The Music Annotation Pattern

Jacopo de Berardinis[1,*], Albert Meroño-Peñuela[1], Andrea Poltronieri[2,*] and Valentina Presutti[2]

[1]King's College London
[2]University of Bologna

Abstract

The annotation of music content is a complex process to represent due to its inherent multifaceted, subjectivity, and interdisciplinary nature. Numerous systems and conventions for annotating music have been developed as independent standards over the past decades. Little has been done to make them interoperable, which jeopardises cross-corpora studies as it requires users to familiarise with a multitude of conventions. Most of these systems lack the semantic expressiveness needed to represent the complexity of the musical language and cannot model multi-modal annotations originating from audio and symbolic sources. In this article, we introduce the *Music Annotation Pattern*, an Ontology Design Pattern (ODP) to homogenise different annotation systems and to represent several types of musical objects (e.g. chords, patterns, structures). This ODP preserves the semantics of the object's content at different levels and temporal granularity. Moreover, our ODP accounts for multi-modality upfront, to describe annotations derived from different sources, and it is the first to enable the integration of music datasets at a large scale.

Keywords

Semantic Web, Ontology, Music Information Retrieval, Computational Musicology

1. Introduction

Similarly to other forms of artistic expression, the analysis of music can be considered as a quest for meaning – a process driven by musical theories and perceptual cues attempting to shed light on the potentially ambiguous and intricate messages that artists have encoded in their music [1]. Starting from a composition or a performance, music analysis usually focuses on detecting elements related to harmony, form, texture, etc., along with the identification of potential interrelated functions they may exert in the piece (creating or releasing tension, evoking images, inducing emotions, etc.) [2].

At the core of this multifaceted process lies the ability to effectively annotate music For example, if the goal of a harmonic analysis is to identify chords from a composition, a music annotation may correspond to a list of chords together with a reference to their onset (i.e. when they occur in the piece). Besides contributing to the more general goal of understanding

WOP2022: 13th Workshop on Ontology Design and Patterns, October 23–24, 2022, Hangzhou, China

[*]Corresponding author.

jacopo.deberardinis@kcl.ac.uk (J. d. Berardinis); albert.merono@kcl.ac.uk (A. Meroño-Peñuela); andrea.poltronieri2@unibo.it (A. Poltronieri); valentina.presutti@unibo.it (V. Presutti)
0000-0001-6770-1969 (J. d. Berardinis); 0000-0003-4646-5842 (A. Meroño-Peñuela); 0000-0003-3848-7574 (A. Poltronieri); 0000-0002-9380-5160 (V. Presutti)

music, these annotations are also of pedagogic interest (e.g. teaching material for classrooms in analysis, harmony, or composition) and of musicological relevance. They also provide valuable data for training and evaluating algorithmic methods for music information retrieval (MIR) and computational music analysis (CMA), and for supporting performers studying scores and preparing their own interpretation [3]. This interdisciplinary interest in music annotations has also fuelled the development of novel applications and workflows focused on their collection [4], interaction [5] and sharing [3].

Nevertheless, annotating music has always been a challenging task in many respects [6]. Musical content is rich in components (voices, sections, etc.) and nuances (accents, prolongations, modulations) that are often difficult to represent and to consistently relate to the content of an annotation. Several types of musical notations have been introduced to address this problem, although primarily focused on representing musical scores (c.f. Section 2.1). Even the score itself is based on conventions and symbols that have evolved diachronically – as musical periods have changed, as well as stylistically – as musical genres vary [7]. This evanescent aspect of music is then more pronounced when focusing on representing music annotations. For example, when annotating chords, different notation systems have been used over the years, starting with the *basso continuo*, almost universally used in the Baroque era, to the modern *Leadsheet notations*, mainly used to annotate chords in Jazz music [8].

A multitude of notation systems have been developed, proposing different approaches on how to annotate music. This fragmentation is reflected in a vast heterogeneity of file formats and extensions, with consequent interoperability problems. When annotations are encoded within a score, software tools for music processing and computer-aided musicology, like `music21` [9] and *note-seq*[1], have rapidly evolved to parse a variety of symbolic formats [2]. When annotations are decoupled from the music content [10, 11], these are often encoded using dataset-specific standards and conventions. As a result, retrieving and integrating music annotations from different sources is a challenging, time-consuming task, which stems from the encoding problem and the lack of well-established standards for releasing music datasets [12]. This brings a cascade of effects: (i) it limits the ability to perform cross-corpora studies, especially in multi-modal settings – involving both audio and score annotations; (ii) it leaks ambiguity in the annotations due to the poor semantic expressiveness of the current approaches; and (iii) it confines users to familiarising with a multitude of standards.

