pname="g" oct="3"/>
```

```
      </chord>
      <chord stem.dir="up" dur="8">
         <note n="1" pname="g" oct="4"/>
         <note n="2" pname="g" oct="3"/>
      </chord>
      <chord stem.dir="up" dur="8">
         <note n="1" pname="g" oct="4"/>
         <note n="2" pname="g" oct="3"/>
      </chord>
    </beam>
  </tuplet>
...
```

Note that this is only one of several possible solutions.

2. Using @copyof

As you've probably noticed, an encoding can be quite verbose when there's a lot of repetition of content. When encoding the XML directly using a QWERTY keyboard, this verbosity can lead to encoding errors. It would be helpful to be able to encode something correctly once and simply copy the encoded feature wherever it's needed. This can be accomplished through the use of the **@copyof** attribute.

Using the **@copyof** attribute, however, requires the use of another attribute, **@xml:id**. Placing this attribute on an element gives it a name by which it can be referred to later. Each **@xml:id** attribute must have a value that is unique within the current document. This allows you to unambiguously refer to the element bearing this value elsewhere in your document. The value of an **@xml:id** attribute must also be an XML name. An XML name must follow some basic rules:

- contain letters, numbers, and other characters, except spaces and colons
- cannot start with a number or punctuation character
- cannot start with the letters "xml" (or XML, or Xml, etc.)

In the current example, copying the tuplet will shorten your file and reduce the need for duplicate, and error-prone, encoding. First, you need to give the first tuplet an

@xml:id attribute. Any tuplets that are exact copies of the first one can then be represented by an empty **\<tuplet\>** element whose **@copyof** attribute contains the name you gave to the first tuplet. Don't confuse the **@copyof** attribute with the **@sameas**attribute. For more information on relationships between elements, please have a look at chapter 7.1 of the **Guidelines** .

```xml
<measure n="1">
   <staff n="2">
      <layer n="1" xml:id="g-triplets">
         <tuplet bracket.visible="false" num="3"
num.visible="true" num.place="above" xml:id="triplet1">
            <beam>
               <chord stem.dir="up" dur="8">
                  <note n="1" pname="g" oct="4"/>
                  <note n="2" pname="g" oct="3"/>
               </chord>
               <chord stem.dir="up" dur="8">
                  <note n="1" pname="g" oct="4"/>
                  <note n="2" pname="g" oct="3"/>
               </chord>
               <chord stem.dir="up" dur="8">
                  <note n="1" pname="g" oct="4"/>
                  <note n="2" pname="g" oct="3"/>
               </chord>
            </beam>
         </tuplet>
         <tuplet copyof="#triplet1"/>
         <tuplet copyof="#triplet1"/>
         <tuplet copyof="#triplet1"/>
      </layer>
...
```

3. Instrument/Voice labels

Let's return to the beginning of the song and go on to the next feature. You've already learned to encode score definitions (**\<scoreDef\>**) and staff definitions (**\<staffDef\>**). Now we'll talk about encoding the names of instruments and voices.

You can use the attribute **@label** to encode information about instrumentation that appears at the beginning of the song. **@label** may occur as an attribute of the element **<staffDef>**. It can also appear on **<staffGrp>**, when an instrument, like the piano, is written on multiple staves.

```
...
<score>
   <section>
      <scoreDef meter.unit="4" meter.sym="common">
         <staffGrp>
            <staffDef n="1" label="Singstimme." clef.shape="G"
clef.line="2" key.sig="2f"/>
            <staffGrp label="Piano Forte." symbol="brace">
               <staffDef n="2" clef.shape="G" clef.line="2"
key.sig="2f"/>
               <staffDef n="3" clef.shape="F" clef.line="4"
key.sig="2f"/>
            </staffGrp>
         </staffGrp>
      </scoreDef>
...
```

4. Tempo instructions

You can encode the tempo instructions using the **<tempo>** element. **<tempo>** elements should be placed inside the **<measure>**, following the encoding of the staff content. Details of the placement of the tempo marking may be captured using **@tstamp**, **@staff** and **@place** attributes. The quarter note symbol that occurs within the tempo marking can be represented by the Unicode character reference ♩ (U+2669, see https://www.fileformat.info/info/unicode/char/2669/index.htm for more information).

```
...
   </layer>
   <tempo place="above" staff="1" mm="152">Maelzel's metronom
♩=152.   <lb/> Schnell.</tempo>
</staff>
...
```

You will also note that we encoded the line break in the tempo marking by using the **<lb>** element, which is the same in MEI as in TEI.

5. Dynamics

In the next step, you will learn how to encode some dynamic symbols, such as the one at the beginning of "Erlkönig". Dynamics are encoded using the **<dynam>** element. Like the **<tempo>** element, **<dynam>** should be placed within the measure, following the **<staff>** elements. Since each one makes reference to its related staff, the order in which the **<dynam>** and **<tempo>** elements occur is not important.

```
...
   </layer>
   <tempo place="above" staff="1" mm="152">Maelzel's metronom
♩=152. <lb/> Schnell.</tempo>
   <dynam tstamp="1" place="below" staff="2">f</dynam>
</staff>
...
```

6. Note/Chord repetition

Congratulations! You've encoded the first measure of the "Erlkönig." Are you ready for the second measure?

In the next measure of the "Erlkönig", the tuplets in the right hand of the piano are replaced by shorthand notation that indicates repetition of a chord, called "bowed tremolo" in MEI.

MEI provides the **<bTrem>** element for this kind of repetition. Check chapter 4.2.5.3 of the **Guidelines** for information about its use.

```
...
<measure n="2">
```

44

```xml
        <staff n="2">
          <layer>
            <bTrem num="3" num.visible="false" xml:id="btrem1">
              <chord stem.dir="up" stem.mod="1slash">
                <note pname="g" oct="4" dur="8" dots="1"/>
                <note pname="g" oct="3" dur="8" dots="1"/>
              </chord>
            </bTrem>
            <bTrem copyof="#bTrem1"/>
            <bTrem copyof="#bTrem1"/>
            <bTrem copyof="#bTrem1"/>
          </layer>
        </staff>
        <staff n="3">
          <layer>
            <tuplet bracket.visible="false" num="3"
num.visible="true" num.place="below" xml:id="tuplet2">
              <beam>
                <note pname="g" oct="2" dur="8" stem.dir="up"
xmi:id="sli"/>
                <note pname="a" oct="2" dur="8" stem.dir="up"/>
                <note pname="b" oct="2" dur="8" stem.dir="up"/>
              </beam>
            </tuplet>
            <tuplet bracket.visible="false" num="3"
num.visible="true" num.place="below" xml:id="tuplet3">
              <beam>
                <note pname="c" oct="3" dur="8" stem.dir="up"/>
                <note pname="d" oct="3" dur="8" stem.dir="up"/>
                <note pname="e" oct="3" dur="8" stem.dir="up"
xml:id="slt"/>
              </beam>
            </tuplet>
            <note pname="d" oct="3" dur="4" stem.dir="up"
artic="stacc"/>
            <note pname="d" oct="3" dur="4" stem.dir="up"
artic="stacc"/>
          </layer>
        </staff>
```

