

# »Play it again, Sam« – Levels of Complexity in Encoding Performance Personnel

Benjamin W. Bohl  
Goethe-University Frankfurt,  
Bernd Alois Zimmermann-Complete Edition  
Germany  
[bohl@em.uni-frankfurt.de](mailto:bohl@em.uni-frankfurt.de)

Dennis Ried  
Paderborn University,  
Henze-Digital  
Germany  
[dennis.ried@uni-paderborn.de](mailto:dennis.ried@uni-paderborn.de)

## Abstract

Capturing the personnel needed to perform a musical work in MEI metadata is straightforward with standard ensemble configurations, such as string quartets. In contrast, it can be highly complex for extensive orchestral settings, stage music, or, e.g., twentieth-century 'Neue Musik.' Especially in the latter case, the degree of possible variation is virtually limitless. While MEI 4.0.1 offers places within `<perfMedium>` (descendants of `<castList>`, `<perfResList>`) for capturing such data and provides means for quite complicated data structures through, e.g., nesting or referencing, there is still room for improvement. First of all, data structures should stay as simple and generic as possible. That is to say, that structural modification and a more detailed description of MEI's data model for the benefit of a more concise encoding should be the target, especially when envisioning a more structured encoding of more complex setups. For example, representing dependencies between performers and instruments is extremely limited in the data model for `<perfMedium>` in MEI 4.0.1 (see definition of `<perfMedium>` and also [Gubsch & Ried, 2021](#)).

This poster takes as a starting point issues from two edition projects dealing with music from the twentieth century to illustrate philological intricacies and investigate the possibilities to encode them with MEI 4.0.1.

## Introduction

As easy as it might seem to encode scoring information for works with standardized instrumentation in MEI (as metadata), it becomes more difficult with the expansion of the standard setup of instruments from the 1900s onwards – especially when it comes to prepared instruments. Although one can capture this information in MEI, it is mostly just loosely structured text and thus less machine-readable (basic markup). Our projects focus on encoding structured metadata to support a high level of machine-readability.

Our vision is to facilitate the retrieval of musical works from a catalog according to instrumentation specificities. Envisioned use cases could be the compilation of a list of pieces matching the configuration of an ensemble by a conductor or an analytic approach to a

composer's oeuvre. Therefore, we investigate which parameters to record and how this could be done in a machine-readable form.

## 1 Insight into issues

### 1.1 Alternative instrumentation and fallbacks

In the instrument list to his *Photoptosis. Prélude für großes Orchester* (1968) Bernd Alois Zimmermann requires: "3 OBOES / 1. and 2. also oboe d'amore / 3. also cor anglais" ([Zimmermann, 1970](#)). This general requirement of 3 oboes, and their doublings, can be transcribed in MEI's `<perfResList>` and `<perfRes>` elements ([Listing 1](#)).

```
<perfResList xmlns="http://www.music-encoding.org/ns/mei" meiversion="4.0.1">
  <perfRes>3 OBOES
    <lb/>1. and 2. also oboe d'amore
    <lb/>3. also cor anglais
  </perfRes>
</perfResList>
```

**Listing 1:** XML snippet with basic transcription of an instrumentation list in MEI 4.0.1.

The content model of `<perfRes>` allows text-phrase-level markup (except markup of page beginnings, i.e., `model.textPhraseLike.limited`); this provides an easy and human-readable solution, satisfying all desires concerning the transcription of the source and accompanying editorial markup. According to the MEI Guidelines, one can enrich this markup by introducing nested `<perfRes>`-elements to capture the information on the instrumental doublings in a more structured manner. To ease the identification of the instruments, some standard vocabulary, e.g., the UNIMARC: Medium of performance ([IFLA, n.d.](#)), should be referenced from the `@auth.uri` and `@codedval` attributes ([Listing 2](#)).

```
<perfResList xmlns="http://www.music-encoding.org/ns/mei" meiversion="4.0.1">
  <perfRes count="3" codedval="wob" auth.uri="http://iflstandards.info/ns/unimarc/terms/mop/">3 OBOES
    <lb/><perfRes codedval="woa" auth.uri="mop">1. and 2. also oboe d'amore</perfRes>
    <lb/><perfRes codedval="weh" auth.uri="mop">3. also cor anglais</perfRes>
  </perfRes>
</perfResList>
```

**Listing 2:** XML Snippet with instrument doublings and references to standard vocabulary in MEI 4.0.1. The repeated duplication of `@auth.uri` has been shortened for readability reasons to 'mop', the preferred namespace prefix for `http://iflstandards.info/ns/unimarc/terms/mop/`.