In this article, we introduce the *Music Annotation Pattern*, an Ontology Design Pattern for modelling a wide set of music annotations. The *Music Annotation Pattern* is a reusable block for representing annotations of different types, from different sources, and addressing heterogeneous timing conventions. The ODP has been used in preliminary experiments integrating harmonic datasets (chord annotations from multiple sources) in the Polifonia project[3]. To our best, it is the first attempt at achieving semantic interoperability of music annotations collected from multi-modal sources.

[1] https://github.com/magenta/note-seq

[2] See, for example https://web.mit.edu/music21/doc/moduleReference/moduleConverter.html

[3] https://polifonia-project.eu

2. Related Work

The complexity of representing musical content is related to the manifold sources that are available when studying music. To contextualise this process, Vinet [13] introduces two different *Representation Levels* to categorise different types of music content: *signal representations* and *symbolic representations*. A symbolic representation is context-aware and describes events in relation to formalised concepts of music (music theory), whereas the signal representation is a blind, context-unaware representation, thus adapted to transmit any, non-musical kind of sound, and even non-audible signals.

In this paper, we focus on symbolic representation systems and how these can be semantically described to address the three challenges outlined in the introduction.[4]

2.1. Modelling scores and score-embedded annotations

Over the years, various representation systems have been developed, some of which are still used today. A notable example is MIDI (Musical Instrument Digital Interface) [14], which also provides a data communication protocol for music production and live performance. A MIDI file can be described as a stream of events, each defined by two components: *MIDI time* and *MIDI message*. The time value describes the time to wait (a temporal offset) before executing the following message. The message value, instead, is a sequence of bytes, where the first one is a command, often followed by complementary data.

The ABC notation [15] is a text-based music notation system and the de facto standard for folk and traditional music. An ABC tune consists of a *tune header* and a *tune body*, terminated by an empty line or the end of the file. The *tune header* contains the tune's metadata, and can be filled with 27 different fields that describe composer, tempo, rhythm, source, etc. The tune body, instead, describes the actual music content, such as notes, rests, bars, chords, and clefs.

MusicXML [16] is an XML-based music interchange language. It is intended to represent common western musical notation from the seventeenth century onwards, including both classical and popular music. Similarly to MIDI, MusicXML defines both an interchange language and a file format (in this case XML).

The Music Encoding Initiative (MEI) [17] is a community-driven, open-source effort to define a system for encoding musical documents in a machine-readable structure. The community formalised the MEI schema, a core set of rules for recording physical and intellectual characteristics of music notation documents expressed with an XML schema. This framework aims at preserving the XML compatibility while expressing a wide level of music nuances.

Other systems of symbolic notation include the *CHARM system* [18], ***kern* [19] and *LilyPond* [20]. All these formats differ dramatically in their syntax, which may exacerbate the interoperability problem and the consequent fragmentation of music data.

2.2. Modelling decoupled annotations

To overcome these problems, annotation standards have been proposed to decouple annotations from the scores, and to encode them in a separate yet unified format. The most notable example

[4] This does not imply that a symbolic annotation cannot also refer to audio music (*alias* tracks, recordings).

is the Annotated Music Specification for Reproducible MIR Research (JAMS) [21, 22], a JSON-based format to encode music annotations. It is primarily used to train and evaluate MIR algorithms, especially in the audio domain. JAMS supports the annotation of several music object types – from notes and chords to patterns and emotions, unambiguously defining the onset, duration, value and confidence of each observation (e.g. ”C:major” starting at second 3, lasting for 4 seconds, detected with a confidence level of 90%). This standard also offers the possibility of storing multiple and heterogeneous annotations in the same file, as long as they pertain to the same piece. Notably, JAMS provides a loose schema to record metadata, both related to the track (title, artists, etc.) and to each annotation (annotator, annotation tools, etc.).

Nonetheless, JAMS supports annotations collected from signal representation (audio), as it was not originally designed for the symbolic domain. This is due to a discrepancy between audio-based annotations – expressing temporal information in absolute times (seconds), and symbolic annotations – using relative or metrical temporal anchors (e.g. beats, measures). Also, from a descriptive perspective, it is not possible to disambiguate certain attributes in the metadata sections. For instance, the “`artist`” field in the current JAMS definition may refer to the composer or to the performer of the piece. Finally, JAMS is limited to the expressiveness of JSON, which does not allow for the semantic expression of concepts that are sometimes essential for describing musical content. For example, even if the specification of composers and performers was possible in the standard, this would still be insufficient to express the semantic relationships occurring between these concepts.