```
    <dynam tstamp="1" place="below" staff="3">f</dynam>
    <slur startid="s1i" endid="s1t" place="below" staff="2"/>
</measure>
...
```

Exploring the MEI Header - Part 1

The following sections on the MEI header aim to create metadata for an electronic transcription of Robert Schumann's song "Der Abendstern". If you were cataloging this work for a library database, you would capture the relevant bibliographic information for the work as metadata stored within the appropriate fields in the library database. Compiling metadata using MEI is not essentially different from that procedure. The MEI header also provides various places (called "fields" when working in a database, but "elements" in XML) where you can insert bibliographic information.

Working with the MEI header, you will learn that elements with the same name can be used in many different places, but that their precise meaning can change, depending on the context in which they are used. A **<title>** of a **<work>** is different than a **<title>** of a file. Therefore, you must always be aware of the current context when working on the header.

1. New MEI File

We'll explore the MEI header step by step, starting with the description of the electronic file itself. You learned how to create a new MEI document within the tutorial Creating a new MEI document. Take the file you created there or start a new one (ignoring the **<music>** part for the moment). The file should be completely empty but valid.

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-model href="mei-cmn.rng" type="application/xml"
schematypens="http://relaxng.org/ns/structure/1.0"?>
<mei xmlns="http://www.music-encoding.org/ns/mei">
    <meiHead>
    <fileDesc>
        <titleStmt>
```

```
            <title/>
        </titleStmt>
        <pubStmt/>
    </fileDesc>
    </meiHead>
    <music/>
 </mei>
```

2. File Description

The first section of the header is the **<fileDesc>** element. As its name implies, this element contains a description of the file you're creating, not to be confused with the musical work that is the content of the file.

Obviously, it would be good to give the file a title and to indicate who is responsible for creating it. In addition, information about the ownership, publication, and distribution of the file would be helpful. In fact, these basic descriptive components are required by MEI. Looking at the structure of the empty MEI file just created, you'll notice a couple of mandatory elements. The **<meiHead>** element always requires a **<titleStmt>** (title statement) and a **<pubStmt>** (publication statement).

```
<meiHead>
    <fileDesc>
    <titleStmt>
        <title>Der Abendstern</title>
    </titleStmt>
    <pubStmt/>
    </fileDesc>
</meiHead>
```

3. File Title

Let's have a detailed look at the **<titleStmt>** element of your new MEI header. It contains a **<title>** element, which should be filled out with the title given to the electronic work by its creator. Most often, this is the same title as the source of the encoding. So start the encoding of your MEI header by choosing an appropriate title.

When creating an electronic transcription of an existing source, it is highly recommended that the title be derived from the source while at the same time being clearly distinguishable from it. You might add a simple phrase, such as "an electronic transcription" or "a digital edition", to distinguish the electronic work from the source in citations and in catalogs containing descriptions of both types of material.

For further information about the file description, please refer to the **MEI Guidelines.**

Now, add the distinguishing phrase of your choice to the title in the file description.

```
<meiHead>
    <fileDesc>
    <titleStmt>
        <title>Der Abendstern: an electronic
transcription.</title>
    </titleStmt>
    <pubStmt/>
    </fileDesc>
</meiHead>
```

4. Responsibility Statement

The responsibility statement element **\<respStmt\>** is included within the title statement. This is the element within your file description where you'll encode one or more persons responsible for the creation of the electronic work (for example the encoder, editor, author or compiler) as well as persons who have been responsible for the creation of the existing source text, the composer(s).

The **\<respStmt\>** is optional. If used, it may contain a **\<resp\>** element (holding a phrase describing the nature of a person's intellectual responsibility). It may also have **\<name\>**, **\<corpName\>**, and **\<persName\>** elements indicating the responsible person or group.

Please add a **\<respStmt\>** element to your MEI header. Include information about the composer of the original work (Robert Schumann) and indicate who is responsible for the machine-readable transcription. (This should be you, as you are creating this file at the moment).

```
<meiHead>
    <fileDesc>
    <titleStmt>
        <title>Der Abendstern: an electronic
transcription</title>
        <respStmt>
            <resp>Composed by:</resp>
            <persName>Robert Schumann</persName>
            <resp>Machine-readable transcription by:</resp>
            <persName>John Doe</persName>
        </respStmt>
    </titleStmt>
    <pubStmt/>
```

```
        </fileDesc>
</meiHead>
```

5. Attributes @role, @dbkey, @authority, and @authURI

Because the order of elements in the MEI header is not predetermined, it can be somewhat complicated to programmatically extract the correct combinations of **<resp>** and **<persName>** elements. To avoid this problem, we recommend that you use the **@role** attribute on **<persName>** to specify a person's responsibility. The controlled vocabulary from the [MARC code list for relators](#) is a good source for values for this attribute. Using a controlled vocabulary makes your files easier to understand and thus facilitates data interchange.

