Int J Digit Libr (2019) 20:25–47
https://doi.org/10.1007/s00799-017-0223-9

On providing semantic alignment and unified access to music library metadata

David M. Weigl[1] · David Lewis[2] · Tim Crawford[2] · Ian Knopke[3] · Kevin R. Page[1]

Received: 1 March 2016 / Revised: 27 June 2017 / Accepted: 30 June 2017 / Published online: 28 August 2017

Abstract A variety of digital data sources—including institutional and formal digital libraries, crowd-sourced community resources, and data feeds provided by media organisations such as the BBC—expose information of musicological interest, describing works, composers, performers, and wider historical and cultural contexts. Aggregated access across such datasets is desirable as these sources provide complementary information on shared real-world entities. Where datasets do not share identifiers, an alignment process is required, but this process is fraught with ambiguity and difficult to automate, whereas manual alignment may be time-consuming and error-prone. We address this problem through the application of a Linked Data model and framework to assist domain experts in this process. Candidate alignment suggestions are generated automatically based on textual and on contextual similarity. The latter is determined according to user-configurable weighted graph traversals. Match decisions confirming or disputing the candidate suggestions are obtained in conjunction with user insight and expertise. These decisions are integrated into the knowledge base, enabling further iterative alignment, and simplifying the creation of unified viewing interfaces. Provenance of the musicologist's judgement is captured and published, supporting scholarly discourse and counter-proposals. We present our implementation and evaluation of this framework, conducting a user study with eight musicologists. We further demonstrate the value of our approach through a case study providing aligned access to catalogue metadata and digitised score images from the British Library and other sources, and broadcast data from the BBC Radio 3 Early Music Show.

Keywords Linked Data · Metadata · Semantic alignment · Contextual matching · Musicology · Early music

1 Introduction

The reconciliation of corpora providing access to historical catalogue data in a digital libraries context is made difficult by a range of challenges from ambiguities concerning the names of individuals to disputed or erroneous attribution (e.g. [1]). Relevant sources include digital libraries of institutions such as the British Library, formal digital library resources provided by organisations such as the OCLC[1] (e.g. VIAF[2]), data feeds provided by commercial and media industry institutions such as the BBC (the UK's national public service broadcaster), and community resources such as MusicBrainz.[3] Such datasets provide complementary information concerning the same historical entities (e.g. composers, works), but the corresponding records may not share identifiers across the datasets. This greatly complicates convenient access to the entirety of the information available on a given entity.

Automated approaches employing heuristics to identify similarly named individuals or similarly titled works are insufficient to resolve ambiguous cases such as a name being shared by multiple individuals or an individual being known by multiple names. Further complications arise from

✉ David M. Weigl
david.weigl@oerc.ox.ac.uk

1 Oxford e-Research Centre, University of Oxford, Oxford, UK

2 Department of Computing, Goldsmiths, University of London, London, UK

3 British Broadcasting Corporation, London, UK

[1] http://www.oclc.org.

[2] http://viaf.org.

[3] http://musicbrainz.org.

potential differences in language between datasets (e.g. via anglicisation of Latin names), non-standardised spellings (a prominent issue with historical catalogue data), and simple errors. The manual resolution of these concerns by domain specialists is tedious and error-prone, as the number of potential alignment candidates grows exponentially with the size of the datasets in question.

In this paper, we address this issue by combining the expert guidance of the domain specialist with the efficiency and tractability of an automated solution, combining surface similarity and contextual semantics to generate match candidates for confirmation or disputation by the musicologist. By providing computational *assistance*, we ensure that the user's attention is required only for those aspects of the alignment task providing or demanding insight. By tracking provenance information regarding the user's alignment activities, we explicitly capture their judgement on contested attributions and similar topics of scholarly dispute.

We adopt a Linked Data approach [5] employing the Resource Description Framework (RDF) [16], a standard model for online data exchange. Linked Data extends the linking structure of the World Wide Web by employing URIs to specify directed relationships between data instances. These data instances may themselves be encoded by URIs or represented by literal values. A set of two such instances, linked by such a relationship, is referred to as a *triple*.[4] Collections of triples may be stored in flat files on a server, accessed via HTTP, or housed within specialised RDF databases, known as *triplestores*. By employing this Linked Data approach, the meanings of the relationships between the data are made explicit, allowing them to be understood by both humans and machines. Information represented in this way may be linked to external datasets and can in turn be linked to from external datasets, embedding the information within a wider web of knowledge and making it discoverable and reusable in other contexts [7].

A key strength of the RDF model compared to the traditional relational model is that it is robust to underspecification and mutability in the data schema, and thus facilitates the merging of disparate datasets that may each employ radically different schemas or ontologies. Since it involves publishing alignment outcomes as interlinked RDF triples, our solution facilitates the creation of combined views of the unified data within the now-reconciled corpora, while supporting reuse of the musicologist's decisions to drive further scholarly activity.

The remainder of this article is organised as follows: Sect. 2 describes the musicological motivations that have led to the development of the work presented here and introduces a specific motivating case in early music. Section 3 details related work in the matching of disparate datasets. Section 4 presents the design of the data model and framework underlying our approach towards addressing this problem. Section 5 discusses the implementation of a Semantic Alignment and Linking Tool (SALT)[5] that builds on this model. Section 6 reports on a case study applying SALT to link academic and media industry datasets focusing on our early music scenario. Section 7 presents a user evaluation of the SALT data model and user interface, employing eight musicologist participants with expertise in early music. Section 8 presents the implementation of a unified viewing interface[6] providing access to datasets linked by a domain specialist using SALT. Finally, Sect. 9 concludes this discussion and presents our plans for future work.

2 Musicological motivation and case study in early music

Exploratory investigations have been conducted into the information needs and information seeking behaviours of (potential) users of music information systems [24,42], and to a lesser degree, of musicologists [2,19,28]. However, research in the field of Music Information Retrieval has predominantly focused on system-centric concerns [42], and the needs and behaviours of musicologists in particular remain relatively underexplored [19].