2.3. Modelling semantics in music data

To encode semantics in music data, and account for the ambiguity problem in music annotations, Semantic Web technologies can be useful, as shown in other domains such as Cultural Heritage [23]. Over the past two decades, several ontologies have been developed in the music domain. Some ontologies have been designed for describing high-level descriptive and cataloguing information, such as the The Music Ontology [24] and the DOREMUS Ontology [25].

Other ontologies describe musical notation, both from the music score and the symbolic points of view. For example, the MIDI Linked Data Cloud [26] models symbolic music descriptions encoded in MIDI format. The Music Theory Ontology (MTO) [27] aims to describe theoretical concepts related to a music composition, while The Music Score Ontology (Music OWL) [28] represents similar concepts with a focus on music sheet notation. Finally, the Music Notation Ontology [29] focuses on the core “semantic” information present in a score. The Music Encoding and Linked Data framework (MELD) [30] reuses multiple ontologies, such as the Music and Segment Ontologies, FRBR in order to describe real-time annotation of digital music scores. The Music Note Ontology [31] proposes to model the relationships between a symbolic representation and the audio representation, but only considering the structure of the music score and the granularity level of the music note.

Each of these ontologies covers a specific aspect of music notations. Our ODP reuses and extend their modelling solutions to provide a comprehensive, scalable and coherent representation music annotations.

3. The Music Annotation Pattern

The Music Annotation ODP addresses the goal of modelling different types of musical annotations. For example, this ODP can be used to describe musical chords, notes rather than patterns, both harmonic and melodic and structural annotations. The *Music Annotation* ODP also aims to represent annotations derived from different types of sources, such as audio and score.
The ODP is represented in Figure 1 an it is available online at the following URI:

https://purl.org/andreapoltronieri/music-annotation-pattern

The complete implementation and documentation of the pattern, as well as its documentation and all the examples presented in this paper are available on a dedicated GitHub repository[5].
To be compliant with the practice of the Music Information Retrieval community, we reuse the terminology from JAMS[6] [21]. The following terms are used for the ODP vocabulary:

- *Music Annotation:* a music annotation is defined as a group of `Music Observations` (see below) that share certain elements, such as the method used for the annotation and the type of object being annotated (e.g. chords, notes, patterns); an annotation has one and only one annotator, that can be of different types e.g., a human, a computational method, and which is the same for all its observations.

- *Music Observation:* a music observation is defined as the content of a music annotation. It includes all the elements that characterise the observation. For example, in the case of an annotation of chords, each observation is associated with one chord, and it specifies, in addition to the chord value, its related temporal information and the confidence of the annotator for that observation.

The structure of the Music Annotation ODP consists of the relations between an `MusicAnnotation` and its `MusicObservations`.
An integration effort of a set of datasets containing chord annotations, in the context of the Polifonia project[3], provided a useful empirical ground to define a set of Competency questions (CQs) to drive the design of the Music Annotation ODP. They are listed in Table 1. Each competency question is associated with a corresponding SPARQL query, they are all available on the project's GitHub repository[5].
The ODP was modelled by following a CQ-driven approach [32], and by reusing a JAMS-based terminology.

Annotation. Addressing CQ1, CQ4, CQ10: a `MusicAnnotation` has to be intended as a collection of `MusicObservations` about a `MusicalObject`. For musical objects, in this context, we refer to a concept generalising over audio tracks and scores. `MusicAnnotations` can be of two types: `ScoreMusicAnnotation` and `AudioMusicAnnotation`.

[5] Music Annotation Pattern repository: https://github.com/andreamust/music-annotation-pattern
[6] Official JAMS documentation: https://jams.readthedocs.io/en/stable/

ID	Competency questions
CQ1	What is the type of a music annotation/observation for a musical object?
CQ2	What is the time frame within the musical object addressed by an annotation?
CQ3	What is its start time (i.e. the starting time of the time frame)?
CQ4	Which are the observations included in an annotation?
CQ5	For a specific music observation, what is the starting point of its addressed time frame, within its reference musical object?
CQ6	For a specific music observation, what is its addressed time frame, within the musical object?
CQ7	What is the value of a music observation?
CQ8	Who/what is the annotator of a music annotation/observation, and what is its type?
CQ9	What is the confidence of a music observation?
CQ10	What is the musical object addressed by a music annotation?

Table 1
Competency questions addressed by the Music Annotation ODP.