It's also highly recommended that you standardize the form of names in the various responsibility statements. For example, you could take the normalized form of a person's name from an authority file. The **@authority** and **@authURI** attributes can be used to indicate the name and URI of the authority file you've consulted and the **@dbkey** attribute can be used to link to the record number for the individual name within the authority database.
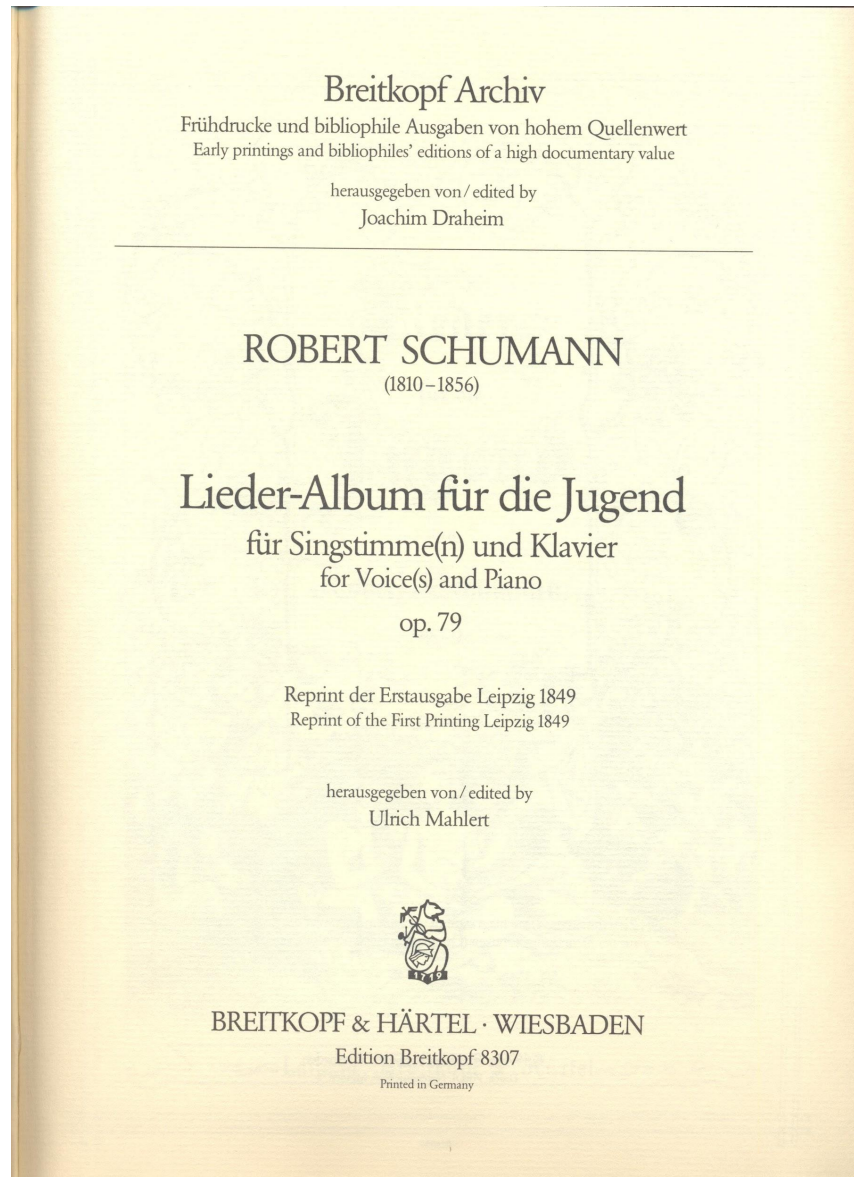
```
<meiHead>
    <fileDesc>
        <titleStmt>
            <title>Der Abendstern: an electronic
transcription</title>
            <respStmt>
                <resp>Composed by:</resp>
                <persName role="composer" dbkey="118611666"
authority="GND" authURI="http://d-nb.info/gnd">Robert
Schumann</persName>
                <resp>Machine-readable transcription by:</resp>
                <persName role="encoder">John Doe</persName>
            </respStmt>
        </titleStmt>
        <pubStmt/>
    </fileDesc>
</meiHead>
```

6. Publication Statement

Let's go on to the second mandatory element of the file description: the publication statement.

The **<pubStmt>** element is a container for information regarding the publication or distribution of a bibliographic item. It includes the publisher's name and address, the date of publication, and other relevant details. It may also contain a single **<unpub>** element, if the file has yet to be published.

As your file has not been published yet, please add the **<unpub>** child element to your file description. You might also add a short phrase indicating the nature of or the reason for the unpublished status as the element's content, but this is optional.

```
<meiHead>
    <fileDesc>
    <titleStmt>
        <title>Der Abendstern: an electronic
transcription</title>
        <respStmt>
            <resp>Composed by:</resp>
            <persName role="composer" dbkey="118611666"
authority="GND" authURI="http://d-nb.info/gnd">Robert
Schumann</persName>
            <resp>Machine-readable transcription by:</resp>
            <persName role="encoder">John Doe</persName>
        </respStmt>
    </titleStmt>
    <pubStmt>
        <unpub>Incomplete!</unpub>
    </pubStmt>
    </fileDesc>
</meiHead>
```

7. Revision Description

For the last step in this tutorial let's have a short look at another section of the MEI header: the revision description, which is represented by the **<revisionDesc>** element. The revision description is always encoded at the end of the MEI header. It may contain only **<change>** child elements.

Although it is not mandatory, it is a good practice to provide a **<revisionDesc>**. Its purpose is to record modifications to the file. Each modification is described in a separate **<change>** element, which may hold information about the modification date, affected content and responsibility. Major revisions of the header itself should also be documented in the **<revisionDesc>.**

What you could do right now is to create the beginning of a list of **<change>** elements. So fill in the **<revisionDesc>** element, along with the sub-element **<change>** and its sub-elements **<respStmt>**, **<changeDesc>** and **<date>**. Add an **@n** attribute to the **<change>** element and give it a value of "1", since this is the first change made to the MEI file. Please also enter the name of the person responsible for the change (this would be you again), the date of change in ISO format (YYYY-MM-DD) and a description of the changes you made.

```
<meiHead>
     <fileDesc>
     <titleStmt>
          <title>Der Abendstern: an electronic
transcription</title>
          <respStmt>
               <resp>Composed by:</resp>
               <persName role="composer" dbkey="118611666"
authority="GND" authURI="http://d-nb.info/gnd">Robert
Schumann</persName>
               <resp>Machine-readable transcription by:</resp>
               <persName role="encoder">John Doe</persName>
          </respStmt>
     </titleStmt>
     <pubStmt>
          <unpub>Incomplete!</unpub>
```

```xml
    </pubStmt>
    </fileDesc>
    <revisionDesc>
      <change n="1">
        <respStmt>
            <persName>[Name of the person responsible for the
change]</persName>
        </respStmt>
        <changeDesc>
            <p>[Description of the change made]</p>
        </changeDesc>
        <date isodate="YYYY-MM-DD"/>
      </change>
    </revisionDesc>
</meiHead>
```

Exploring the MEI Header - Part 2

Now that we've covered how to encode a minimal header, we'll proceed to encoding some additional details for Robert Schumann's song "Der Abendstern". We'll use the following sources:

- the reprint of the first Leipzig edition 1849, published by Breitkopf & Härtel:

- and the entry describing this manifestation in the catalog of the German National Library:

Furthermore, you will learn how to indicate the relationship between this song and the song cycle of which it is a part.

Please use the file from Part 1 as the basis of your further encodings.