In the course of our work, we have observed that musicologists gather the information they require about music and its contexts from a variety of sources. An academic may find literature through tools including Répertoire International de Littérature Musicale (RILM)[7] and JSTOR,[8] through general-purpose resources such as Wikipedia or Google Scholar, or using topic-specific encyclopaedias, for example *The New Grove Dictionary of Music and Musicians (New Grove)* [36], and *Musik in Geschichte und Gegenwart* [8], and through library catalogues. Resources about notated music are more diverse, including *Répertoire International des Sources Musicales (RISM)* [34] and more specialised catalogues such as Brown's *Instrumental Music Printed Before 1600: A Biography* [10], and even more varied still are the sources of information on performances, recordings, and broadcasts.

Historically, the task of coordinated study of these areas of musical understanding has necessarily been a manual one. Many of the resources are or have been published in book or serial form, and the reader can carefully assemble a narrative from the separate parts. Online versions of these, along with the new digital-only materials, provide opportunities

[4] In the parlance of Linked Data, a *subject* is related by a *predicate* to an *object*.

[5] http://github.com/oerc-music/salt.

[6] http://github.com/oerc-music/slobr-ui.

[7] http://rilm.org.

[8] http://jstor.org.

for faster, more comprehensive study, along with larger-scale apprehension of a topic, but they also carry greater risks of misinterpretation.

Musicologists are aware of this tension between physical and digital methods. They are frustrated when the materials they seek are not available in physical libraries [28], and acknowledge the benefits of the digital in terms of breadth and immediacy of access, and of increased workflow efficiency, while recognising the difficulties inherent in the excessively abundant information, making it difficult to "separate the wheat from the chaff" [19].

The ability to navigate easily between large stores of musicological knowledge is immensely valuable, provided the navigation is reliable. Even where it is occasionally incorrect, perhaps mistakenly linking the twentieth-century John Tavener with the sixteenth-century John Taverner, these errors are usually easily spotted and ignored. Where such connections are used as part of an automated process that summarises a large amount of gathered information, errors are harder to spot and can make nonsense of the results. In a commercial or otherwise public-facing environment, such mistakes can affect user confidence in the data at large.

Our motivation is the belief that a combined—or rather, linked—dataset better serves the research needs of musicologists and musicians, as well as acting as a means for enriching the experience of a general user, for example exploring music on a broadcaster's website with supplemental insight generated *by* the musicologist. We are clearly not alone in this belief. The implicit linking provided by library authority files is progressively being turned into explicit linking through VIAF [4] in ever more bibliographic resources, the RISM dataset is now published as Linked Data, and broadcasters such as the BBC publish some of their musical information with MusicBrainz artist links.

2.1 Use case

The work presented here employs, as a use case, an early music topic involving the combination of datasets generated in academic projects and in the media industry. Both sets of stakeholders—academic and industrial—are intended beneficiaries of the technology rather than merely providers of data.

The Semantic Linking of Information, Content, Knowledge and Metadata in Early Music (SLICKMEM) project [13] produced a Linked Data resource combining the Early Music Online (EMO) dataset [35] of digitised images and metadata on a collection of music books from the British Library, with the machine-readable Electronic Corpus of Lute Music (ECOLM) [14].

Radio 3 is the music and arts station of the BBC. The station broadcasts The Early Music Show (EMS) weekly, as an "exploration of early music, looking at early developments in musical performance and composition both in Britain and abroad". It plays almost exclusively European Classical music from the eighteenth century and earlier. Like all regular BBC programmes, EMS has a dedicated area on the broadcaster's website[9] with clips, podcasts, and supporting information about current and past editions. The BBC exposes structured broadcast data about its programmes, including EMS, encompassing a list of featured works for each episode, and information on the contributors associated with these works (performers, composers, singers, arrangers, etc).

SLICKMEM resources intersect with a subset of the repertory broadcast on EMS, both in historical period and genre, with EMO consisting of primarily sixteenth-century vocal music and ECOLM featuring music for the solo lute extending into the eighteenth century. Through the Creative-Commons licensed SLICKMEM resources, 15,485 pages of digitised score representing 2756 works by over 400 composers are available.

None of these data sources are born as Linked Data publications. In the case of the EMS data, basic alignment and export to RDF is performed by translating programme data exposed as JSON by the BBC into RDF using JSON-LD [39], a simple extension of the JSON format that enables the incorporation of semantic context within the more widely familiar JSON syntax. In the case of EMO, the dataset is a version of the British Library catalogue further enriched as part of the EMO project itself. As a part of the library's catalogue, the data are represented in MARC [40]. This means that while associating information with books is reasonably well catered for, the same is not true for musical items contained in them, and the relationship between composers and arrangers and the precise musical items they worked on is seldom clearly specified. Since the first phase of EMO concentrated on books with multiple composers represented, this is a more significant issue than it might otherwise have been.

ECOLM uses a bespoke relational data model for its catalogue, implemented with a MySQL database. Decisions about authority lists were made by project members at launch and so, for example, although multiple names for personal entries are permitted, the primary name form and spelling is taken as authoritative from the *New Grove* [36]. The intricate model is carefully designed to avoid misrepresenting subtle historical data—and uncertainty and provenance associated with it—so harmonising it with other models is far from trivial.

2.2 Problem statement

In linking collections such as those from the BBC, the British Library, and musicologists, we aim to maintain the separation

[9] http://www.bbc.co.uk/programmes/b006tn49.

of concerns between the actors providing the data—the BBC, the British Library, and the musicologist—recognising that each will have distinct requirements, while still reaping the benefits of intersecting interests that are manifest by the links between the corpora.

To do so requires us to semantically align the EMS programme data published by the BBC with data available through SLICKMEM. Further, we integrate links to complementary datasets from sources including LinkedBrainz[10] [20] and DBPedia[11] [25], projects that publish structured content, extracted respectively from the open online music database MusicBrainz, and from Wikipedia, as Linked Data. It is difficult to automate the alignment process as each dataset uses its own distinct unique identifiers to address particular data instances. Nevertheless, the datasets overlap significantly in terms of describing the same real-world entities, such as particular composers of early music and their works. Valuable alignment cues are provided by indicators such as the similarity of textual labels (e.g. composer name or work titles) or shared contextual information (e.g. birth place or publication date), but can be too ambiguous to be reliable without manual verification. Further, they may fail to capture valid matches, e.g. when the same composer is known by two different names. A knowledgeable musicologist is able to resolve many of these issues by drawing on personal domain expertise, but manually aligning datasets with thousands or tens of thousands of entities is likely to be prohibitively time-consuming.