Time information. Addressing CQ2, CQ3, CQ5, CQ6. The temporal information of a `MusicAnnotation` and a `MusicObservation` is expressed in the same way, thus effectively creating an independent pattern for describing musical time intervals. This pattern is composed by a `MusicTimeInterval` which in turn defines a `MusicTimeIndex` and a `MusicTimeDuration`. They

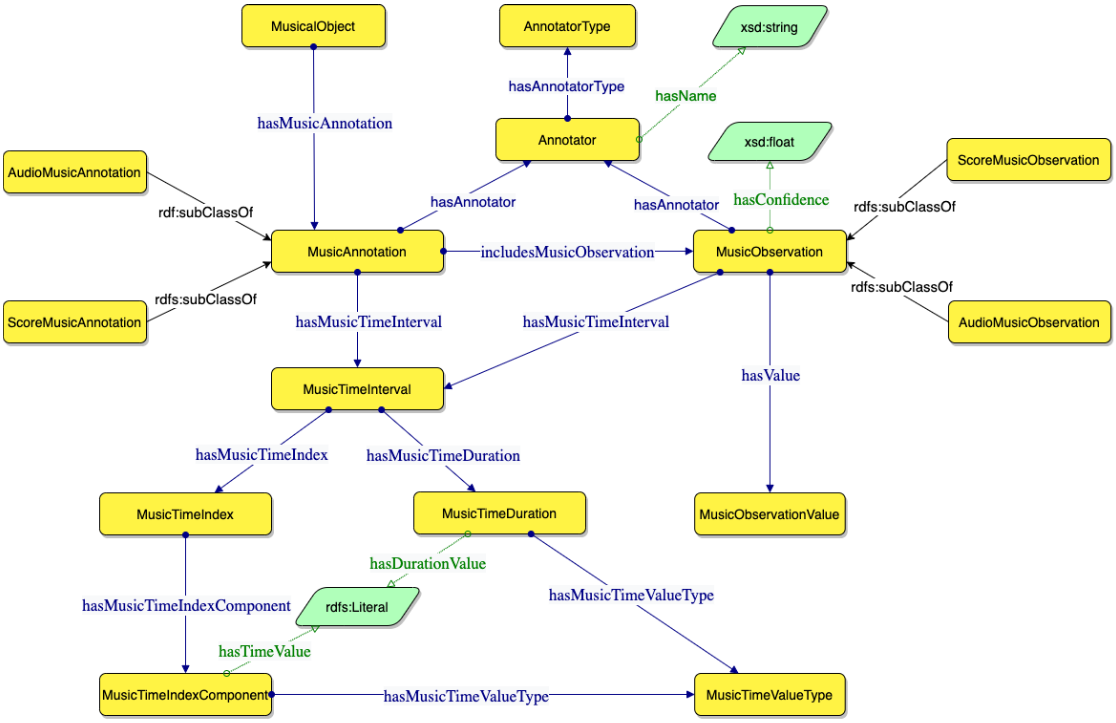

Figure 1: The Music Annotation Pattern. We use the Graffoo notation: yellow boxes are classes, blue/green arrows are object/datatype properties, purple circles are individuals, green polygons are datatypes.

indicate the time frame, within the referenced musical object, addressed by a music annotation/observation. More specifically, a MusicTimeIndex defines the start point of the annotation, while MusicTimeDuration describes the duration of the annotation.
Each MusicTimeIndex is composed of one or more components, namely MusicTimeIndexComponents. The latter, as well as the MusicTimeDuration, defines the value of the temporal annotation via a datatype property hasTimeValue, which has as range rdfs:Literal, and the format of the annotation itself, expressed by the MusicTimeValueType class.
In the case of AudioMusicAnnotation and AudioMusicObservation, the start time of the annotation shall be expressed by a single MusicTimeIndexComponent, which will have as MusicTimeValueType a time format in seconds, minutes or milliseconds. Instead, in the case of ScoreAnnotation and ScoreObservation two MusicTimeIndexComponents will be needed to describe the start time, the first to describe the beat in which the annotation begins and the second to describe the beat within the measure in which the annotation starts.

```
Class: MusicTimeInterval
    SubClassOf:
        hasMusicTimeDuration only MusicTimeDuration,
        hasMusicTimeIndex only MusicTimeIndex,
        hasMusicTimeDuration exactly 1 MusicTimeDuration,
        hasMusicTimeIndex exactly 1 MusicTimeIndex

Class: MusicTimeIndex
    SubClassOf:
        hasMusicTimeIndexComponent only MusicTimeIndexComponent,
        hasMusicTimeIndexComponent min 1 MusicTimeIndexComponent

Class: MusicTimeIndexComponent
    SubClassOf:
        hasMusicTimeValueType only MusicTimeValueType,
        hasMusicTimeValueType exactly 1 MusicTimeValueType,
        hasTimeValue only rdfs:Literal,
        hasTimeValue exactly 1 rdfs:Literal
```