1. Source Description (sourceDesc)

The file description is the place to encode the sources used to create the electronic file, so insert a **<sourceDesc>** element into your file description. As the reprint of the first print by Breitkopf & Härtel (Leipzig 1849) is used as the exemplar in this case, you should encode its metadata there.

```
<meiHead>
    <fileDesc>
        <titleStmt>
            <title>Der Abendstern: an electronic
transcription</title>
            <respStmt>
                <resp>Composed by:</resp>
```

```
                <persName role="composer" dbkey="118611666"
authority="GND" authURI="http://d-nb.info/gnd">Robert
Schumann</persName>
                <resp>Machine-readable transcription by:</resp>
                <persName role="encoder">John Doe</persName>
            </respStmt>
        </titleStmt>
        <pubStmt>
            <unpub>Incomplete!</unpub>
        </pubStmt>
        <sourceDesc>
            <source></source>
        </sourceDesc>
    </fileDesc>
    ...
</meiHead>
```

2. Work description (workDesc)

As a next step, fill in the element for the work description. The work description is represented by the **<workDesc>** element. It follows the file description.

```
<meiHead>
    <fileDesc>
        ...
    </fileDesc>
    <workDesc>
        <work></work>
    </workDesc>
    ...
</meiHead>
```

3. Title statements of the source description and the work description

In Part 1 you learned that the file description includes a title statement that contains the title and the persons responsible for the electronic file you're creating. The **<titleStmt>** element is one of the elements that can occur in different places within

a single MEI file. The source description and the work description have their own title statements. So you need to decide where to encode each one.

```
<meiHead>
    <fileDesc>
        ...
        <sourceDesc>
            <source>
                <titleStmt>
                    <title>Lieder-Album für die Jugend für
Singstimme(n) und Klavier ; op. 79</title>
                </titleStmt>
            </source>
        </sourceDesc>
    </fileDesc>
    <workDesc>
        <work>
            <titleStmt>
                <title>Der Abendstern</title>
            </titleStmt>
        </work>
    </workDesc>
    ...
</meiHead>
```

4. Uniform titles

If you'd like to encode any uniform titles for the „Liederalbum" and the song "Der Abendstern", you may add another **<title>** element and a **@type** attribute with the value of "uniform".

```
<meiHead>
    <fileDesc>
        ...
        <sourceDesc>
            <source>
                <titleStmt>
```

```
                    <title>Lieder-Album für die Jugend für
Singstimme(n) und Klavier ; op. 79</title>
                    <title type="uniform">Lieder für die
Jugend</title>
                </titleStmt>
            </source>
        </sourceDesc>
    </fileDesc>
    <workDesc>
        <work>
          <titleStmt>
                <title>Der Abendstern</title>
                <title type="uniform">Lieder für die Jugend
&lt;Der Abendstern&gt;</title>
          </titleStmt>
        </work>
        </workDesc>
      ...
</meiHead>
```

5. Persons responsible in sourceDesc and workDesc

Now encode the persons responsible for the source(s) used for the electronic transcription by adding a responsibility statement (**\<respStmt\>**) to the **\<titleStmt\>** within the **\<sourceDesc\>** element. Please include role information for everyone involved in the creation of the work.

Do the same for the work description (**\<workDesc\>**), adding the persons responsible for the song "Der Abendstern", the first song of the song cycle.

```
<meiHead>
    <fileDesc>
        ...
        <sourceDesc>
            <source>
                <titleStmt>
```

```
                    <title>Lieder-Album für die Jugend für
Singstimme(n) und Klavier ; op. 79</title>
                    <title type="uniform">Lieder für die
Jugend</title>
                    <respStmt>
                        <persName role="composer">Robert
Schumann</persName>

                        <resp>Herausgegeben von:</resp>
                        <persName role="editor">Ulrich
Mahlert</persName>
                    </respStmt>
                </titleStmt>
            </source>
        </sourceDesc>
    </fileDesc>
    <workDesc>
        <work>
            <titleStmt>
                <title>Der Abendstern</title>
                <title type="uniform">Lieder für die Jugend
&lt;Der Abendstern&gt;</title>
                <respStmt>
                    <persName role="composer">Robert
Schumann</persName>
                    <persName role="lyricist">August Heinrich
Hoffmann von Fallersleben</persName>
                </respStmt>
            </titleStmt>
        </work>
    </workDesc>
    ...
</meiHead>
```

6. @dbkey, @authority, @authURI, @role

We hope you're following the recommendations to use standardized forms of any names for the responsible parties. If so, add information about any authority file or

controlled vocabulary which you've consulted or which provides further information.

```xml
<meiHead>
    <fileDesc>
        ...
        <sourceDesc>
            <source>
                <titleStmt>
                    <title>Lieder-Album für die Jugend für
Singstimme(n) und Klavier ; op. 79</title>
                    <title type="uniform">Lieder für die
Jugend</title>
                    <respStmt>
                        <persName role="composer"
authority="GND" authURI="http://d-nb.info/gnd"
dbkey="118611666">Robert Schumann</persName>
                        <resp>Herausgegeben von:</resp>
                        <persName role="editor"
authURI="http://d-nb.info/gnd" authority="GND"
dbkey="10400097X">Ulrich Mahlert</persName>
                    </respStmt>
                </titleStmt>
            </source>
        </sourceDesc>
    </fileDesc>
    <workDesc>
        <work>
            <titleStmt>
                <title>Der Abendstern</title>
                <title type="uniform">Lieder für die Jugend
&lt;Der Abendstern&gt;</title>
                <respStmt>
                    <persName role="composer" authority="GND"
authURI="http://d-nb.info/gnd" dbkey="118611666">Robert
Schumann</persName>
                    <persName role="lyricist" authority="GND"
authURI="http://d-nb.info/gnd" dbkey="118552589">August Heinrich
Hoffmann von Fallersleben</persName>
```

```
                    </respStmt>
              </titleStmt>
          </work>
     </workDesc>
     ...
</meiHead>
```

7. Information on publication of the source

The next exercise deals with the edition statement (**<editionStmt>**) and the
publication statement (**<pubStmt>**) of the source description (**<sourceDesc>**). This
exercise encodes information about the publication of the reprint that can be found
in the catalog entry illustrated above.

1. Enter an edition statement (**<editionStmt>**) after the **<titleStmt>** element.
2. Enter a publication statement (**<pubStmt>**) after the **<editionStmt>**
   element.

```
<meiHead>
    <fileDesc>
        ...
       <sourceDesc>
            <source>
                <titleStmt>
                    ...
                </titleStmt>
                <editionStmt>
                    <edition></edition>
                </editionStmt>
                <pubStmt></pubStmt>
            </source>
       </sourceDesc>
    </fileDesc>
    ...
</meiHead>
```

8. Contents of the edition statement of the source description

The edition statement (**<editionStmt>**) is a container for metadata pertaining to a particular edition of the material being described. Use the **<edition>** child element for capturing information about the reprint of the first 1849 printing from Leipzig. The **<edition>** element contains a word or text phrase that indicates a difference in either content or form between the item being described and a related item previously issued by the same publisher.