While the above encoding takes a step towards the goals defined above, it can only be considered machine-readable to a limited degree. While it is possible to extract the number of oboe performers (`@count`), it is unclear how the doublings are divided amongst them. Although this might be addressed by (ab-)using the `@count` and `@n` attributes ([Listing 3](#)), the MEI-Guidelines 4.0.1 ([Roland & Kepper, 2019](#)) are not clear about applying these attributes in this way. Furthermore, in terms of machine-readability, `@n` can be considered underspecified, as it is not defined for this explicit use case.

```

<!-- ENCODING ALTERNATIVE 1 -->
<perfResList xmlns="http://www.music-encoding.org/ns/mei" meiversion="4.0.1">
  <perfRes count="3" codedval="wob" auth.uri="http://iflstandards.info/ns/unimarc/terms/mop/">3 OBOES
    <lb/><perfRes n="1-2" count="2" codedval="woa" auth.uri="mop">1. and 2. also oboe d'amore</perfRes>
    <lb/><perfRes n="3" count="1" codedval="weh" auth.uri="mop">3. also cor anglais</perfRes>
  </perfRes>
</perfResList>
<!-- ENCODING ALTERNATIVE 2 -->
<perfResList xmlns="http://www.music-encoding.org/ns/mei" meiversion="4.0.1">
  <perfRes count="3" codedval="wob" auth.uri="http://iflstandards.info/ns/unimarc/terms/mop/">3 OBOES
    <lb/><perfRes n="1" count="1" codedval="woa" auth.uri="mop">oboe d'amore</perfRes>
    <lb/><perfRes n="2" count="1" codedval="woa" auth.uri="mop">oboe d'amore</perfRes>
    <lb/><perfRes n="3" count="1" codedval="weh" auth.uri="mop">cor anglais</perfRes>
  </perfRes>
</perfResList>

```

**Listing 3:** Alternative encodings of a `<perfResList>` with another step towards machine-readability by using `@n` and `@count` to associate player numbers with doublings.

Nonetheless, nesting `<perfRes>` elements to deal with cases of instrumental doublings still raises some questions: What is the distinguishing difference between encoding doubling vs. non-doubling cases? What does that nesting imply – is one of the instruments subordinated to the other? At least, this is not the case here; Zimmermann wants one performer to play both instruments alternately without implying any preference or rating of one as the ‘primary’ instrument.

To add a level of complexity to this case, Zimmermann added a note at the foot of the instrument list offering fallback instrumentation: “If necessary, the oboe d’amore can be replaced by an oboe, [...]” (Zimmermann, 1970). Trying to capture this with MEI’s `<perfRes>` or `<perfResList>` elements pushes against the capabilities of MEI’s current content model for those elements.

Based on the experiences in the Zimmermann-Complete Edition, Herold (2018) presented the project’s custom-tailored solution to address the above questions. This solution added `<bazga:alternationGrp>` and `<bazga:alternate>` as wrapper elements around `<perfRes>` (using MEI 3.0.0 at the time, but the example easily translates to MEI 4.0.1) to make the distinction on a higher level. We adapted Herold’s encoding for this specific example and added the encoding of the fallback instruments (Listing 4).

Although we do not see a complete solution for the issues raised above in this encoding for several reasons,<sup>1</sup> it points in the right direction, namely in the relation of the individual performance resources.

---

<sup>1</sup> (1) Pulling the alternation to the level of the whole instrument group instead of making it on the level of individual instrumentalists (3 oboes vs. 2 oboe d’amore and 1 cor anglais) is problematic when investigating a single instrument; (2) `@type` on `<bazga:alternate>` holds values that could be considered judgemental, i.e., not free of preference; (3) the level of detail is not consistent between the three `<bazga:alternate>`-elements – while “primary” grasps all three oboe(-players) in one `<perfRes>`, “secondary” and “fallback” offer individual `<perfRes>`-elements if they hold a value for a player; the relation of a `<perfRes>` to a player is captured in `@n` which is weakly typed in terms of semantic information.

```

<perfResList xmlns="http://www.music-encoding.org/ns/mei" meiversion="3.0.0"
  xmlns:bazga="http://www.zimmermann-gesamtausgabe.de">
  <bazga:alternationGrp>
    <bazga:alternate type="primary">
      <perfRes count="3" auth.uri="mop" codedval="wob">3 Oboes</perfRes>
    </bazga:alternate>
    <bazga:alternate type="secondary">
      <perfRes count="1" auth.uri="mop" codedval="woa" n="1">oboe d'amore</perfRes>
      <perfRes count="1" auth.uri="mop" codedval="woa" n="2">oboe d'amore</perfRes>
      <perfRes count="1" auth.uri="mop" codedval="weh" n="3">cor anglais</perfRes>
    </bazga:alternate>
    <bazga:alternate type="fallback">
      <perfRes count="1" auth.uri="mop" codedval="woa" n="1">oboe</perfRes>
      <perfRes count="1" auth.uri="mop" codedval="woa" n="2">oboe</perfRes>
    </bazga:alternate>
  </bazga:alternationGrp>
</perfResList>

```

**Listing 4:** `<perfResList>` introducing BAZ-GA-specific elements for encoding instrument alternations and fallbacks, based on [Herold \(2018\)](#).

