Audio Commons Ontology: A Data Model for an Audio Content Ecosystem

Miguel Ceriani[✉] and György Fazekas[✉]

Centre for Digital Music, Queen Mary University of London, London, UK
{m.ceriani,g.fazekas}@qmul.ac.uk

Abstract. Multiple online services host repositories of audio clips of different kinds, ranging from music tracks, albums, playlists, to instrument samples and loops, to a variety of recorded or synthesized sounds. Programmatic access to these resources maybe used by client applications for tasks ranging from customized musical listening and exploration, to music/sounds creation from existing sounds and samples, to audio-based user interaction in apps and games. We designed an ontology to facilitate interoperability between repositories and clients in this domain. There was no previous comprehensive data model for our domain, however the new ontology relates to existing ontologies, such as the Functional Requirements for Bibliographic Records for the authoring and publication process of creative works, the Music Ontology for the authoring and publication of music, the EBU Core ontology to describe media files and formats and the Creative Commons Licensing ontology to describe licences. This paper documents the design of the ontology and its evaluation with respect to specific requirements gathered from stakeholders.

1 Introduction

Over the last decade there has been an explosion in the amount of multimedia content available online. This is due in part to the advent of Web 2.0, i.e. the availability of online tools that facilitate creating and sharing user-generated content. The change can also be attributed to growth in internet connectivity and bandwidth that permitted the progressive increase of quality in streamed multimedia content. Audio content is a fundamental part of the multimedia content consumed, as shown by the popularity of audio streaming services such as Spotify and SoundCloud.

Most of the online audio content is available also to software agents through web-based Application Programming Interfaces (APIs) developed by the maintainers of online repositories primarily to provide access to their content. This enables scenarios, still mostly unexplored, that go beyond simple consumption through maintainer-provided apps. Applications include highly customized user interfaces, seamless exploration of multiple repositories, advanced analysis of content-based on audio features, integration in creative workflows for transformation and reuse of sounds and music.

In order to facilitate the integration of the multiple existing repositories as wel as content consumption by software agents, we propose a common data model called the Audio Commons (AC) ontology. This paper describes the design process of the AC ontology and its first (and current) version, i.e. 1.0.0. The ontology is available online[1] with licence CC0.

Our ontology design is novel in many of its aspects: 1. it represents audio media in the broader context of audio production and sharing, going beyond media object-centric models like the EBU Core ontology or the W3C Ontology for Media Resources (e.g., including audio categories and collections as "first-class citizens"); 2. it employs a layered approach in which information can be represented at multiple levels of granularity (e.g., including optional details on how/when a content was recorded) associated with different perspectives (e.g., using a genre classification for music content or a sound effects taxonomy for sound effects); 3. it includes as a requirement support for an API from the developers' perspective, considering that this is a central aspect in the adoption of models nowadays.

The ontology is described following the MIRO (minimum information for the reporting of an ontology) guidelines [6]. All required information items are provided in the text. For reference, we will use the MIRO designations (e.g., C.3 for Communication), where the specific information item is provided[2]. Ontology need (B.1), its name (A.1), and its licence (A.3) have been already introduced.

Section 2 of the paper introduces the methodology and scope. Section 3 describes existing related models while Sect. 4 describes knowledge acquisition through a user survey. Section 5 details use cases and requirements, and Sect. 6 shows how existing ontologies are reused. The resulting ontology is described in Sect. 7 and evaluated in Sect. 8. Conclusions are discussed in Sect. 9.

2 Methodology and Scope

In order to frame the ontology design, this section describe more in detail its audience (B.3) and scope (C.1).

2.1 Methodology for Ontology Development

The ontology is developed and maintained by the Audio Commons consortium[3], composed of leading research institutes in sound and music computing and key players in the creative industries (A.2 and C.2). The development happens in an online public git repository on GitHub[4] (A.5). The GitHub issue tracking system associated with the repository will be used as communication channel for maintenance and future development of the ontology (C.3).

---

[1] https://w3id.org/ac-ontology/aco.

[2] The role of the information is conveyed anyway in the text, so the reader does not need necessarily to check the MIRO codes.

[3] http://www.audiocommons.org/team/.

[4] https://github.com/AudioCommons/ac-ontology.

Ontology design and development broadly follows the METHONTOL-OGY [2] methodological framework (A.6) that identifies six phases: 1. the *specification* i.e., the identification of the audience, scope, scenarios of use, and requirements (Sects. 2 and 5); 2. the *conceptualization* of an informal model (first paragraph of Sect. 7); 3. the *formalization* of the ontology in OWL [9] (Sect. 7); 4. the *integration* of existing ontologies (Sect. 6); 5. the *implementation* of the ontology with a JSON-LD OWL serialization.

METHONTOLOGY identifies also two *activities* carried on during the whole design process and orthogonal to the six phases: 1. *acquiring knowledge* through research of related ontologies and models (Sect. 3) and gathering data from potential users (Sect. 4), to inform multiple phases of the design process, mainly conceptualization and integration; 2. *documentation* of the process phases (internal) and the ontology specification (public).