The title page of the reprint reads:
Reprint der Erstausgabe Leipzig 1849
Reprint of the First Printing Leipzig 1849

```
<meiHead>
    <fileDesc>
        ...
        <sourceDesc>
            <source>
                <titleStmt>
                 ...
                </titleStmt>
                <editionStmt>
                    <edition>Reprint der Erstausgabe Leipzig
1849</edition>
                </editionStmt>
                <pubStmt></pubStmt>
            </source>
        </sourceDesc>
    </fileDesc>
    ...
</meiHead>
```

9. Contents of the publication statement of the source description

The title pages also display other information regarding the publication of the reprint. Have a look at the scans below:

Breitkopf Archiv

Frühdrucke und bibliophile Ausgaben von hohem Quellenwert
Early printings and bibliophiles' editions of a high documentary value

herausgegeben von / edited by
Joachim Draheim

ROBERT SCHUMANN
(1810–1856)

Lieder-Album für die Jugend
für Singstimme(n) und Klavier
for Voice(s) and Piano
op. 79

Reprint der Erstausgabe Leipzig 1849
Reprint of the First Printing Leipzig 1849

herausgegeben von / edited by
Ulrich Mahlert

BREITKOPF & HÄRTEL · WIESBADEN
Edition Breitkopf 8307
Printed in Germany

To encode information about publisher, the printer, the publisher's address, the order number and the copyright notice, you can use several special child elements within the publication statement (**<pubStmt>**) of the source description:

1. Please add a **<respStmt>** element to the **<pubStmt>** element
2. Now add a **<corpName>** element to the **<respStmt>** and enter the name of the publisher of the reprint.
3. The sample reads: "Druck: PIROL Notendruckerei, Minden". Add a **<resp>** element and a second **<corpName>** element to the **<respStmt>** to encode this information on the printer.
4. You can also encode the address of the publisher, which is given on the title page of the reprint. You may use the **<address>** element, which requires one or more **<addrLine>** child elements to specify the place of publication.
5. For encoding the order number of the reprint, please add an **<identifier>** element to the publication statement. The **@type** attribute can be used to specify which kind of identifier you're capturing.

There is also a copyright notice within the reprint. For encoding this, please add an **<availability>** element as well as an **<useRestrict>** child element to the publication statement of the source description.

```
<meiHead>
    <fileDesc>
        ...
        <sourceDesc>
            <source>
                <titleStmt>
                    ...
                </titleStmt>
                <editionStmt>
                    ...
                </editionStmt>
                <pubStmt>
                    <respStmt>
                        <corpName authority="GND"
authURI="http://d-nb.info/gnd" dbkey="330001418">Breitkopf und
Härtel</corpName>
                        <resp>Druck:</resp>
```

```
                        <corpName>PIROL Notendruckerei,
Minden</corpName>
                    </respStmt>
                    <address>
                        <addrLine>Wiesbaden</addrLine>
                        <addrLine>Germany</addrLine>
                    </address>
                    <identifier type="ordernumber">Breitkopf &
Härtel 8307</identifier>
                    <availability>
                        <useRestrict>Copyright 1991 by Breitkopf
& Härtel, Wiesbaden</useRestrict>
                    </availability>
                </pubStmt>
            </source>
        </sourceDesc>
    </fileDesc>
    ...
</meiHead>
```

10. Series statement of the source description

As shown on the title page, the "Lieder-Album für die Jugend" is published within the series "Breitkopf Archiv", which is edited by Joachim Draheim.

You can capture this information in the series statement by adding a **<seriesStmt>** after the publication statement (**<pubStmt>**) of the source description.

```
<meiHead>
    <fileDesc>
        ...
        <sourceDesc>
            <source>
                ...
                <pubStmt>
                    ...
                </pubStmt>
                <seriesStmt>
```

```xml
                    <title>Breitkopf-Archiv</title>
                    <respStmt>
                        <resp>Herausgegeben von:</resp>
                        <persName role="editor"
authURI="http://d-nb.info/gnd" authority="GND"
dbkey="123921228">Joachim Draheim</persName>
                    </respStmt>
                </seriesStmt>
            </source>
        </sourceDesc>
    </fileDesc>
    ...
</meiHead>
```

11. Physical description of the source description

The catalog entry of the German National Library gives some information on physical descriptions of the reprint:

Create a physical description (**<physDesc>** element) within the source description and enter the child elements **<dimensions>**, which encodes information about the

physical size of a bibliographic source, and **<extent>**, which is used to express size in terms other than physical dimensions, such as number of pages etc.

To encode the plate number of the reprint, found at the foot of page 3 of the source, enter a **<plateNum>** child element to the physical description.

```
<meiHead>
    <fileDesc>
        ...
        <sourceDesc>
            <source>
                ...
                <pubStmt>
                    ...
                </pubStmt>
                <physDesc>
                    <dimensions unit="cm">30 cm</dimensions>
                    <extent unit="page">57 S.</extent>
                    <plateNum>8062</plateNum>
                </physDesc>
                ...
            </source>
        </sourceDesc>
    </fileDesc>
    ...
</meiHead>
```

12. Classification of the source description

Also within the source description, the **<classification>** element can be used to classify a musical text according to one or more of the following methods:

1. by reference to a recognized international classification scheme such as the DDC, the Universal Decimal Classification, the Library of Congress Classification, or any other system widely used in library and documentation work;
2. by providing a set of keywords, as provided, for example, by British Library or Library of Congress Cataloguing in Publication data.

The following elements are provided for this purpose: **<termList>** and **<classCode>**.

The **<termList>** element categorizes an individual text by supplying a set of terms that may describe its topic or subject matter, its physical or intellectual form, date, etc. Each term is represented by a **<term>** element. In some schemes, the order of items in the list is significant, for example, from major topic to minor; in others, the list has an organized substructure of its own. Wherever possible, such terms should be taken from a recognized source.

Try to classify the song cycle and enter the data in the **<classification>** element.

```
<meiHead>
    <fileDesc>
        ...
        <sourceDesc>
            <source>
                ...
                <seriesStmt>
                    ...
                 </seriesStmt>
                <classification>
                    <termList>
                        <term classcode="#DNB_Sachgruppen">02a
Werke für Einzelstimmen</term>
                    </termList>
                    <classCode>DNB_Sachgruppen</classcode>
                </classification>
            </source>
        </sourceDesc>
    </fileDesc>
    ...
</meiHead>
```

Well done! Within this tutorial you learned to encode all required metadata for identifying the source of your electronic transcription. You also learned how to incorporate a work description in you header.

Exploring the MEI Header - Part 3

Although you created quite an extensive header in Part 2, we can step on and enhance your encoding a little bit more. Level 3 will deal with more special features of the MEI header. You will learn how to transcribe a title page and the table of contents of a source, how to encode non-bibliographic aspects of a text within the work description, how to link a facsimile or information on the performance of a work.

Please take the file you created in parts 1 and 2 as a basis for your further encodings.