Annotator. Addressing CQ8. Annotations have one and only one Annotator, relation expressed through the object property hasAnnotator. Annotators are classified by their type (AnnotatorType), for example Human, Machine, Crowdsourcing, etc., which is exactly one.

```
ObjectProperty: hasAnnotator
    SubPropertyChain:
        isAnnotatorOf o includesMusicObservation
    Domain:
        MusicAnnotation
    Range:
        Annotator
```

Music Observation. Addressing CQ1, CQ4, CQ7, CQ9. Each MusicAnnotation includes a set of MusicObservations. MusicObservations can be of two types: ScoreMusicObservation and AudioMusicObservation. The type of an observation must be compatible with the type

of the annotation that contains them. If the annotation is `ScoreMusicAnnotation`, it contains `ScoreMusicObservations`, otherwise it contains `AudioMusicObservations`. The annotator (and its type) of an observation is the same and only from the annotation that includes it: this is formalised by means of a property chain in the ODP. However, the level of confidence of an annotator is associated to each observation (`hasConfidence`).
Each `MusicObservation` has an `MusicObservationValue`, which characterises its content. The `MusicObservationValue` class is meant to be specialised depending on the subject being observed (and annotated), e.g. Chord, Note, Structural Annotation. For example, it can generalise over concepts from existing ontologies, such as the Chord Ontology[7] for chord annotations. Musical object, music annotation, music observation, music observation value, music time interval, annotator, and annotator type are disjoint concepts.

4. Usage Example

In this section, we describe two examples of usage of the *Music Annotation* ODP. We remind that this ODP addresses different types of annotations for different types of sources (e.g. score, audio). The examples show how the Music Annotation ODP can be used to describe: (i) musical chord annotations and (ii) structural annotations of a song.

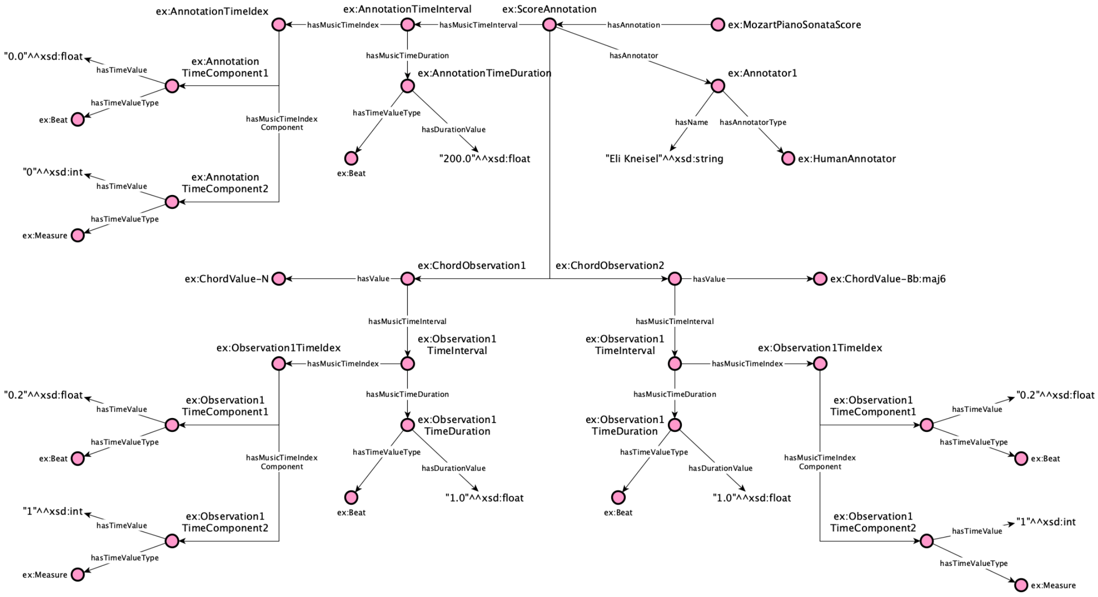

Figure 2: The Music Annotation ODP used to represent chord annotations from Wolfgang Amadeus Mozart's *Piano Sonata no. 1 in C major (Allegro)*. The original annotation is taken from [33].