2.2 Audience and Scope: Audio Commons Ecosystem

The role of the AC ontology is to offer a common data model enabling an ecosystem that integrates multiple online repositories and tools and allows agents to seamlessly explore, access, transform, and redistribute audio content, the Audio Commons Ecosystem (ACE) [3].

As a first step towards the ACE, a web API, the Audio Commons API, has been designed. This provides integrated access to a set of existing repositories (currently Freesound, Jamendo, and Europeana Sounds). Tailor made clients were developed that integrate with common tools in standard production workflows and use the AC API to access the repositories. This process validated the general idea of the ecosystem and informed the design of the ontology.

3   Related Ontologies and Data Models

This section describes related ontologies and data models (B.2). They have been gathered through the research of literature and online resources (D.1 and D.2) and evaluated as part of the design process (D.3).

In the 1990s, the International Federation of Library Associations and Institutions (IFLA) developed a conceptual model called **Functional Requirements for Bibliographic Records (FRBR)** [7]. FRBR defines four main entities to represent the products of intellectual or artistic endeavour: "*Work* (a distinct intellectual or artistic creation) and *expression* (the intellectual or artistic realization of a work) reflect intellectual or artistic content. *Manifestation* (the physical embodiment of an expression of a work) and *item* (a single exemplar of a manifestation) reflect physical form." [7] The entities of the FRBR model and their relationships have been later represented as an OWL ontology[5]. The model is relevant to the audio publishing domain, but the concepts are too generic by themselves. They need to be specialised to clarify the usage.

---

[5] http://purl.org/vocab/frbr/core#.

The **Music Ontology**[6] [14] aims to provide a comprehensive, yet easy to use and easily extended domain specific knowledge representation for describing music related information. It relies on, and extends the FRBR model, and provides an event based conceptualisation of music production workflows. The Music Ontology describes a domain that is very close to the one we model. Its terms are bound however to the music production workflow [1], without considering the broader, non-musical audio domain that includes e.g. natural sounds or field recordings with their own unique production model.

The European Broadcasting Union (EBU) developed the **EBU Core ontology**[7], which, among other things, specify how to describe properties of media files and formats. The EBU Core ontology has much broader scope, modelling other aspects of media handling. But its approach is centred on broadcast, hence most other entities cannot be easily reused outside of that domain.

The World Wide Web Consortium (W3C) developed an **Ontology for Media Resources**, which aims to integrate multiple metadata vocabularies in the context of media resources. This work is very interesting but the model presented is flat: it is a set of properties that can be attached to a single type of individual, the *MediaResource*. It is hence, as EBU Core, too limited to completely fulfil the requirements of the Audio Commons ontology.

In order to retrieve and explore repositories of audio content, it is useful to have some structured classification of audio. Several classifications have been developed both for manual and automatic categorisation of audio content. Some of them may be applied to audio [4,11] without restrictions, while others are specifically tailored for relevant subsets. Given the importance of musical audio, several classification deal with music, for example organising content by genre or musical instruments. These are usually taxonomies (i.e., simple hierarchical classifications) which may be represented in RDF through the Simple Knowledge Organization System (SKOS) [8].

4   Knowledge Acquisition

Given the use cases, audience, and basic requirements of the ontology, the Audio Commons consortium designed a survey to gather specific requirements from potential users.

The survey contained 24 questions (15 questions with predefined answers and 9 open ended questions) asking people working with audio content about various subjects. Besides demographics, we enquired about the workflows they use and metadata they would like to acess when searching for new audio content on the Web[8] (D.1). The first 8 survey questions assessed the context of audio content usage in the participants' work. Twelve questions (including 7 open ended questions) asked them to describe their ideal query interface, the attributes

---

[6] http://purl.org/ontology/mo/.

[7] https://www.ebu.ch/metadata/ontologies/ebucore/.

[8] The questionnaire is available online at https://goo.gl/forms/gWdzeHJuPIZhUwzD3.

they would use to query/filter audio content, and the frustrations faced using current tools. The last 4 questions gathered basic demographic information on the participants. The Audio Commons industry partners were given the task to ask their user base to fill in the survey (D.2).

4.1   Survey Results and Analysis

The survey had 661 responses[9]. Participants are split almost in half between professionals (45.5%) and amateurs (54.5%). 42.1% of the participants have more than 10 years of experience with audio technologies and 84.6% of the participants have at least 2 years of experience. The two main contexts of use are music production/performance (63.5%) and audio creation for either film and TV programmes or games (56.3%). There is a significant overlap between these two usages (26.6%).

Most of the participants work with an Internet connected device (84.4%) and get at least *sometimes* audio content from the web (84.8%). For the majority (52.7%), finding the right file is the most time consuming activity in their workflow. This is in strong contrast with the fact that most of the participants consider audio processing as the creative part of their work (65.6%). As for the types of audio content they look for, it is mostly sound effects (82.9%), but also audio loops (36.8%) and full songs (22.3%).