The act of alignment is made more difficult—and reliant on domain expertise—by the nature of the information the historical music catalogues are modelling. Not all people named in the catalogue may have entries, even in the authority lists used by the originating organisation. The information available is often insufficient to disambiguate between candidate matches. An example is music by both Domenico and Alfonso Ferrabosco[12] in SLICKMEM, often with the simple attribution string 'Ferrabosco' (variously spelled). With the piece titles and a list of works by each composer, these can be disambiguated given sufficient domain knowledge, but not without. In cases where a work is untitled, anonymous or both, disambiguation is impossible without access to the music in some form. More difficult still are works for which attribution is disputed or erroneous, i.e. works subject to ongoing scholarly disagreement.

The approach presented here makes use of the contextual and string similarity-based alignment cues discussed above in order to generate candidate match suggestions for confirmation or disputation by a musicologist user. In doing so, we simultaneously address the issues of ambiguity and the lack of reliability inherent in a fully automated approach, while minimising the musicologist's workload to require interaction only where human insight is required, ensuring that the alignment task remains tractable.

3 Related work

Identification of multiple data instances corresponding to shared real-world entities has a long history of research in the literature on relational databases, where related issues are framed in a wide variety of terms including *deduplication*, *record linkage*, *instance identification*, *coreference resolution*, and *reference reconciliation* [18]. Related instances are typically detected using heuristics operating on the similarity of strings (so-called "fuzzy matching") contained within the fields of the records in question; this similarity may be determined by various means [18], including variations of Levenshtein edit distance [27] or phonetic similarity (e.g. via the metaphone family of search algorithms [33]). Where differences in data schema must be overcome, this can be achieved by using shared instance values as cues that differently named fields may refer to the same kinds of entities [26].

The problem is no less widespread in the Linked Data world. Instance matching here is complicated by the high degree of schema variability between data sources, and the potentially widely distributed nature of related data instances contained within these sources (see reviews in [11,37,38]). A common approach is to focus on alignment of ontologies, rather than individual instances. In their recent review paper, Shvaiko and Euzenat [38] presented a comprehensive discussion of the challenges and applications of *ontology matching*. Such approaches may make use of a number of different techniques, including matches based on shared terminology (i.e. string similarity of predicate labels), and on structural similarity (based on *is-a* or *part-of* hierarchies relative to already matched concepts).

One of the challenges outlined is that of designing ways for users to be involved in the matching process without becoming lost in the huge number of results inherent in the merging of large datasets. The need for matching tools to be user-configurable and customisable is emphasised. Shvaiko and Euzenat note that existing matching tools tend to lack graphical user interfaces and emphasise the utility of enabling the

[10] http://linkedbrainz.org.

[11] http://www.dbpedia.org.

[12] The Ferrabosco family of musicians and composers included three Alfonso Ferraboscos, along with a Domenico, Henry and John. All but Domenico were active in the English court, held similar employment and wrote in similar musical genres. The Alfonso Ferraboscos are now usually disambiguated using Roman numerals, but are seldom distinguished textually in contemporary documents. One rare exception is the British Library manuscript Add. 29427, which contains works by both Alfonso Ferrabosco II and III, where attribution is clarified by the epithets 'senior' and 'junior'.

customisation and configuration of such tools by users who are not ontology matching specialists.

Tools that do exhibit graphical interfaces for user interaction include the System for Aligning and Merging Biomedical Ontologies [23], a matching tool developed at Linköping University, Sweden; AgreementMaker [15], a general-purpose ontology matching system developed at the University of Illinois at Chicago; and the Silk linking framework [41] developed at the University of Mannheim, Germany. This framework comprises a link discovery engine determining matches based on predefined heuristic rules, a user interaction component enabling rapid evaluation of matching outcomes and tweaking of heuristics in order to improve results, and a protocol for maintaining links in conditions of potentially mutable data.

Each of these tools is capable of matching based on measures of string distance and structural considerations of data schemas: the Linköping tool makes structural alignment recommendations by considering class and sub-class relationship hierarchies relative to previously matched concepts, making use of biomedical domain-specific knowledge bases; AgreementMaker propagates similarity measures determined for ancestors and siblings in the hierarchy; and Silk provides measures of taxonomic distance, as well as a bespoke selector language that allows the description of arbitrary structural relationships. Each tool combines the outcomes from these measures in order to determine final alignment proposals, based on some notion of relative weighting.

The Linköping tool creates one-to-one alignments between concepts and relations, whereas AgreementMaker is also capable of generating one-to-many and many-to-many alignments on the schema level. Both tools use string similarity between instance labels to inform higher-level schema alignment, but neither is targeted at the creation of links between data instances (*instance matching*). Silk is concerned with the creation of links at the instance level, but takes a 'top-down' approach; the user interacts with the system to calibrate heuristic rules on the schema level until the resulting instance match outcomes are deemed acceptable.

A further tool situated broadly within our area of interest is OpenRefine,[13] an open-source tool formerly known as Google Refine. This tool provides a user interface supporting the exploration, tidying, and reconciliation of large datasets, with a focus on creating new datasets derived from the original source data; in contrast, alignment tools generate auxilliary metadata linked to the original datasets in a hyperstructure. The OpenRefine interface enables the user to merge entities deemed to be identical, based on entity value matching; the procedure does not make use of contextual cues.

In the domain of digital musicology, the MusicNet tool [9] takes a 'bottom-up' approach, allowing the user to create matches through interactions on the instance level. However, the tool only assists the user by generating alignment cues based on string similarity; the underlying schematic structure of the data is not taken into account.

The evaluation of interactive tools has received increasing interest in the ontology alignment community in recent years, with an *interactive matching evaluation* track running as part of the annual Ontology Alignment Evaluation Initiative[14] campaign since 2013. Paulheim et al. [32] detailed the evaluation strategy involved in this track, outlining quality measures including *generic cost* per user action, which is defined according to specific task context (e.g. time consumed, number of interactions required, or the money paid to the domain expert user); and the *F-measure*, the traditional measure of classification accuracy corresponding to the harmonic mean of *precision* and *recall*. Paulheim et al. noted that while it is relatively easy to optimise for either the cost measure (by relying on a fully automated alignment solution) or the F-measure (by making the domain expert perform all the work manually), achieving a reasonable trade-off between the two is more challenging.