The above circled how to use `<perfResList>` and `<perfRes>` for our ‘pure metadata’ use case. We are fully aware that their parent node `<perfMedium>`, although defined in the MEI.header module, may also appear in `<titlePage>` (as opposed to `<work>` or `<expression>` from the MEI.frbr module) and consequently has to serve transcriptional purposes, too.

Focusing on the pure metadata use case, one solution could be to understand or redefine `<perfRes>` as something like a ‘performing force’ ([Roland & Kepper, 2019a](#)). This could be deemed tag-abuse when considering the element’s definition: “Name of an instrument on which a performer plays, a performer’s voice range, or a standard performing ensemble designation” ([Roland & Kepper, 2019b](#)) because `<perfRes>` does not define a ‘human resource’ or person. Also, `<perfResList>` is not of use here, as it is intended for grouping purposes only ([Roland & Kepper, 2019c](#)). Consequently, another solution could be the introduction of a separate encoding of ‘instrument’ and ‘performing force’, e.g., accompanying `<castList>` and `<perfResList>` – being the list of performers – with a new element `<instrList>` (instrument list) ([Listing 5](#)).

```

<perfMedium>
  <castList><!-- cast: roles and actors --></castList>
  <perfResList><!-- performers: 'human resources' --></perfResList>
  <instrList><!-- instrumentation: instruments needed --></instrList>
</perfMedium>

```

**Listing 5:** Separation of performance resources and instrumentation.

This proposed `<instrList>` element would capture instruments with their specificities, such as tuning, ambitus, or other attributes, in a machine-readable way. Especially for instruments tuned to certain pitches (like bells<sup>2</sup>), it is not enough to specify their number or grouping. Encoding relations between these resources (performer–instrument OR instrument–instrument)

---

<sup>2</sup> In this case, the octave could also be important.

could be made possible with nested `<relationList>` and `<relation>` elements. Furthermore, these relations could express qualities such as: ‘either ... or’, ‘both ... and’, ‘ad libitum’. The utilization of existing models like `<relationList>` or `<ambitus>` is straightforward as illustrated in [Listing 6](#).

```
<perfMedium>
  <perfResList>
    <perfRes count="1" xml:id="oboe.i">1st oboe</perfRes>
    <perfRes count="1" xml:id="oboe.ii">2nd oboe</perfRes>
    <perfRes count="1" xml:id="oboe.iii">3rd oboe</perfRes>
  </perfResList>
  <instrList>
    <instr count="3" xml:id="oboe">
      <label xml:lang="en">oboe</label>
      <tune pname="c"/>
      <ambitus/>
      <relationList>
        <relation plist="#oboe.i #oboe.ii #oboe.iii"/>
        <relation rel="isFallback" plist="#oboeDAMore"/>
      </relationList>
    </instr>
    <instr count="1" xml:id="oboeDAMore">
      <label xml:lang="en">oboe d'amore</label>
      <tune pname="c"/>
      <ambitus/>
      <relationList>
        <relation rel="isAlternation" plist="#oboe.ii"/>
      </relationList>
    </instr>
    <instr count="1" xml:id="englHorn">
      <label xml:lang="en">oboe</label>
      <tune pname="e" accid="f"/>
      <ambitus/>
      <relationList>
        <relation rel="isAlternation" plist="#oboe.iii"/>
      </relationList>
    </instr>
  </instrList>
</perfMedium>
```

**Listing 6:** One proposal on how to encode instrumentation separately, using `<relationList>` and `<relation>` to address the issues of alternating or fallback instruments. We used the semantically relatively weak `@rel` for readability reasons instead of pointing to relations defined in taxonomies with `@ref`.

## 1.2 Performer and set of instruments

In the following, the work *Minotauros Blues. Concert music for six percussionists* (1996) by the German composer Hans Werner Henze will be used as an example ([Henze, 1996](#)). The assignment of instruments to the performers is simple here: for each of the six percussionists, there is a list of instruments that he or she is supposed to play in the course of the work. The solution is

obvious: create a list (`<perfResList>`) for each percussion setup, which lists the required instruments with `<perfRes>` ([Listing 7](#)).<sup>3</sup>

```
<perfResList>
  <perfResList label="perc.i">
    [...]
  </perfResList>
  <perfResList label="perc.ii">
    [...]
  </perfResList>
  <perfResList label="perc.iii">
    [...]
    <perfRes>Gran Cassa (o O-daiko)</perfRes>
    [...]
    <perfRes>Marimba (a cinque ottavi)</perfRes>
  </perfResList>
  <perfResList label="perc.iv">
    [...]
    <perfRes>Marimba (a quattro ottavi)</perfRes>
  </perfResList>
  <perfResList label="perc.v">
    [...]
    <perfRes>Marimba (a quattro ottavi)</perfRes>
  </perfResList>
  <perfResList label="perc.vi">
    [...]
    <perfRes>Frusta (o Hyoshigi) a piede</perfRes>
  </perfResList>
</perfResList>
```