As an ideal way of retrieving audio content, most participants would use a web browser (67.7%), while a substntial part of them would prefer not to leave their digital audio workstation (DAW) software (41.1%). Regarding the query interface, most participants desire to search textually using keywords (86.5%). Half of them would find it useful to have keywords suggested through drop-down lists or similar (46.7%). Some of them would be interested in writing queries using natural languages (26.1%). A relatively small fraction of participants would like to use a full-fledged query language (16.5%) or a graphical interface (12.4%).

Most participants would like to use audio perceptual attributes like "Punchy", "Bright", or "Powerful" (71.4%) while many would also use musical attributes like key, tempo, or instrumentation (47.7%). The analysis of the open ended questions reveals a wide range of attributes used for audio search, ranging from musical properties (e.g., rhythm), to used hardware (e.g., equipment), to moods (e.g., happy).

The answers to the question on frustration show problems related with licensing (not clear enough, hard to understand the rules), syntax (problematic labelling of audio content), sparseness of metadata, lack of workflow integrations (easily retrieving the data into some part of the workflow), bad recording quality of audio sources, various interface problems (bad design, pop-ups, redirections, etc.) and lack of quality curation/recommendation.

---

[9] All the responses, along with the list of questions are publicly available online at https://doi.org/10.5281/zenodo.832644.

4.2 Conclusions

Answers to those questions allowed us to get the insight into how users would like to search for specific files and how such strategy would impact the design of the user interfaces. Some answers can guide the general proposed approach of the AC ontology while others inform the specific development strategy and content (D.3).

The expressed need for keyword-based search using descriptive text and a variety of attributes, alongside the perceived heterogeneity and sparseness of metadata, support the need of a common data model that unify how metadata is represented and facilitates spotting missing information in data sets.

The declared limits and idiosyncrasies of user interfaces or tools, make a case for having a common API (based on the ontology) that fosters the development of an ecosystem of tools while decoupling the tools from the audio repositories. The fact that most users work on internet connected devices in order, among other reasons, to access potentially "unlimited" audio content, mitigate the most-obvious drawback of a web API based architecture.

Regarding the structure of the ontology, the main takeaway is that audio categorisation should be flexible. The expressed desire to have a text-based search interface, possibly augmented with keyword suggestion/selection, and the variety of attributes used for search/filter would not be compatible with a simple monolithic centralized universal categorisation of audio content. To answer the desiderata the AC ontology need to support multiple categorisations instead.

Another important result of the survey is that there is a need for supporting specific subdomains of musical content associated with musical attributes, on top of supporting the more general domain of *audio content*, not necessarily musical. The significant overlap among the contexts in which musical and non-musical content is used confirms the argument for a comprehensive ontology.

5 Specification

Based on the scope and the survey results, the ontology design is framed by developing use cases and requirements.

5.1 Use Cases

Three user stories have been identified as highy relevant.

– As a café owner I would like to search for whole songs, which are free of any licensing fees. As an example, I would like to search via a browser for "Slow funk track without vocals". Once I found something I like, I would like to find tracks that play well together.
– As an audio producer, I would like to have access to high-quality audio loops from within my digital audio workstation (DAW). I want to search by instrument type, genre, key, tempo.
– As a game sound designer, I would like to have access to high-quality audio files from within my DAW. I want to search by effect type, mood, and perceptual features like "warm", "bright", etc.

5.2   Requirements

Using the analysis of the scope and the use cases the ontology designers identified
a set of requirements. They are represented as a list of example questions that the
ontology should be able to support answering, and a list of formal requirements.

Competency Questions.   The following sample questions are meant to be
asked with respect to a set of source repositories of audio content.

1. Which are the songs that are slow (tempo) funk (genre) tracks without vocals
   (instrumentation)?
2. What other tracks "play well" together with a given song in a playlist (e.g.,
   are in the same category according to some classification)?
3. Retrieve high-quality (sample rate, bits per sample) audio loops (type of audio
   content) for a given instrument type, genre, key, tempo.
4. Retrieve high-quality (sample rate, bits per sample) sound effects (type of
   audio content) for a given effect type, mood, and a set of perceptual features
   (e.g., warm and bright).

Formal Requirements.   The AC ontology should be able to ...

1. represent the concept of an audio clip as a piece of audio content published
   in a repository, alongside basic metadata (e.g., title, duration, licence);
2. describe attributes of the digital signal related to an audio clip (e.g., number
   of channels, sample rate);
3. describe attributes of the media file(s) related to an audio clip (e.g., media
   format, bit rate);
4. permit the classification of audio content along multiple axes (e.g., musical
   genre, mood, effect type);
5. represent the organisation of audio clips in collections (e.g., music albums,
   sound packs, search result sets);
6. optionally, describe additional details of the audio production/publishing pro-
   cess (e.g., where and when an audio clip was recorded).