As discussed in Sect. 2.1, portions of the source data factoring into the use case presented in this article are rooted in the library catalogue, comprising metadata in the MarcXML format, a serialisation of standard MARC records. The use of RDF to supplement—or even replace—catalogue records with bibliographic ontologies remains a topic of active research and ongoing discussion both in libraries [22] and in the digital humanities [30]. Available ontologies include BIBFRAME,[15] a conceptual bibliographic description model; RDF ontologies expressing the Metadata Object and Metadata Authority Description Standards (MODS/RDF[16] and MADS/RDF[17]), as well as the FRBR-aligned Bibliographic Ontology (FaBiO),[18] among others [21].

As we will see in Sect. 4.1, the approach presented in this article simply requires data to be expressible as RDF, remaining agnostic as to ontological and vocabulary choices aside from a few very basic requirements. This maximises the applicability of our tool to a broad range of datasets; however, careful consideration should be given to appropriate data modelling choices, for example employing the ontologies listed above, if the datasets are to be adopted for use with digital library systems.

[13] http://openrefine.org.

[14] http://oaei.ontologymatching.org.

[15] http://www.loc.gov/bibframe.

[16] http://www.loc.gov/standards/mods/rdf.

[17] http://www.loc.gov/standards/mads/rdf.

[18] http://vocab.ox.ac.uk/fabio.

4 Data model and framework

4.1 Linked Data compatibility

The principle aim of the work reported here was to design a model and framework supporting domain experts in the semantic alignment of complementary datasets through linking structures published as Linked Data. These published outcomes may then form the semantic scaffolding for a unified view of the data. Our design stipulates minimal requirements upon the datasets to be aligned, permitting the framework to be cross-applicable to corpora from a variety of domains. These requirements are:

1. The data can be expressed as RDF triples.
2. Each entity in the data that is subject to alignment decisions is addressable using a persistent URI.
3. Each such entity exposes a human-comprehensible label.
4. It is possible for entities to be linked to additional sources of contextual information.

The first requirement relates to the data model underlying our design. It should be noted that the data does not have to be expressed as RDF at source; it is relatively trivial to convert legacy data stored in tabular spreadsheets or relational databases into an RDF format using open-source tools [29]. Examples include the D2RQ platform[19] [6] that produces semantic structures by mapping from a particular derivative of the relational structure describing the tables in a database, and Web-Karma,[20] a tool that supports the interactive definition of a graph of relationships between the columns of a tabular dataset, using a graphical user interface.

The second requirement relates to the mechanism by which the alignment decisions of the domain expert—confirmations or disputations of a match between pairs of specific entities across two datasets—are asserted and stored. A persistent URI that uniquely identifies each entity is required in order to serve as a handle to which data representing individual alignment decisions may be attached. Queries and browsing interfaces that make use of the outcomes of the alignment process are then able to address specific entities on either side of the dataset divide, and may easily discover all matches that have been asserted across the gap (Sect. 4.5).

The third requirement is necessary in order to make the system useful to human users. Assigning comprehensible labels to all specific data entities—typically by asserting triples encoding `rdfs:label` relationships—is considered good Linked Data practice [17] and in this case is required in order to give the user an indication of the specific data instances available. While the system currently focuses on textual labels, multimedia and multimodal applications may be envisioned for future development (Sect. 9).

Finally, while recommendations on potential alignment candidates can be made based on surface similarity of the labels of the entities to be compared (i.e. string distance), these entities may be linked to additional sources of contextual information in order to make profound use of the underlying semantic capabilities of the system. All that is required is that some graph relationship can be described between the entities on either side of the alignment, and the same, shared contextual item. Each such contextual item represents a potential alignment point, upon which a match of two entities from either side may be suggested for user confirmation or disputation. The exact schematic relationship between the entity and the contextual item may take an entirely different form within either dataset. As an illustrative example drawn from our case study in early music, the SLICKMEM data encodes people both in their capacity as composers of works, and as authors of books that compile works. While the relationship of:

```
composer composes work
```

is schematically different from:

```
author creates book;
book contains work,
```

we can, nevertheless, connect `composer` and `author` instances via `work` using these relationships, in order to aid the alignment task (Fig. 1).

4.2 Datasets and saltsets

It is a strength of RDF that the same dataset may incorporate diverse types of entities. However, in order to simplify the alignment task, it is desirable to present entities of a consistent type with direct user relevance for the given task—the focal points that will anchor one side of an alignment decision. Referring again to our use case, the SLICKMEM dataset encodes entities representing people, works, books, publications, and places and relates them within a shared graph structure; however, in terms of the alignment task, it is more useful to operate on sub-graphs relating directly to the entities that act as alignment anchors, e.g. SLICKMEM composers, SLICKMEM authors, or SLICKMEM works. By virtue of their different positions within the overall graph structure of the dataset, different types of entities also have distinct schematic relationships with contextual items of interest.

To disambiguate between the shared graph structure incorporating all data from a particular source and the sub-graphs of this structure that are directly relevant to a given alignment task, we refer to the former as *dataset* and to the latter as *saltset*. A saltset is thus a subset of a dataset, consisting specifically of those entities (and their labels) that will form the focal points of an alignment task, combined with a collection of templates that define potential contextual alignment

[19] http://d2rq.org.

[20] http://usc-isi-i2.github.io/karma.