[7] Chord Ontology documentation available at: http://motools.sourceforge.net/chord_draft_1/chord.html

4.1. Chord Annotations

The first example is an annotation of chords from a music score of Wolfgang Amadeus Mozart's *Piano Sonata no. 1 in C major (Allegro)*. The original annotation is taken from the Mozart Piano Sonatas Dataset [33]. Figure 2 depicts the resulting RDF graph using the Grafoo Notation[8]. In all the examples, dummy prefix and namespace (`ex:` and http://example.org/) are defined for instances.

In this case, the `MusicalObject` is a musical score, defined by the `ex:MozartPianoSonataScore` instance, which has `ex:ScoreAnnotation` as its annotation. The annotation is linked to its annotator, in this case a human and to its `MusicTimeInterval`. The `MusicTimeInterval` defines the duration of the annotation, by means of the `MusicTimeDuration` class, and the start point of the annotation, by means of the `MusicTimeIndex` class. The latter, being the annotation is of type *score*, contains two different `MusicTimeIndexComponents`: the first has as its `MusicTimeValueType` a `ex:Measure`, which indicated the measure at which the annotation starts, while the second has as value type a `ex:Beat`, which describes the beat within the measure at which the annotation begins. Duration is instead expressed only in *beats*.

The annotation then contains two different observations (the actual number has been reduced for demonstration purposes), namely `ex:ChordObservation1` and `ex:ChordObservation2`.

Each of these observations has a value, i.e. the chord per se, and a time interval. In this example, observations have no `Confidence`, as this is not provided by the original annotation.

4.2. Structural Annotations

The second example is an annotation of segments from an audio track of The Beatles' *Michelle*. The original annotation is available in JAMS format and is taken from the Isophonics[9] [34]. Figure 3 depicts the example graphically using the Grafoo notation.

In this example, the `MusicalObject` is instead a track, defined by the `ex:BeatlesMichelleTrack` instance, which has an `ex:AudioMusicAnnotation`, as it was annotated from the audio signal. The annotation has a human-type annotator and an annotation time interval.

The annotation then contains two different `SegmentObservation`, which define the structure of the track. Each observation has a starting time and duration, defined by the classes `MusicTimeIndex` and `MusicTimeDuration`, respectively. In this case, there is only a single `MusicTimeIndexComponent`, since the time information is expressed in seconds (`ex:Seconds`). Finally, the value of each observation corresponds to the structural segment itself, in this case `ex:Silence` and `ex:Intro`.

5. Conclusions and Future Work

We propose the *Music Annotation* ODP for modelling annotations of music scores and audio tracks. A distinction at the core of this ODP is the different encoding of time information, which depends on the type of the subject of observation (score or audio). The ODP is the result of the analysis of many relevant different existing formats used for music annotation (MusicXML,

[8] https://essepuntato.it/graffoo/

[9] Isophonics dataset: http://isophonics.net/datasets

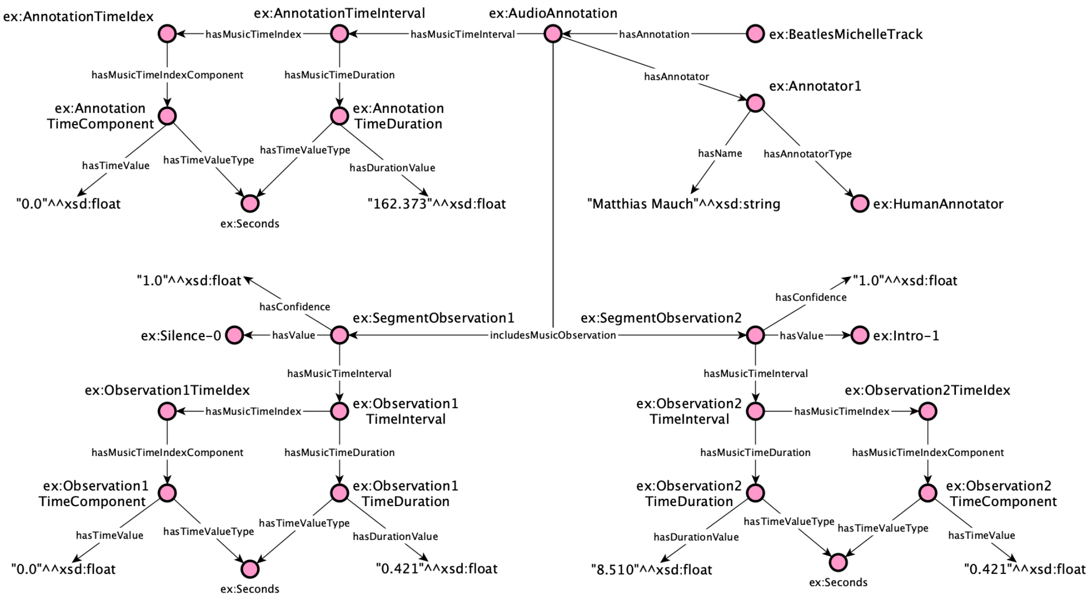

Figure 3: The Music Annotation ODP used to represent segment annotations from an audio track of The Beatles' *Michelle*. The original annotation is taken from the Isophonics.