6   Integration of Existing Ontologies

Following what is considered good practice, this section describes how external
vocabularies and ontologies are reused for the AC ontology (E.4). In some cases,
owing to discrepancies in the exact meaning or usage context, certain related
terms from other vocabularies could not be used directly. In these cases, in order
to promote interoperability, we tried to formally express the relationship between
new terms and existing ones in the new ontology. This is often expressed by defin-
ing the novel classes (properties) as superclasses (superproperties) or subclasses
(subproperties) of existing classes (properties). The structure of existing data
models also informed our own modelling decisions.

The FRBR concepts, while generic, are relevant to the present case, hence the AC ontology specialize them to the audio production and publication domain. The Music Ontology model is also relevant, namely when dealing with musical content. So the classes (or properties) of the AC ontology are often defined as subclasses (subproperties) of the corresponding classes (properties) in the FRBR ontology (version 2005-08-10) and as superclasses (superproperties) of their counterparts in the Music Ontology (revision 2.1.5). The FRBR model fulfils the role of an upper ontology for the AC ontology, so no general-purpose upper ontologies are used (E.8).

The EBU Core ontology (version 1.8) is used for the detailed formalization of media resources, their metadata (e.g., file size) and their formats (e.g., encoding format). As there is a formal mapping from part of the EBU Core ontology to the attributes defined in the W3C Ontology for Media Resources, the W3C vocabulary is indirectly supported too.

For the generic metadata items (title, description, depiction, ...) the Dublin Core Metadata Initiative (DCMI Metadata Terms[10], version 2012-06-14) and schema.org[11] (Version 3.3) vocabularies are used.

To manage the life-cycle of creative works, e.g. most published audio content, it is especially important to track licensing information, in order to know how a content may be used and redistributed. Dublin Core defines a simple model to attach licence information to a resource. This simple model however does not establish semantics for this licence information and hence does not support comparison and reasoning about properties (permissions, prohibitions, etc.) of licences. The AC ontology reuses the more detailed model specified in the Creative Commons Licensing ontology[12] (version 2017-11-17).

The production of certain entities in AC, for instance, the recording of a track or a sound, are temporal in nature and thus best described as events. The Event ontology[13] [13] (version 1.0) is used for this purpose. This ontology describes different aspects of temporal events, which could either be instantaneous or have a duration.

7   Ontology Description

This section introduces the Audio Commons ontology. Rather than providing a formal specification in this paper, we focus on practical and theoretical considerations in the design of the ontology. We contrast the Audio Commons ontology with other related ontologies and provide rationale for design decisions. The formal specification is provided as an on-line document using OWL (E.1). It can be accessed at https://w3id.org/ac-ontology/aco (A.4). The design is based on a layered approach in which entities are organised in three main groups (see Fig. 1): 1. the *content* of a repository, i.e. the physical *sounds*, the (digital) *signals*, the

---

[10] http://purl.org/dc/terms/.
[11] http://schema.org/.
[12] https://creativecommons.org/ns.
[13] http://purl.org/NET/c4dm/event.owl.

(published) *audio clips*, and the *audio files*; 2. the *events* associated with the entities and their transitions, i.e. *recording* or *synthesis* producing a signal and the *publication* of a signal as an audio clip; 3. the multiple *categorisations* that can be used to classify content.
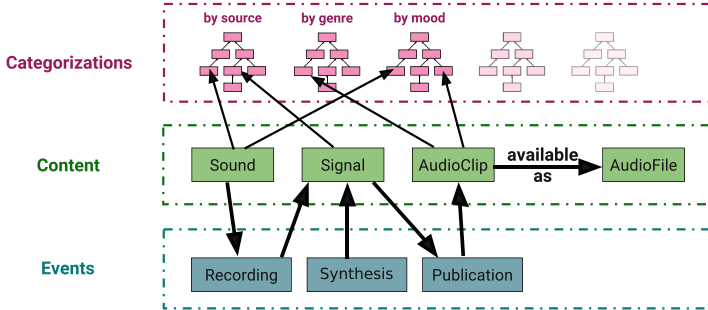

Fig. 1. A layered view of the Audio Commons ontology

Figure 2 shows the most general classes and properties of Audio Commons ontology and their relationship with elements of the FRBR and the Music ontologies. Following the FRBR model, the following three base classes have been defined: `ac:AudioExpression`, the specific intellectual or artistic form that a work takes each time it is "realized", in the audio domain (e.g., the recording or synthesis of music or sounds); `ac:AudioManifestation`, the physical embodiment of an audio expression (e.g., a musical track, a sound, an album); `ac:AudioItem`, a single exemplar of an audio manifestation (a copy of a CD or a specific media file).

The FRBR class Work, representing a distinct intellectual or artistic creation on a more conceptual level, has not been specialized in Audio Commons because this does not generalise sufficiently to all types of sounds relevant in the Audio Commons ecosystem. This concept is used to represent the common creation act in FRBR between different expressions, for instance, different drafts of a symphony, or its existence in the composer's mind at its most abstract level. For musical resources, the `mo:MusicalWork` class can still be used instead. An interesting crossing point is artistic conceptualisation for instance in *sound design* which we consider musical at this stage.