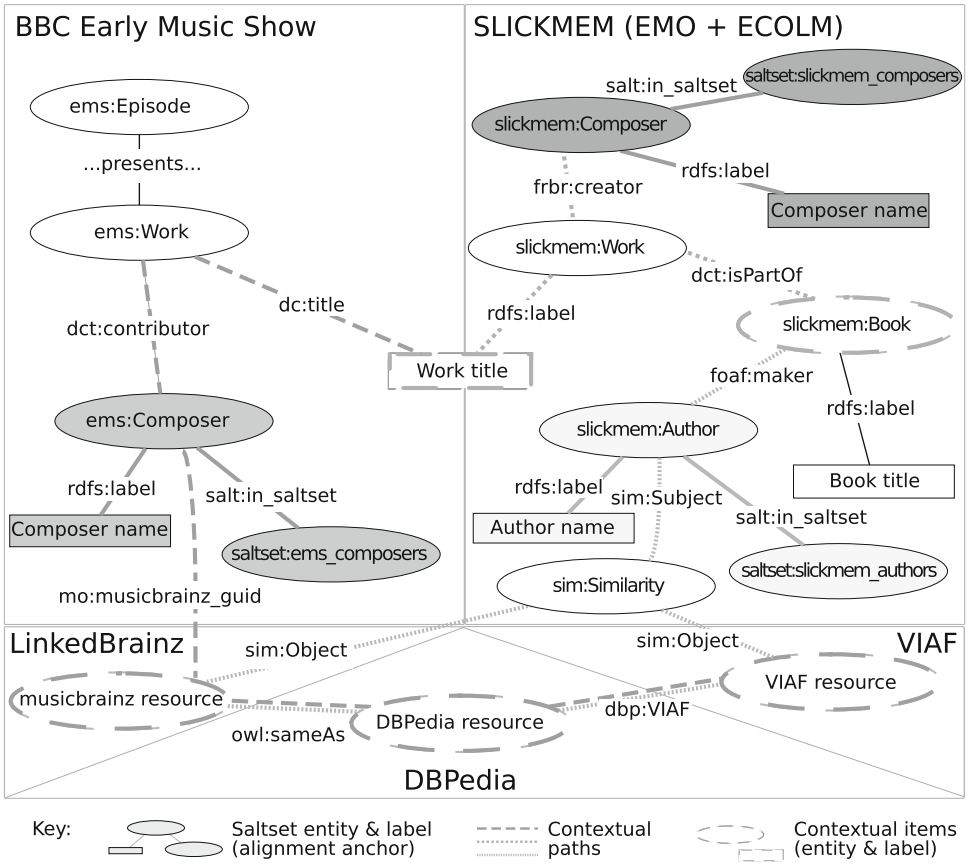

Fig. 1 A saltset configuration used in the alignment of programme data from the BBC Radio 3 Early Music Show with metadata on early music from the British Library and the Electronic Corpus of Lute Music, comprising five source datasets: BBC EMS broadcast data, SLICKMEM, LinkedBrainz, DBPedia, & VIAF

points. Each such point is an entity in the dataset, a *contextual item* which stands in some defined relation (a *contextual path*) to such an anchoring entity.

4.3 Contextual paths

A contextual path is specified as a graph traversal—a walk through the graph structure—that begins at the focal point of a given saltset, and ends at a particular graph node expected to offer significant alignment cues to the user. These nodes of significance—the contextual items associated with a particular saltset—are selected by the domain expert during the configuration of the tool (Sect. 5.3). Examples are illustrated in Fig. 1, where contextual paths are represented as patterned lines according to their associated saltset.

In a particular alignment task, any contextual items associated with saltsets on either side of the alignment gap form explicit connections between the focal entities of the saltsets being compared. These contextual items are made accessible to the user in order to serve as cues to alignment decisions. By associating weights with shared contextual items in a given alignment task, a score is calculated to provide the user with a view of potential alignment candidates sorted according to contextual relevance. The magnitudes of the associated weights are determined by the user. This allows for fine-grained differentiation of significance between the association of particular alignment anchors with different contextual items. For instance, consider a scenario where item A is of small but non-negligible interest in terms of offering identification cues for the alignment of two particular saltsets; item B may be of somewhat greater interest, but item C, perhaps an identifier in an external authority file, may trump the presence of both items A and B combined. This situation is easily accommodated by simply ensuring that item B has a greater weighting than item A, and that item C has a greater weighting than the aggregated weightings of items A and B.

The alignment process is iterative in nature. In constructing matches between entities, changes in the significance of contextual items may become apparent, and new nodes of contextual significance can be discovered as a corpus of match decisions takes shape. This may prompt the user to reconfigure the contextual paths and weightings associated with the saltsets, which may in turn support the discovery of further match instances.

4.4 Match decisions

User match decisions are represented by RDF triples stored in a dedicated area (or named graph) within the triplestore.

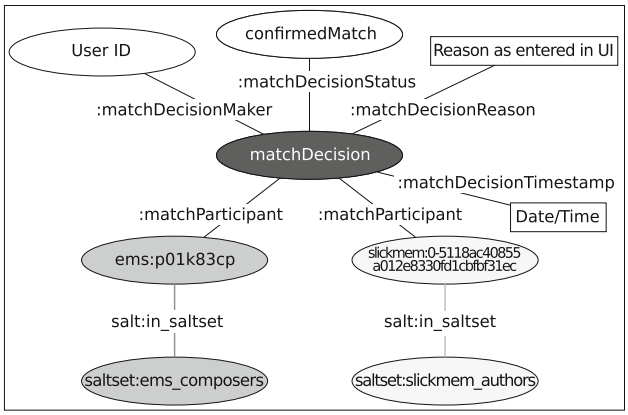

Fig. 2 Structure of match decisions produced by SALT to reflect decisions made by a user. All match decisions produced by a given user are published to a dedicated named graph specific to that user in the triplestore

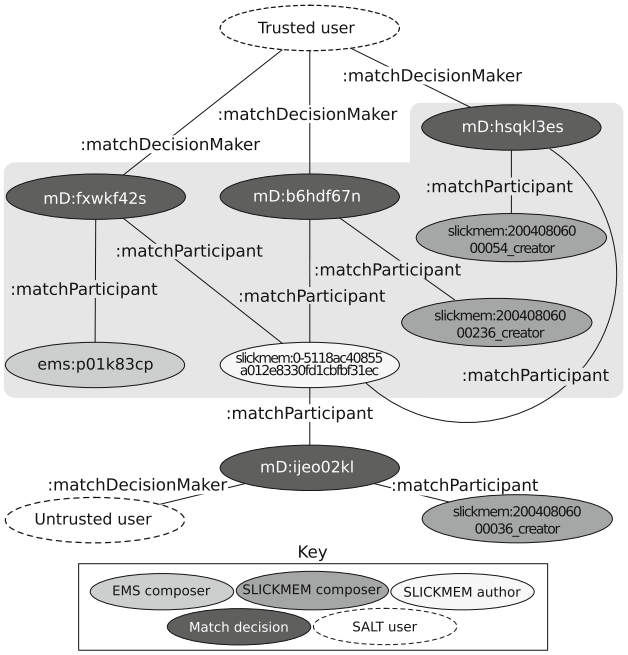

Fig. 3 Linked match decisions forming *match chains*. Entities contained within the *shaded area* are connected by match decisions generated by a trusted user and are therefore included in the match chain. Entities outside this area are connected by decisions generated by an untrusted user and are thus excluded. Match decisions are stored within named graphs specific to each user, denoted here by direct connection to the *dashed nodes*