1. Contents of a source

It is often useful to give information about works contained in a source. The reprint of the "Liederalbum" contains 29 songs. Each of these songs could be encoded with a separate <work> element within the work description.

INHALT.

As a first exercise, please search an appropriate element within the source description where you could encode such information. Explore the given child elements and try to encode all the information on the page with the table of contents and including the headline.

```
<meiHead>
    <fileDesc>
        ...
        <sourceDesc>
            <source>
                ...
```

```xml
        <classification>
          ...
         </classification>
        <contents>
            <head>INHALT.</head>
            <label>Kinderlieder von Hoffmann von
Fallersleben.</label>
            <contentItem n="1">1. Der
Abendstern.</contentItem>
            <contentItem n="2">2.
Schmetterling.</contentItem>
            <contentItem n="3">3.
Frühlingsbotschaft.</contentItem>
            <contentItem n="4">4.
Frühlingsgruss.</contentItem>
            <contentItem n="5">5. Vom
Schlaraffenland.</contentItem>
            <contentItem n="6">6. Sonntag.</contentItem>
            <contentItem n="7">Zwei Zigeunerliedchen,
aus dem Spanischen von E. Geibel.</contentItem>
            <contentItem n="8">Des Knaben Berglied, v.
L. Uhland.</contentItem>
            <contentItem n="9">Mailied.</contentItem>
            <contentItem n="10">Käuzlein, aus des Knaben
Wunderhorn.</contentItem>
            <contentItem n="11">Hinaus in's Freie, von
Hoffmann von Fallersleben.</contentItem>
            <contentItem n="12">Der Sandmann, von H.
Kletke.</contentItem>
            <contentItem n="13">Marienwürmchen, aus des
Knaben Wunderhorn.</contentItem>
              ...
          </contents>
       </source>
     </sourceDesc>
   </fileDesc>
   ...
</meiHead>
```

2. titlePage element of the source description

Within the source description you can encode further details on the source of your electronic transcription. In addition to information on the provenance of a certain source, you could also add special information like inscriptions or a hand list to the physical description. You can also encode detailed information on the title page within the physical description. In the case of the reprint you could encode information on even two title pages, the title page of the reprint and the included original title page of the first print.
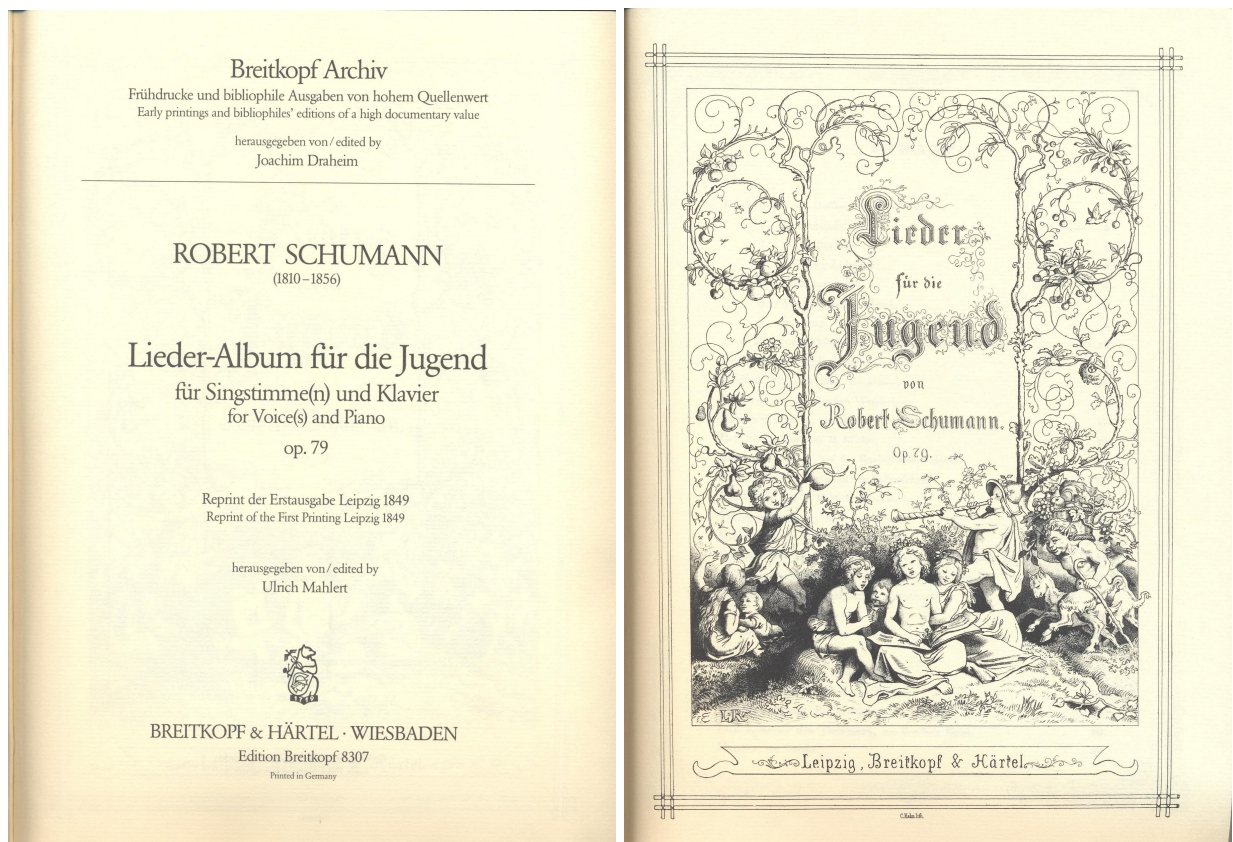
Please add a **\<titlePage\>** element to the physical description of the source, explore the allowed child elements and encode the information given in the scans above.

```
<meiHead>
    <fileDesc>
        ...
        <sourceDesc>
            <source>
                ...
```

```xml
            <pubStmt>
                ...
            </pubStmt>
            <physDesc>
                <dimensions>30 cm</dimensions>
                <extent>57 S.</extent>
                <plateNum>8062</plateNum>
                <titlePage n="1">
                        <p>Breitkopf Archiv<lb/>Frühdrucke und
bibliophile Ausgaben von hohem Quellenwert<lb/>Early printings
and bibliophiles' editions of a high documentary value</p>
                                <p>herausgegeben von / edited
by<lb/>Joachim Draheim</p>
                        <p>Robert Schumann<lb/>(1810−1856)</p>
                        <p>Lieder-Album für die Jugend<lb/>für
Singstimme(n) und Klavier<lb/>for Voice(s) and Piano<lb/>op.
79</p>
                                <p>Reprint der Erstausgabe Leipzig
1849<lb/>Reprint of the First Printing Leipzig 1849</p>
                                <p>herausgegeben von / edited
by<lb/>Ulrich Mahlert</p>
                                    <p>Breitkopf & Härtel,
Wiesbaden<lb/>Edition Breitkopf 8307<lb/>Printed in Germany</p>
                </titlePage>
                <titlePage n="2">
                                    <p>Lieder<lb/>für
die<lb/>Jugend<lb/>Robert Schumann<lb/>op. 79</p>
                        <p>Leipzig, Breitkopf & Härtel.</p>
                        <p>C.Hahn.lith.</p>
                </titlePage>
            </physDesc>
            ...
        </source>
    </sourceDesc>
  </fileDesc>
  ...
</meiHead>
```

3. Encoding of key, tempo and meter elements within the work description

Now let's have a detailed look at the work description. In the work description, multiple **<work>** elements could be encoded. At the moment we just need one and you already encoded the title of it in the previous tutorial. This **<work>** element contains non-bibliographic information on the song "Der Abendstern".