In the Music Ontology some specific properties (e.g., `mo:genre` and `mo:instrument`) are used orthogonally to classify both musical works, expressions, manifestations, and items, attaching them to specific instances of some classification schema (e.g., instances of `mo:Genre` and `mo:Instrument`). In the Audio Commons ontology these properties are generalised by the `ac:audioCategory` property that associates any audio expression or manifestation (or item, but the practical use of the latter case seems limited) to some generic `ac:AudioCategory`. Using this formalisation, different taxonomies specific to a domain of interest or a content provider can be "plugged in" and
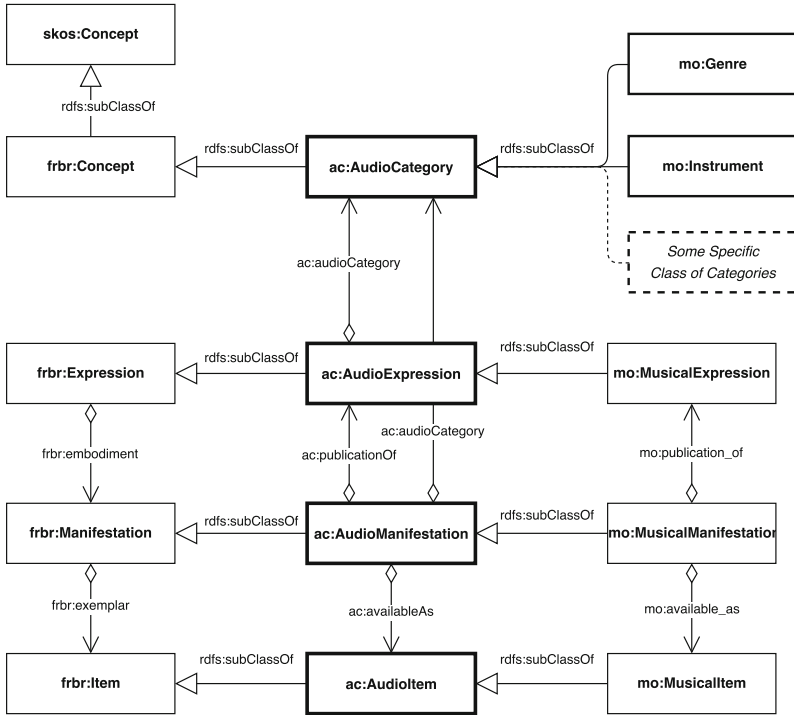
Fig. 2. Audio Commons ontology: a UML-like diagram of the top-level entities

matched to our core concepts enabling interoperability for generic tools, but retaining specificity required for expert users. Specific subclasses and properties related to audio expressions, manifestations, and items will be described in the rest of this section.

7.1 Audio Clips and Audio Collections

The class `ac:AudioManifestation` is a generalisation of (i.e., superclass of) a central entity in the Audio Commons ecosystem, `ac:AudioClip`. An instance of `ac:AudioClip` is any audio segment that has been published in some form or uploaded for consumption, for example, a track in a music label's repository or a sound in an audio repository, library or archive.

In order to represent collections of audio clips, the Audio Commons ontology offers an abstraction termed `ac:AudioCollection`, which is itself a subclass of `ac:AudioManifestation`. The content of each node of a collection is not limited to an `ac:AudioClip`, but may contain any `ac:AudioManifestation`. Collection can thus contain other collections to support specific cases, e.g. a mapping to the Music Ontology model where an `mo:Release` can contain multiple `mo:Record`(s) that can in turn contain multiple `mo:Track`(s).

The Dublin Core vocabulary is used for basic meta-data (e.g., title, description), while the Creative Commons licensing ontology is used for licensing information. Other information more specific to the domain is represented through audio-specific properties, which generalise music-specific properties defined in the Music Ontology: `ac:compiled` and `ac:published`, that associate an agent with manifestations he/she/it respectively created or published; `ac:homepage` and `ac:image`, that associate a manifestation with its page on a site (e.g., Jamendo) or with its depiction (e.g., the cover art of an album); `ac:duration`, that associate an audio clip with its duration (in milliseconds).

7.2   Audio Files and Signals

The class `ac:AudioItem` represents a concrete exemplar of an audio manifestation. In our domain, the main exemplars are the actual audio files. The corresponding class `ac:AudioFile` is a subclass of `ac:AudioItem`. To represent the information related to the audio file and its format, part of the EBU Core ontology is reused. The class `ac:AudioFile` is subclass of `ebu:MediaResource` too and the properties EBU Core properties having `ebu:MediaResource` as domain are used to describe the file (e.g., `ebu:hasEncodingFormat`, `ebu:fileSize`). The property `ac:availableAs` associates an `ac:AudioManifestation`with one or more corresponding `ac:AudioItem` instances.