**Listing 7:** Extract of performance resources for Henze’s *Minotaurus Blues*.

As indicated earlier, `<perfResList>` shall not be used for describing a performance resource but for grouping instrumental resources exclusively. In the current example ([Listing 7](#)), the semantic meaning of `<perfResList>` becomes bilateral because each `<perfResList>` defines a set of instruments and also represents a performance resource.

Even though this looks like a simple example, there are already problems in the capture here. For percussionist III, an alternative is called: Gran Cassa or O-Daiko. As mentioned above, the problem arises in the hierarchical structure of nested elements: nesting implies a hierarchy that is not meant here because it is an alternative on the same level: ‘use A or B’ and not ‘use B instead if A is not available’ (see [Listing 3](#)). Another problem is posed by specific characteristics of the instruments needed for a performance. For example, two marimbas are needed for the performance of this work, but not just any marimbas: percussionist III needs a marimba with an ambitus of 5 octaves, while percussionists IV and V need one with 4 octaves each (see [Listing 7](#)). The whole thing can be pushed even further: The alternation for percussionist VI “Frusta or Hyoshigi a piede” not only assigns the instrument to a ‘human resource’, but also indicates that it should be played by foot (“a piede”). Gubsch & Ried ([2021](#)) mentioned the need to address the

---

<sup>3</sup> The full instrumentation is recorded on the homepage of the Hans-Werner-Henze-Stiftung ([Henze, 1996](#)).

hands and feet of a performer for encoding, e.g., piano and organ music with more than one performer on one instrument.

## Conclusion

Because of the complexity of the examples, just two have been discussed here. Nevertheless, this poster shows the state of our work, which is still in the thinking stage. We are currently investigating which parameters to record and how to do so in a machine-readable form. Care has to be taken to avoid overloading existing elements with content models for different purposes (transcription vs. structured metadata). Although some elements (e.g., `<perfMedium>`) do already support different content models depending on their context in an MEI file (transcription vs. structured metadata), care has to be taken to avoid overloading their content options. Introducing new elements and constraining or adjusting the definition of existing elements could prevent this. A well-balanced solution will certainly take time but promises to be fruitful.

## References

- Gubsch, C., & Ried, D. (2021). *METAdata and metaDATA* [Conference poster presentation]. Music Encoding Conference 2021, Alicante. [publication in preparation]
- Henze, H. W. (1996). *Minotaurus Blues. Concert music for six percussionists*. Retrieved April 24, 2022, from <https://www.hans-werner-henze-stiftung.de/en/hans-werner-henze/list-of-works/detail/news/detail/News/minotauros-blues/>, accessed: 24. April 2022.
- Herold, K. (2018). *Zimmermann as a Proof of Concept for an MEI Work Catalogue* [Conference poster presentation]. Music Encoding Conference 2018, College Park, MD.
- IFLA. (n.d.). *UNIMARC: Medium of performance*. Retrieved April 24, 2022, from <https://www.iflstandards.info/unimarc/terms/mop.html>
- Roland, P., & Kepper, J. (Eds.). (2019). Chapter 3.6.7. Performance Medium. In *Guidelines (4.0.1)*. Retrieved April 24, 2022, from <https://music-encoding.org/guidelines/v4/content/metadata.html#headerWorkMedium>
- Roland, P., & Kepper, J. (Eds.). (2019a). `<perfMedium>`. In *Guidelines (4.0.1)*. Retrieved April 24, 2022, from <https://music-encoding.org/guidelines/v4/elements/perfmedium.html>
- Roland, P., & Kepper, J. (Eds.). (2019b). `<perfRes>`. In *Guidelines (4.0.1)*. Retrieved April 24, 2022, from <https://music-encoding.org/guidelines/v4/elements/perfres.html>
- Roland, P., & Kepper, J. (Eds.). (2019c). `<perfResList>`. In *Guidelines (4.0.1)*. Retrieved April 24, 2022, from <https://music-encoding.org/guidelines/v4/elements/perfreslist.html>
- Zimmermann, Bernd Alois (1970). *Photoptosis. Prélude für großes Orchester (1968)* [Study score ED 6311].