Information about key, tempo and meter could be encoded here. Using the **<key>** element you can describe the key, the **<tempo>** element is used to encode information about the tempo indication, and the **<meter>** element captures information on the time signature.

Add the three elements to the work description of your MEI file and fill them.

```
<meiHead>
  <fileDesc>
    …
  </fileDesc>
  <workDesc>
    <work>
      <titleStmt>
        …
      </titleStmt>
      <key pname="a" mode="major">A major</key>
      <tempo>Langsam.</tempo>
      <meter count="2" unit="4"/>
    </work>
  </workDesc>
  …
</meiHead>
```

4. Performance medium of the work description

The next element you should pay attention to is **<perfMedium>**. The performance medium indicates the number and character of the performing forces used in a musical composition.

Please add a **<perfMedium>** element to the work description and encode the instrumental parts of the song "Der Abendstern".

```
<meiHead>
  <fileDesc>
    …
  </fileDesc>
  <workDesc>
    <work>
      <titleStmt>
        …
      </titleStmt>
      <key pname="a" mode="major">A major</key>
      <tempo>Langsam.</tempo>
      <meter count="2" unit="4"/>
       <perfMedium>
         <instrumentation>
           <instrVoice>Singstimme.</instrVoice>
           <instrVoice>Pianoforte.</instrVoice>
         </instrumentation>
       </perfMedium>
    </work>
  </workDesc>
  …
</meiHead>
```

5. Encoding of incipit information - I

Within a single **<work>** element you can encode information on the incipit of a musical composition. The beginning of a piece of music is often used for identification purposes, especially when the piece has only a generic title, such as „Sonata no. 3". They appear in catalogs of music and in tables of contents of printed music that include multiple works. The following elements are provided for the inclusion of incipits: **<incip>**, **<incipCode>**, **<incipText>**, **<score>** and **<graphic>**. The elements **<incipCode>** and **<incipText>** are available for the inclusion of coded incipits and textual incipits, respectively. The **<incipText>** element should contain only the initial performed text of the work, while **<incipCode>** may contain both words and music, depending on the capabilities of the scheme used to encode it. When both music and text are provided in **<incipCode>**, it may be helpful to repeat the text in **<incipText>** in order to provide easier access to only the text, for example, for indexing of the text without having to extract it from the coded incipit.

Both **<incipCode>** and **<incipText>** allow reference to an external file location via the target attribute and specification of the internet media type of external file via the mimetype attribute.

Begin with the incipit text. Enter an **<incip>** element to your work description and encode the opening words of the song.

```
<meiHead>
  <fileDesc>
    …
  </fileDesc>
  <workDesc>
    <work>
      <titleStmt>
        …
      </titleStmt>
      <key pname="a" mode="major">A major</key>
      <tempo>Langsam.</tempo>
      <meter count="2" unit="4"/>
      <incip>
```

```
        <incipText>
          <p>Du lieblicher Stern,</p>
        </incipText>
      </incip>

      …
    </work>
  </workDesc>

  …
</meiHead>
```

6. Encoding of incipit information - II

Now encode the complete incipit of the song "Der Abendstern". Use the **<score>** element as a starting point. The encoding doesn't differ from music being encoded within the **<music>** element. If you don't know how to encode music yet, please have a look at the tutorials on the music first.

```
<incip>
    <incipText>
    <p>Du lieblicher Stern</p>
    </incipText>
    <score>
    <scoreDef>
        <staffGrp>
            <staffDef n="1" label="Singstimme" lines="5"
meter.count="2" meter.unit="4" clef.shape="G" clef.line="2"
key.sig="3s"/>
```

```xml
<staffGrp label="Pianoforte" symbol="brace">
    <staffDef n="2" lines="5" meter.count="2"
meter.unit="4" clef.shape="G" clef.line="2" key.sig="3s"/>
    <staffDef n="3" lines="5" meter.count="2"
meter.unit="4" clef.shape="F" clef.line="4" key.sig="3s"/>
</staffGrp>
</staffGrp>
</scoreDef>
<section>
    <measure n="0" metcon="false">
        <staff n="1">
            <layer>
                <note pname="a" oct="4" dur="8">
                <verse n="1">
                    <syl>V.1. Du</syl>
                </verse>
                <verse n="2">
                    <syl>V.2. Wie</syl>
                </verse>
                <verse n="3">
                    <syl>V.3. So</syl>
                </verse>
                <verse n="4">
                    <syl>V.4. Wie</syl>
                </verse>
                </note>
            </layer>
        </staff>
        <staff n="2">
            <layer>
                <note pname="a" oct="4" dur="8">
            </layer>
        </staff>
        <staff n="3">
            <layer>
                <note pname="a" oct="3" dur="8">
            </layer>
        </staff>
```

```xml
                <slur tstamp="0.5" curvedir="below" dur="1m+2.5"
staff="2"/>
                <dynam place="below" tstamp="0.5"
staff="1">p</dynam>
                <dir place="above" staff="1" layer="1"
tstamp="0">Langsam.</dir>
                <dynam place="below" tstamp="0.5"
staff="2">p</dynam>
            </measure>
            <measure n="1">
                <staff n="1">
                    <layer>
                        <note pname="c" oct="5" dur="4">
                        <verse n="1">
                            <syl con="d">lieb</syl>
                        </verse>
                        <verse n="2">
                            <syl>lieb'</syl>
                        </verse>
                        <verse n="3">
                            <syl>blick'</syl>
                        </verse>
                        <verse n="4">
                            <syl>nickst</syl>
                        </verse>
                        </note>
                        <note pname="a" oct="4" dur="8">
                        <verse n="1">
                            <syl>cher</syl>
                        </verse>
                        <verse n="2">
                            <syl>doch</syl>
                        </verse>
                        <verse n="3">
                            <syl>nach</syl>
                        </verse>
                        <verse n="4">
                            <syl>mir</syl>
                        </verse>
```

```xml
            </note>
        </layer>
    </staff>
    <staff n="2">
        <layer>
            <chord dur="4">
            <note pname="c" oct="4"/>
            <note pname="e" oct="4"/>
            <note pname="a" oct="4"/>
            <note pname="c" oct="5"/>
            </chord>
            <beam>
            <chord dur="8">
                <note pname="d" oct="4"/>
                <note pname="e" oct="4"/>
                <note pname="g" oct="4"/>
                <note pname="b" oct="4"/>
            </chord>
            <chord dur="8">
                <note pname="e" oct="4"/>
                <note pname="a" oct="4"/>
            </chord>
            </beam>
        </layer>
    </staff>
    <staff n="3">
        <layer>
            <chord dur="4">
            <note pname="a" oct="2"/>
            <note pname="e" oct="3"/>
            <note pname="a" oct="3"/>
            </chord>
            <beam>
            <chord dur="8">
                <note pname="b" oct="2"/>
                <note pname="e" oct="3"/>
            </chord>
            <chord dur="8">
                <note pname="c" oct="3"/>
```

```xml
                                    <note pname="e" oct="3"/>
                                    <note pname="a" oct="3"/>
                            </chord>
                            </beam>
                    </layer>
            </staff>
        </measure>
        <measure n="2">
            <staff n="1">
                <layer>
                    <note pname="f" oct="4" dur="4"
dots="1">
                        <verse n="1">
                            <syl>Stern,</syl>
                        </verse>
                        <verse n="2">
                            <syl>dich</syl>
                        </verse>
                        <verse n="3">
                            <syl>dir,</syl>
                        </verse>
                        <verse n="4">
                            <syl>zu</syl>
                        </verse>
                        </note>
                </layer>
            </staff>
            <staff n="2">
                <layer>
                    <chord dur="2">
                    <note pname="d" oct="4"/>
                    <note pname="f" oct="4"/>
                    </chord>
                </layer>
            </staff>
            <staff n="3">
                <layer>
                    <chord dur="2">
                    <note pname="d" oct="3"/>
```

```
                            <note pname="a" oct="3"/>
                        </chord>
                    </layer>
                </staff>
            </measure>
        </section>
    </score>
</incip>
```