A match decision ties together two match participants, each of which is the focal topic of alignment for their respective saltset. These match decisions, which may encode confirmations or disputations of a match between the two match participants, are represented as sub-graphs within the named graph. The provenance of each decision is captured by storing associated metadata identifying the user, the date and time, and the reason provided for the decision (Fig. 2), thus accommodating further scholarly activity in the form of replication, agreement, and dispute between different domain experts analysing the same data. Referencing the corpus of match decisions created by a specific individual is simplified by storing decisions in named graphs specific to each user. This storage strategy has two important benefits: the corpus of match decisions generated in the musicologist's alignment activities is represented as a coherent object of scholarly output, addressable via the URI associated with the named graph, facilitating subsequent scholarly exchange, and the named graphs function as a rudimentary trust model, whereby an application building upon the match decision structures can be configured to accept the decisions of users deemed "trustworthy" by the application administrator as valid, while preferring to neglect untrusted match decisions.

As a consequence of the abstract nature of the match decision mechanism, instance alignments are transitive within and across saltsets. For saltsets X, Y, and Z, and the match decision relation R:

$$\begin{aligned} &Let\ U = X \cup Y \cup Z. \\ &\forall x, y, z \in U : (xRy \cap yRz) \Rightarrow xRz \end{aligned} \tag{1}$$

Thus, when instances are matched between saltsets X and Y, and further matches are specified between saltsets Y and Z, implicit instance associations may be inferred between saltsets X and Z. Such match decisions may themselves be configured to function as contextual items in further alignment activities, bootstrapping the task of aligning saltsets X and Z.

4.5 Providing unified access to the matched corpora using the data model and framework

The process of exploiting the published alignment outcomes in order to provide a unified view of the underlying data revolves around the concept of a *match chain*.

This consists of a series of entities linked by match decisions generated by a trusted source. These entities may be included in any saltset subject to alignment activity. Taking the example of our case study in early music, a particular match chain may consist of an EMS composer entity linked to a corresponding SLICKMEM author entity, which in turn is linked to a number of different SLICKMEM composer entities, each associated with a work composed by the person being described (Fig. 3).

The person setting up a unified view can easily limit the graph returned based on an assessment of the trustworthiness of the users making the matches. As match decisions are stored within separate named graphs based on the user responsible for decisions, this is simply a case of requiring the match decision nodes linking the entities participating in

a match chain to be included within a set of named graphs corresponding to users considered reliable.

One of the goals of the work presented here is to harness the power of semantic technologies without requiring the user to be proficient in their use. To support this, our design allows the delivery of a simple JSON object that can serve as the basis of a website presenting the unified view. This JSON object is generated by extracting the information associated with all entities related by a given match chain and retains this information stripped of its semantic context. A web designer wishing to build on a corpus of match decisions is thus able to craft a website presenting data associated with the various entities described in the various datasets, linked by a trusted domain expert, without having to worry about the underlying semantic structure, or indeed about the fact that the datasets were separate in the first place.

5 Semantic Alignment and Linking Tool

5.1 Architecture

We now present a Semantic Alignment and Linking Tool (SALT) that implements the model and design introduced in the previous section. Our tool comprises several interacting software components (Fig. 4):

- A web client presenting the alignment tool's user interface, implemented in HTML/CSS, JavaScript, and the jQuery library.
- An application server implemented in Python using the Flask web application framework.[21]
- An RDF triplestore hosted on the OpenLink Virtuoso database engine platform.

Communication between the client and server uses the HTTP and WebSocket protocols. The triplestore is accessed and updated using SPARQL [44], an RDF query language analogous to SQL in relational databases that enables the retrieval and manipulation of data by specifying patterns of interlinked triples. In our implementation, these queries are performed via the SPARQLWrapper Python module.[22]

A public SPARQL endpoint enabling direct querying of the data is also available. SALT accesses the outcomes of the alignment process via the collection of named graphs containing match decisions that, in turn, link entities from the source datasets. The access control layer provided by Virtuoso grants the back-end process serving the user interface

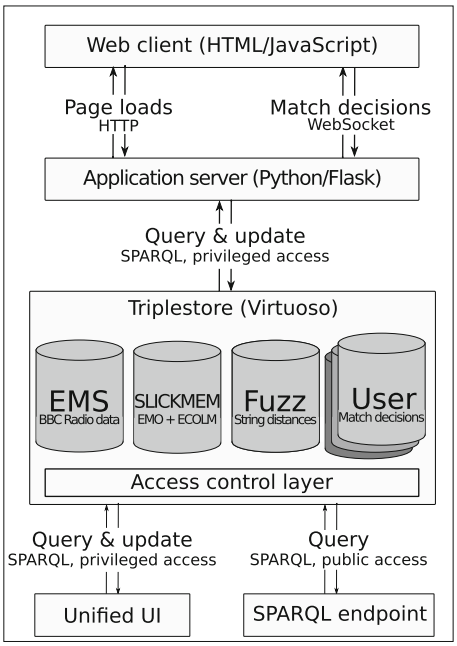

Fig. 4 Semantic Alignment and Linking Tool (SALT) system architecture

privileged access to the data (e.g. if SPARQL updates are to be made available through this interface), or restrict public access to sensitive sections of the data at the SPARQL endpoint. A unified user interface offering a combined view of the underlying data, making use of the same SPARQL endpoint, is described in Sect. 8.

5.2 Configuration

Four steps are required to prepare an RDF dataset for use in SALT. These steps define the entities in the data that are to be aligned, and augment the initial set of RDF triples with additional metadata.

1. Ingest dataset into the triplestore.
2. Assign entities in the dataset to a particular saltset.
3. Calculate string similarity measures between the entities in the saltsets to be aligned.
4. Specify contextual paths between the entities in the saltset and any contextual items in the dataset.

First, the datasets must be loaded into the triplestore. Each dataset is loaded into its own named graph; this is done to facilitate updates and additions to the data. For instance, when new EMS data becomes available (e.g. after a new episode is aired), it is easy to accommodate this by simply reloading the EMS graph with the latest version of the dataset, without affecting any other data housed in the triplestore. This separation into distinct named graphs also facilitates permission

[21] http://flask.pocoo.org.