ABC, JAMS, etc.) and provides a template for supporting the integration of data from such heterogeneous sources. This work demonstrated the use of the ODP for modelling harmonic and structural annotations (chords, segments) collected from symbolic and audio sources. We plan to follow up with a large scale integration experiment on a selection of MIR datasets, and the extension of our pattern to model additional types of music annotations.

Acknowledgments

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101004746.

References

[1] A. Pople, Theory, analysis and meaning in music, Cambridge University Press, 2006.

[2] B. A. Johnston, Harmony and climax in the late works of Sergei Rachmaninoff, University of Michigan, 2009.

[3] M. Giraud, R. Groult, E. Leguy, Dezrann, a web framework to share music analysis, in: International Conference on Technologies for Music Notation and Representation (TENOR 2018), 2018, pp. 104–110.

[4] D. Turnbull, R. Liu, L. Barrington, G. R. Lanckriet, A game-based approach for collecting semantic annotations of music., in: ISMIR, volume 7, 2007, pp. 535–538.

[5] L. Pugin, Interaction perspectives for music notation applications, in: Proceedings of the 1st International Workshop on Semantic Applications for Audio and Music, 2018.

[6] A. Hadjakos, J. Iffland, R. Keil, A. Oberhoff, J. Veit, Challenges for annotation concepts in music, International Journal of Humanities and Arts Computing 11 (2017) 255–275.

[7] P. E. Savage, Cultural evolution of music, Palgrave Communications 5 (2019) 1–12.

[8] J. Kite-Powell, A performer's guide to seventeenth-century music, Indiana University Press, 2012.

[9] M. S. Cuthbert, C. Ariza, music21: A toolkit for computer-aided musicology and symbolic music data (2010).

[10] V. Eremenko, E. Demirel, B. Bozkurt, X. Serra, Jaah: Audio-aligned jazz harmony dataset, 2018. URL: https://doi.org/10.5281/zenodo.1290737. doi:10.5281/zenodo.1290737.

[11] M. Neuwirth, D. Harasim, F. C. Moss, M. Rohrmeier, The annotated beethoven corpus (abc): A dataset of harmonic analyses of all beethoven string quartets, Frontiers in Digital Humanities (2018) 16.

[12] V. A. Carriero, F. Ciroku, J. de Berardinis, D. S. M. Pandiani, A. Meroño-Peñuela, A. Poltronieri, V. Presutti, Semantic integration of mir datasets with the polifonia ontology network, in: ISMIR Late Breaking Demo, 2021.

[13] H. Vinet, The representation levels of music information, in: U. K. Wiil (Ed.), Computer Music Modeling and Retrieval, Springer Berlin Heidelberg, Berlin, Heidelberg, 2004.

[14] International MIDI Association, MIDI Musical Instrument Digital Interface Specification 1.0, Technical Report, Los Angeles, 1983.

[15] C. Walshaw, The ABC music standard 2.1., Technical Report, abcnotation.com, 2011.

[16] M. Good, Musicxml: An internet-friendly format for sheet music, in: XML conference and expo, 2001, pp. 03–04.

[17] P. Roland, The music encoding initiative (MEI), in: Proceedings of the First International Conference on Musical Applications Using XML, volume 1060, 2002, pp. 55–59.

[18] A. Smaill, G. Wiggins, M. Harris, Hierarchical music representation for composition and analysis, Computers and the Humanities 27 (1993) 7–17.

[19] D. Huron, Music information processing using the humdrum toolkit: Concepts, examples, and lessons, Computer Music Journal 26 (2002) 11–26. URL: http://www.jstor.org/stable/3681454.

[20] H.-W. Nienhuys, J. Nieuwenhuizen, Lilypond, a system for automated music engraving, in: Proceedings of the XIV Colloquium on Musical Informatics (XIV CIM 2003), volume 1, 2003, pp. 167–171.

[21] E. J. Humphrey, J. Salamon, O. Nieto, J. Forsyth, R. M. Bittner, J. P. Bello, JAMS: A JSON annotated music specification for reproducible MIR research, in: H. Wang, Y. Yang, J. H. Lee (Eds.), Proceedings of the 15th International Society for Music Information Retrieval Conference, ISMIR 2014, Taipei, Taiwan, October 27-31, 2014, 2014, pp. 591–596. URL: http://www.terasoft.com.tw/conf/ismir2014/proceedings/T106_355_Paper.pdf.