While `ac:AudioFile` represents a concrete file encoded in a certain format, `ac:DigitalSignal` is the representation of the corresponding digital signal. `ac:DigitalSignal` is a subclass of `ac:MusicalExpression`. This conceptualisation was chosen because it pertains to the weakest ontological commitment with respect to how the signal is represented or encoded and where it is situated in a specific workflow. The data properties `ac:sampleRate`, `ac:bitsPerSample`, and `ac:channels`, associate a signal with its basic features specific to digital representations. The property `ac:publicationOf` can be used to associate an `ac:AudioClip` with the corresponding digital signal. The property `ac:encodes` instead, associates an `ac:AudioFile` with the encoded digital signal. The latter property works as a short-cut of traversing the inverse of `ac:availableAs` and `ac:publicationOf` which is introduced for representational convenience.

7.3   Audio Processes and Events

The description of temporal events is crucial to describe transitions in the workflow of audio production and publication. We thus extend the Event Ontology, offering subclasses of `event:Event` for specific actions that are interesting for the audio domain: `ac:SignalProduction`, the act of producing a `ac:Signal` (that could be either a `ac:AnalogSignal` or a `ac:DigitalSignal`), which is specialized by `ac:Recording` that represents the recording of a `ac:Sound` (e.g., the sound created by a musical band that is playing) and by `ac:Synthesis`; `ac:AudioPublication`, the event representing the public release of a piece of work (e.g., the release of a new album by a band). Using `event:Event`, details of the event such as its location in time and space, its factor, and its products

may be explicitly described. Moreover, the events can be composed using the property `event:sub_event`, to build complex events.

7.4   Audio Collections as Lists

The `ac:AudioCollection` entity provides a mechanism to describe collections of audio content in a way that is coherent and integrated with the rest of the Audio Commons ontology. However, the full serialisation of an instance of `ac:AudioCollection` is an explicit representation of a linked list and tends to be quite convoluted no matter what specific RDF syntax is used. For the Audio Commons ecosystem it is important to support usability by conveying information about instances concisely, so a simpler representation should be supported.

For the standard list class `rdf:List`, several RDF syntaxes provide ways to encode lists in a compact way. For this reason, as well as for interoperability reasons, the `ac:AudioCollection` can point to a `rdf:List` representation using the `ac:collectionAsList` property. The ontology constraints the usage of `rdf:first` and `rdf:rest` on the class `ac:AudioCollectionNode` (a member of an `ac:AudioCollection`) so that they "behave well" (e.g., they are functional) and are compatible with the formalisation of `ac:AudioCollection`. Instances of `ac:AudioCollection` can thus be represented either by using our formalism or by using standard RDF lists. They are formally equivalent and hence theoretically interoperable. In practice the transformation from one form to the other requires OWL-DL reasoning and it would not be always feasible or desirable. In that case it makes sense to chose one of the two options and run an ad hoc conversion if needed.

7.5   Evolution of the Ontology

Building an ontology that would encompass the whole audio domain (and all other domains connected with it) in all its complexity would be a very significant task that is beyond the scope of this work. The Audio Commons ontology is, for this reason, an implementation driven ontology evaluated and evolved in use. This means that the ontology will be growing depending on the demand for new services in the Audio Commons ecosystem (F.1). On the technical level, the last version of the ontology will always be accessible at the AC ontology URI, while past versions will accessible using an URI scheme including the version id (F.3). For reasons of backward compatibility, all the defined concepts will remain in the ontology and keep their current meaning. In case at some point the ontology maintainers decide that a concept is "not to be used any more", it will be annotated as *deprecated* (F.2).

8   Evaluation

We carried out an assessment of the ontology by using formal methods as well as checking its fitness for our domain and purposes.

8.1   Metrics and Formal Validation

The AC ontology defines in total 21 classes, 18 object properties (of which 5 functional), and 5 data properties (all of them functional). No individuals are defined (E.3). Every class and property defined has a textual description (`rdfs:comment`) and a label (`rdfs:label`), both are in English (E.7). For every property, domain and range are defined, except for three where only the range is defined, as they can be applied to individuals of a variety of types (`ac:homepage`, `ac:image`, and `ac:audioCategory`). For each entity defined in the ontology, the IRI is dereferenceable and leads via content negotiation either to an OWL syntax or a webpage documenting it (E.11).

The Audio Commons ontology has been checked for correctness, logical consistency, and alignment with established ontology design guidelines (G.1). The correctness of the ontology and its serializations has been checked first by loading it in the widely used ontology editor Protégé [10] and second through the VOWL copy[14] of the online validation service originally developed by the University of Manchester [5]. The logical consistency has been checked by running two reasoners, HermiT (version 1.3.8.413) [15] and FaCT++ (version 1.6.5) [17]. No inconsistencies have been found.