[22] http://rdflib.github.io/sparqlwrapper.

management when controlling public access to sensitive data (e.g. due to copyright issues).

Once ingested into the triplestore, entities within the dataset are assigned to a saltset. These assignments are performed by inserting a triple asserting that a given entity is in a particular saltset, e.g.

```
bbc:p00fkxm4  salt:in_saltset
          saltset:ems_composers .
```

where `bbc:p00fkxm4` is a unique identifier for a particular composer: Bartolomeo Tromboncino, the murderous trombonist.[23] Note that the assignments are performed on the schema level, so that, for instance, only a single action is required to add all composers from the EMS dataset to `saltset:ems_composers`.

In order to enable match suggestions by string similarity of entity labels, string distances are precalculated between each label of any two datasets to be compared. This calculation is performed automatically using a script that applies different variations of the Levenshtein edit distance metric [27] as implemented by the FuzzyWuzzy Python module.[24] The script then reifies the fuzzy string matches as distinct `fuzzyMatch` entities with two match participants (the URIs of the two entities whose labels are being compared), a match algorithm (indicating the variation of the edit distance used in the comparison), and a match score. The resulting triples expressing string similarity are then ingested into a dedicated named graph in the triplestore.

5.3 Specifying contextual information

Contextual information is used by SALT in two different ways: as visual hints to the user when a particular entity is selected and as a relevance criterion that affects display order when the user requests candidates for alignment sorted by contextual proximity. The process of contextual configuration required is summarised in Fig. 5.

We have adopted JSON-LD for use in the configuration of contextual information in order to minimise the degree of expertise in semantic web technologies required from the SALT administrator. For each saltset, a list of *context paths* specifying the schematic relationship between the entity to be aligned and a potential alignment point is specified.

SALT generates a SPARQL query from the specified configuration by serialising the JSON-LD into RDF Turtle [3] syntax and applying textual transformations on the resulting triples. The outcomes of this query are used to retrieve

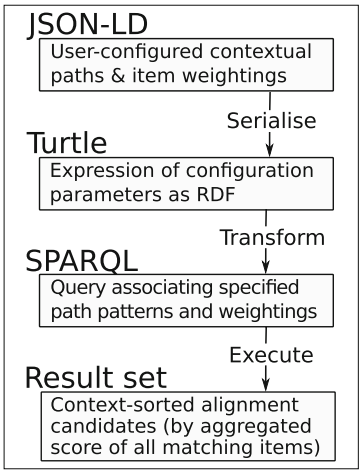

Fig. 5 Process of contextual configuration, from user specification to generation of context-ordered alignment candidates

instances of the specified contextual relationship across the two saltsets to be compared, in order to generate candidate alignment suggestions.

For the purpose of sorting these alignment candidates by contextual proximity, it is important to differentiate between the relative significance of particular kinds of contextual items. For instance, two composer entities sharing a year or place of birth is of minor, but non-negligible, interest, whereas two entities sharing a MusicBrainz ID is a very strong cue that they are likely representations of the same real-world target. In order to address these relative differences in significance, a user-specified weighting is provided for the significance of each contextual item in the context of aligning two specified saltsets during configuration. These weightings are aggregated for each potential combination of cross-saltset entities so that, for instance, an alignment candidate combining entities that share both a year and a place of birth trumps another candidate with entities sharing merely the birthplace, whereas another candidate with entities that share neither birth year nor place, but do share a MusicBrainz ID, trumps both.

5.4 User interface

5.4.1 Matching modes

The web client front-end comprises two scrollable lists corresponding to the two saltsets involved in a given alignment context. Depending on the presentation mode specified by the user in a drop-down menu (labelled 1. in Fig. 6), these lists are either independent of one another, presenting the saltsets in their entirety with entities sorted alphabetically by their labels (*unmatched lists* mode), or the lists are related, so that corresponding rows across lists present a suggested match. These suggestions are made

[23] EMS episode: http://www.bbc.co.uk/programmes/b00zddcm.

[24] http://github.com/seatgeek/fuzzywuzzy.