[22] B. McFee, E. J. Humphrey, O. Nieto, J. Salamon, R. M. Bittner, J. Forsyth, J. P. Bello, Pump Up The JAMS: V0.2 And Beyond, Technical Report, 2015.

[23] V. A. Carriero, A. Gangemi, M. L. Mancinelli, L. Marinucci, A. G. Nuzzolese, V. Presutti, C. Veninata, Arco: The italian cultural heritage knowledge graph, in: C. Ghidini, O. Hartig, M. Maleshkova, V. Svátek, I. Cruz, A. Hogan, J. Song, M. Lefrançois, F. Gandon (Eds.), The

Semantic Web – ISWC 2019, Springer International Publishing, Cham, 2019, pp. 36–52.

[24] Y. Raymond, S. Abdallah, M. Sandler, F. Giasson, The music ontology, in: Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR 2007), Vienna, Austria, 2007, pp. 417–422.

[25] P. Lisena, R. Troncy, Doing reusable musical data (DOREMUS), in: Proceedings of Workshops and Tutorials of the 9th International Conference on Knowledge Capture (K-CAP2017), Austin, Texas, USA, December 4th, 2017, volume 2065 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2017, pp. 64–68.

[26] A. Meroño-Peñuela, R. Hoekstra, A. Gangemi, P. Bloem, R. de Valk, B. Stringer, B. Janssen, V. de Boer, A. Allik, S. Schlobach, K. Page, The MIDI Linked Data Cloud, in: The Semantic Web – ISWC 2017, Springer International Publishing, Cham, 2017, pp. 156–164.

[27] S. M. Rashid, D. De Roure, D. L. McGuinness, A music theory ontology, in: Proceedings of the 1st International Workshop on Semantic Applications for Audio and Music, SAAM ’18, Association for Computing Machinery, New York, NY, USA, 2018, p. 6–14. URL: https://doi.org/10.1145/3243907.3243913. doi:10.1145/3243907.3243913.

[28] J. Jones, D. de Siqueira Braga, K. Tertuliano, T. Kauppinen, MusicOWL: The music score ontology, in: Proceedings of the International Conference on Web Intelligence, WI ’17, Association for Computing Machinery, New York, NY, USA, 2017, p. 1222–1229. URL: https://doi.org/10.1145/3106426.3110325. doi:10.1145/3106426.3110325.

[29] S. S.-s. Cherfi, C. Guillotel, F. Hamdi, P. Rigaux, N. Travers, Ontology-based annotation of music scores, in: Proceedings of the Knowledge Capture Conference, K-CAP 2017, Association for Computing Machinery, New York, NY, USA, 2017. URL: https://doi.org/10.1145/3148011.3148038. doi:10.1145/3148011.3148038.

[30] K. R. Page, D. Lewis, D. M. Weigl, Meld: A linked data framework for multimedia access to music digital libraries, in: 2019 ACM/IEEE Joint Conference on Digital Libraries (JCDL), 2019, pp. 434–435. doi:10.1109/JCDL.2019.00106.

[31] A. Poltronieri, A. Gangemi, The music note ontology, in: K. Hammar, C. Shimizu, H. Küçük McGinty, L. Asprino, V. A. Carriero (Eds.), Proceedings of the 12th Workshop on Ontology Design and Patterns (WOP 2021), Online, October 24, 2021., 2021.

[32] V. Presutti, E. Daga, A. Gangemi, E. Blomqvist, extreme design with content ontology design patterns, in: E. Blomqvist, K. Sandkuhl, F. Scharffe, V. Svátek (Eds.), Proceedings of the Workshop on Ontology Patterns (WOP 2009) , collocated with the 8th International Semantic Web Conference (ISWC-2009), Washington D.C., USA, 25 October, 2009, volume 516 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2009. URL: http://ceur-ws.org/Vol-516/pap21.pdf.

[33] J. Hentschel, M. Neuwirth, M. Rohrmeier, The annotated mozart sonatas: Score, harmony, and cadence, Trans. Int. Soc. Music. Inf. Retr. 4 (2021) 67–80.

[34] M. Mauch, C. Cannam, M. Davies, S. Dixon, C. Harte, S. Kolozali, D. Tidhar, OMRAS2 metadata project 2009, in: In Late-breaking session at the 10th International Conference on Music Information Retrieval (ISMIR), 2009.