To validate the ontology with respect to existing good practices, we used the OntOlogy Pitfall Scanner! (OOPS!) online service [12]. This service, based on the existing relevant literature, checks for common pitfalls in ontology design. No pitfalls have been detected in the Audio Commons ontology.

8.2   Evaluation

The AC ontology is evaluated (G.2) by checking that it can (1) be used to reply to the competency questions described in Sect. 5.2, (2) fit in the current Audio Commons ecosystem, and (3) bring added value to it.

Answering Competency Questions. The questions can be formalised as queries from the data sources (the audio content repositories), for example using SPARQL (the standard query language for RDF). For simplicity and conciseness here the formalisation is described at a higher level, using just bits of SPARQL syntax for the graph patterns. Represented in the vocabulary defined by the AC ontology, all the competency questions consist in queries that get a set of individuals of type `ac:AudioClip`, say values of `?audioClip` where the triple `?audioClip rdf:type ac:AudioClip` exists. The set returned is determined by some filters that are applied on all the individuals available from the data source. Most filters can be represented as belonging to a certain category of a classification (mood, genre, instrumentation, ...), hence can be formalised as the existence of a triple `?audioClip ac:audioCategory <category1>`. Some filters (sample rate, bits per sample, ...) require to assess a numeric value. To require that the sample rate is certainly higher than 40 KHz, both triples `?audioClip ac:publicationOf ?signal` and

---

[14] http://visualdataweb.de/validator/validate.

`?signal ac:sampleRate ?sampleRate` have to exist and must satisfy `?sampleRate > 40E3`.

Fitting in the Audio Commons Ecosystem. The main application of the AC ontology is to provide a common way to represent multiple data models and APIs in the context of the AC ecosystem. As described in Sect. 2.2, a common API has already been defined in the context of the ecosystem. This is the AC API. It integrates multiple repositories by calling their specific APIs and is currently consumed by multiple client applications. The main endpoint of the API is the search endpoint[15] that offers search functionality on audio content that may be in any of the integrated services. Listing 1 is a sample response.

```
1  { "Jamendo": {"num_results": 0, "results": [] },
2    "Freesound": {
3      "num_results": 1,
4      "results": [
5        { "id": "Freesound:385285", "name": "01 - Cars across_city.wav",
6          "description": "Recorded in Moscow. Cars across on street.",
7          "duration": 127.321, "author": "semenov_nick",
8          "channels": 2, "bitdepth": 32, "samplerate": 48000.0,
9          "bitrate": 3001, "filesize": 48915012, "format": "wav",
10         "collection_url": "https://freesound.org/apiv2/packs/21409/",
11         "license_deed_url": "http://creativecommons.org/publicdomain/zero/1.0/",
12         "preview_url": "https://freesound.org/data/previews/385/385285_3596390-hq.ogg",
13         "tags": ["city", "cars"] }
14       ]
15     },
16     "Europeana": {"num_results": 0, "results": [] } }
```

Listing 1: An example JSON response from a search on the AC API

```
1  [ { "schema:actionStatus": "CompletedActionStatus", "schema:target": "services:Jamendo",
2      "schema:result": {"membersCount": 0, "collectionAsList": [] } },
3    { "schema:actionStatus": "CompletedActionStatus", "schema:target": "services:Freesound",
4      "schema:result": {
5        "membersCount": 1,
6        "collectionAsList": [
7          { "@id": "audioClips:Freesound/385285", "dc:title": "01 - Cars across_city.wav",
8            "dc:description": "Recorded in Moscow. Cars across on street.",
9            "duration": 127.321, "audioCategory": ["fstags:city", "fstags:cars"],
10           "isPublishedBy": "users:Freesound/semenov_nick",
11           "encodes": { "bitsPerSample": 31, "channels": 2, "sampleRate": 48000.0 },
12           "availableAs": {
13             "@id": "https://freesound.org/data/previews/170/170992_142024-hq.ogg",
14             "@type": "AudioFile", "ebu:hasEncodingFormat": "ebuformat:wav",
15             "ebu:bitRate": 3001, "ebu:fileSize": 48915012
16           },
17           "isMemberContentOf": "packs:Freesound/21409/",
18           "cc:license": "http://creativecommons.org/publicdomain/zero/1.0/" }
19         ]
20       } },
21    { "schema:actionStatus": "CompletedActionStatus", "schema:target": "services:Europeana",
22      "schema:result": {"membersCount": 0, "collectionAsList": [] } } ]
```

Listing 2: JSON-LD response from a search on the AC API

---

[15] https://m.audiocommons.org/api/v1/search/text/.

Listing 2 shows the output of the next version of the AC mediator. The same content is represented in RDF using AC ontology concepts and serialized as JSON-LD [16] (G.5). It can be seen that the new JSON-LD format is still close to the original JSON format. The need to map the properties to the ontology forces a slightly more structured JSON representation however that could also facilitate API documentation and other uses even without considering RDF interpretation. The associated JSON-LD context, not shown, maps the prefixes with the corresponding namespaces and set the AC ontology namespace as default. Detailed technical discussion of the mapping can be found in the wiki pages of the AC mediator, the software component exposing the AC API[16].