Music21

An extensive users guide for music21 is available for free online.
http://web.mit.edu/music21/doc/usersGuide/index.html

Important Links & Select Existing Projects

MEI Github page, https://github.com/music-encoding
MEI Guidelines, http://music-encoding.org/docs/MEI_Guidelines_v3.0.0.pdf

Online catalogs and aggregators

Doing Reusable Musical Data (DoReMus), http://www.doremus.org/
Digital Image Archive of Medieval Music (DIAMM), https://www.diamm.ac.uk/
Digital Resources for Musicology, http://drm.ccarh.org/
Music Scholarship Online (MuSO), http://muso.arts.gla.ac.uk/
Opera & Ballet Primary Sources (OBPS), https://sites.lib.byu.edu/obps/
Virtual Library Musicology (ViFaMusik), https://www.vifamusik.de/

Digital projects

Beethoven Werkstatt, http://beethovens-werkstatt.de (MEI)

Burns Antiphoner, http://burnsantiphoner.bc.edu/

CHARM, http://www.charm.rhul.ac.uk/index.html

Chopin First Editions Online, http://www.chopinonline.ac.uk/cfeo/

Documenting Teresa Carreno Project, https://documentingcarreno.org/

Du Chemin Lost Voices, http://digitalduchemin.org/ (MEI)

English Broadside Ballad Archive, https://ebba.english.ucsb.edu/

Freischütz Digital, http://freischuetz-digital.de/ (MEI)

John Cage Unbound, http://exhibitions.nypl.org/johncage/

Linked Jazz, https://linkedjazz.org/

Marenzio Online Digital Edition, http://www.marenzio.org/ (MEI)

New York Philharmonic Archive, https://archives.nyphil.org/

Online Chopin Variorum Edition, http://www.chopinonline.ac.uk/ocve/

Schenker Documents Online, http://www.schenkerdocumentsonline.org/index.html

Songs of the Victorians, http://www.songsofthevictorians.com/

Suggested Readings

Readings are attached to this document in the following order.

Burgoyne, John Ashley, Ichiro Fujinaga, and J. Stephen Downie, "Music Information Retrieval." In *A New Companion to Digital Humanities.* Edited by Susan Schreibman, Ray Siemens, and John Unsworth. 2016. 213-228. doi:10.1002/9781118680605.ch15.

Duguid, Timothy. "MuSO: Aggregation and Peer Review in Music." White Paper for the National Endowment for the Humanities. 2016. http://oaktrust.library.tamu.edu/handle/1969.1/157548.

Krabbe, Niels and Axel Teich Geertinger. "MEI (Music Encoding Initiative) as a Basis for Thematic Catalogues: Thoughts, Experiences, and Preliminary Results." *2012 RISM Conference: Music Documentation in Libraries, Scholarship, and Practice.* 2012. http://www.rism.info/fileadmin/content/community-content/events/RISM_Conference_2012/TeichGeertinger_Final.pdf.

Pugin, Laurent. "Interaction with Music Encoding." In *»Ei, dem alten Herrn zoll' ich Achtung gern'«: Festschrift für Joachim Veit zum 60. Geburtstag.* Edited by Kristina Richts and Peter Stadler. 2016. 617-630.

Viglianti, Raffaele. "The Music Addressability API: A draft specification for addressing portions of music notation on the web." *DLfM 2016: Proceedings of the 3rd International workshop on Digital Libraries for Musicology.* New York: Association for Computing Machinery, 2016. 57-60. http://dx.doi.org/10.1145/2970044.2970056.

_____. "Music and Words: Reconciling libretto and score editions in the digital medium." In *»Ei, dem alten Herrn zoll' ich Achtung gern'«: Festschrift für Joachim Veit zum 60. Geburtstag.* Edited by Kristina Richts and Peter Stadler. 2016. 727-746.

Weigl, David M. and Kevin R. Page. "Dynamic Semantic Notation: Jamming Together Music Encoding and Linked Data." *17th International Society for Music Information Retrieval Conference.* New York: International for Music InSociety formation Retrieval Conference, 2016. Late-Breaking Session. <https://wp.nyu.edu/ismir2016/wp-content/uploads/sites/2294/2016/08/weigl-dynamic.pdf>.

Wiering, Frans. "Digital Critical Editions of Music: A Multidimensional Model." In *Modern Methods for Musicology: Prospects, Proposals, and Realities.* Edited by Tim Crawford and Lorna Gibson, 2009. 23-45.