Bringing Added Value. While the general usage context is broader, it can be shown that there is already added value in just using the return format of the AC API as described above. A "semantic client" will not consume the new format as pure JSON; rather, it will use a JSON-LD processor to interpret the result set as an RDF graph. Moreover, responses to different searches may be merged in a richer RDF graph. The graph model and the unique identification of entities that are potentially repeated across results (e.g., audio categories, authors, media formats) enable organising or ordering the result set(s) in multiple ways according to the needs (e.g., group results by author). Furthermore, this information may be enriched by adding linked information from other sources (e.g., a music instruments taxonomy) or even creating new annotations as a local RDF graph. These functionalities, which are quite straightforward using semantic web technologies and the AC ontology, would need to be explicitly programmed if the "old" JSON output was used.

9   Conclusions

The Audio Commons ontology has been designed to model the audio content production and publishing domain. Its aim is to facilitate integration and serendipitous reuse of audio, through an ecosystem centred on this model and composed of multiple repositories and agents. The AC ontology is related to existing relevant ontologies and models. The evaluation shows that it is consistent, follows good practices, and is functional to the ecosystem. We are planning a test with users, based on client applications that make use of the ontology. Furthermore, as the ontology is disseminated and the ecosystem expands, more feedback is expected in the near future. These inputs will allow to evolve the ontology based on potentially unexpected use cases and conduct a more in-depth evaluation.

---

[16] https://github.com/AudioCommons/ac-mediator/wiki/JSON-LD-mapping.

References

1. Fazekas, G., Raimond, Y., Jakobson, K., Sandler, M.: An overview of Semantic Web activities in the OMRAS2 Project. J. New Music Res. Spec. Issue Music Inf. OMRAS2 Project **39**(4), 295–311 (2011)
2. Fernández-López, M., Gómez-Pérez, A., Juristo, N.: METHONTOLOGY: from ontological art towards ontological engineering. In: Proceedings of the Ontological Engineering AAAI-97 Spring Symposium Series. American Asociation for Artificial Intelligence (1997)
3. Font, F., et al.: Audio commons: Bringing creative commons audio content to the creative industries. In: Audio Engineering Society Conference: 61st International Conference: Audio for Games, February 2016. http://www.aes.org/e-lib/browse.cfm?elib=18093
4. Gemmeke, J.F., et al.: Audio Set: an ontology and human-labeled dataset for audio events. In: Proceedings of the IEEE ICASSP 2017 (2017)
5. Horridge, M., Bechhofer, S.: The OWL API: a Java API for working with OWL 2 ontologies. In: Proceedings of the OWLED 2009, vol. 529, pp. 49–58. CEUR-WS.org (2009)
6. Matentzoglu, N., Malone, J., Mungall, C., Stevens, R.: Miro: guidelines for minimum information for the reporting of an ontology. J. Biomed. Semant. **9**(1), 6 (2018). https://doi.org/10.1186/s13326-017-0172-7
7. McBride, B.: Functional Requirements for Bibliographic Records, Final Report. UBCIM Publications, New Series 19 (1998)
8. Miles, A., Bechhofer, S.: SKOS Simple Knowledge Organization System - Reference. https://www.w3.org/TR/skos-reference/
9. Motik, B., et al.: OWL 2 Web Ontology Language: Structural Specification and Functional-Style Syntax, 2nd edn. W3C REC 11 December 2012 (2012)
10. Musen, M.A., the ProtégéTeam: The Protégé project: a look back and a look forward. AI Matt. **1**(4), 4–12 (2015)
11. Nakatani, T., Okuno, H.G.: Sound ontology for computational auditory scence analysis. In: AAAI/IAAI, pp. 1004–1010 (1998)
12. Poveda-Villalón, M., Gómez-Pérez, A., Suárez-Figueroa, M.C.: Oops!(ontology pitfall scanner!): an on-line tool for ontology evaluation. Int. J. Semant. Web Inform. Syst. (IJSWIS) **10**(2), 7–34 (2014)
13. Raimond, Y., Abdallah, S.: The event ontology. Technical report (2007). http://motools.sourceforge.net/event (2007)
14. Raimond, Y., Abdallah, S.A., Sandler, M.B., Giasson, F.: The music ontology. In: ISMIR, vol. 422, Vienna, Austria (2007)
15. Shearer, R., Motik, B., Horrocks, I.: HermiT: A Highly-Efficient OWL Reasoner. In: Proceedings of the OWLED 2008, vol. 432, p. 91 (2008)
16. Sporny, M., Longley, D., Kellogg, G., Lanthaler, M., Lindström, N.: JSON-LD 1.0: a JSON-based Serialization for Linked Data. W3C Recommendation 16 January 2014 (2014)
17. Tsarkov, D., Horrocks, I.: Fact++ description logic reasoner: System description. Automated reasoning, pp. 292–297 (2006)