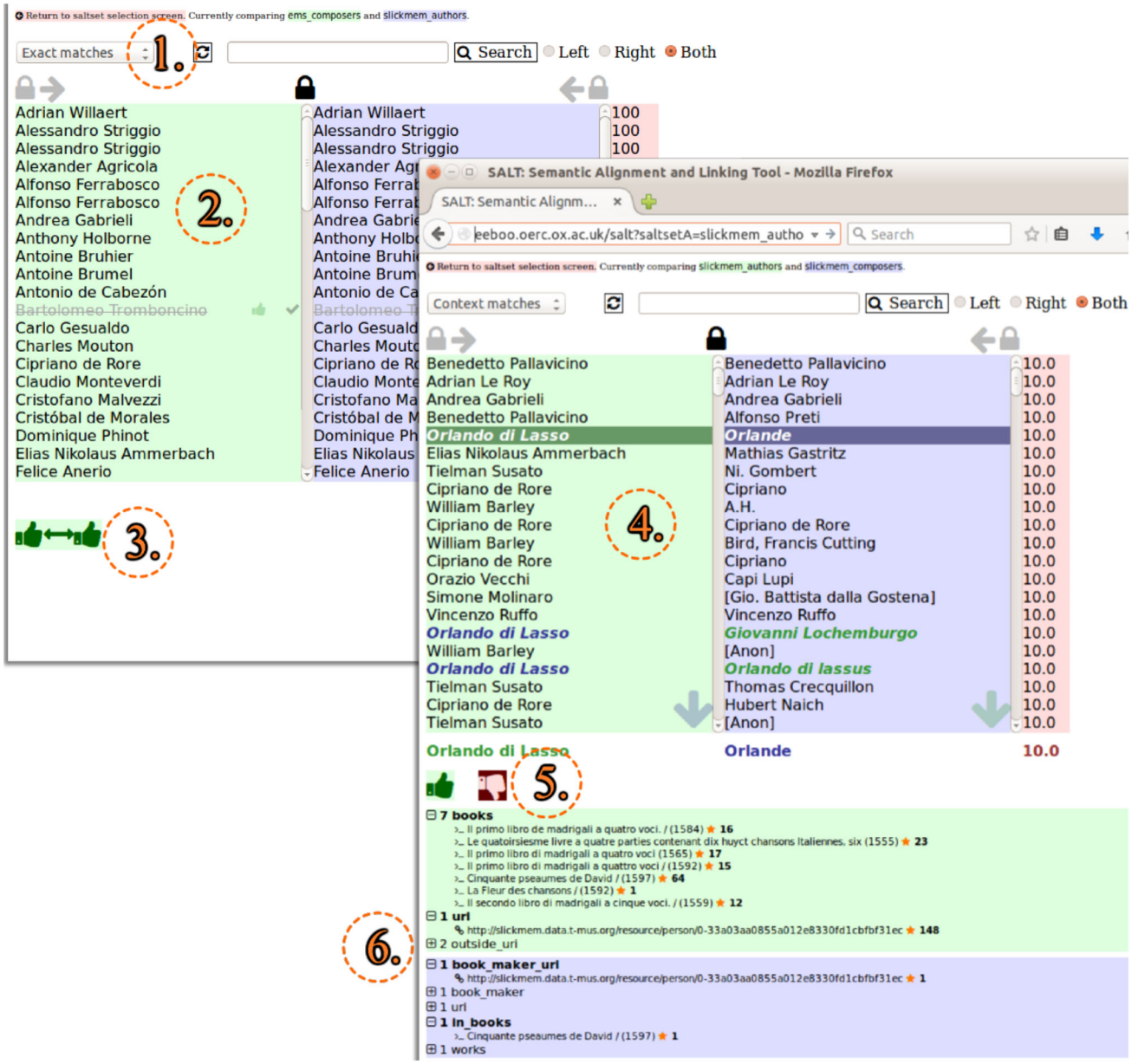

Fig. 6 SALT user interface: exact string matches (*left*) and contextual matches (*right*)

either by similarity of textual labels (2.)—the *exact string match* and *fuzzy string match* modes—or according to contextual similarity (4.) (*contextual match* mode). Scores based on string distance measures or on weighted contextual significance are displayed next to each suggested match. The user may scroll through the lists row-by-row (*locked scrolling*), or unlock the lists and scroll them independently. In order to avoid a combinatorial explosion and to exclude irrelevant information, only those alignment combinations passing a threshold configured separately for textual and contextual similarity are presented to the user.

Searching, filtering, and contextual hinting. Regardless of the currently selected matching mode, the user has several options to filter information: textual search, string match filtering, and contextual item filtering.

Textual searches may be performed via a search box that accepts regular expression inputs. The constraint formulated in the search box may be targeted to either or both lists using a radio button selection. String match filtering is performed by double-clicking a label of interest in either list; upon this action, entities in both lists are filtered to show labels that exactly match that of the target entity. Upon selecting an entity in either list by single-

click, all contextual items associated with the entity are displayed in a UI component situated beneath the two lists (6.). For each associated contextual item, a count is displayed indicating how many entities in the other saltset share this specific contextual item. Further contextual hinting is applied by highlighting the labels of context-sharing entities within the list, and by indicating via the appearance of up and down arrows next to the scroll bar when contextual matches exist above or below the current scroll position. Finally, contextual filtering may be applied by clicking on any item in the contextual display component, constraining the lists to saltset entities sharing the particular item.

Confirming, disputing, and bulk matches. Any combination of inter-saltset entities may be selected in order to *confirm* or *dispute* a match between the two entities (5.), generating one-to-one alignment decisions. Additionally, in any matching mode (i.e. all modes except *unmatched lists*), *row-wise* and *targeted* bulk confirmations may be performed. Row-wise bulk confirmations (3.) may be performed when no item is selected and result in confirmations of matches between every pair of entities sharing a row across the lists (i.e. many-to-many alignment). Targeted bulk confirmations may be performed when one entity is selected in either list (but not both) and produce confirmations of matches between that entity on one side, and every entity on the opposing list (i.e. one-to-many alignment). Prior to either of these actions, individual entities may be *unlisted* in order to withhold them from the bulk confirmation process. Entities that have been subject to a prior match decision are unlisted automatically. The labels of unlisted entities are visually de-emphasised—see Bartolomeo Tromboncino in (2.)—in order to clearly demarcate their status while remaining accessible to the user (e.g. in order to re-list them).

By combining searching, filtering, and unlisting, it is possible to rapidly assert a large number of match decisions. A more fine-grained approach using confirmations and disputations of individual items across the saltsets may then be applied in order to cover more abstruse cases of alignment. Any entities that have been subject to at least one match decision are subtly indicated by a translucent check-mark, in order to help the user "keep their place" during alignment.

When a match decision is made via any confirming or disputing action, the user is prompted to enter a reason for the decision. The match decision is then reified with its own persistent URI, according to the specification in Sect. 4.4. These triples are then passed to the application server and persistently stored in the triplestore. The triples are transmitted from the web client via the WebSocket protocol in order to allow the user to continue their activities without being interrupted by page reloads.

6 Application to case study: alignment of early music corpora

A musicologist used SALT to perform corpus-scale alignments of the early music datasets described in Sect. 2.1. As a first step, this involved aligning entities within the SLICKMEM dataset. These data were sourced to a large degree from a traditional library context (the British Library catalogue) in which the basic unit of description was the book; as such, the names of persons contributing to the creation of the book ("book authors") were subject to a tightly controlled vocabulary enforced by reference to authority files, whereas names associated with composers were less carefully controlled in the source data and thus more variable and ambiguous.

As a consequence, the SLICKMEM dataset publishes book authors as distinct entities, each author with their own persistent URI, whereas composers exist "merely" as name strings attached to works using `rdfs:label`. This limitation had to be addressed if a robust link from a work presented on the BBC radio programme to its representation in SLICKMEM and thus to the corresponding digitised score was to be established. As such, we have bootstrapped distinct composer entities, minting a new persistent URI to represent the composer of each individual work, and associating the composer's name with the new entity, rather than with the work directly. The musicologist then aligned these new SLICKMEM composer entities with the more tightly controlled SLICKMEM authors using SALT, in order to address the ambiguity inherent in representing composers with a distinct entity for each of their works.

For the reason discussed above, the digitised resources available through SLICKMEM are book-centric, rather than work-centric. Thus, the musicologist's next task was to align the composers represented in the EMS data with the authors represented in the SLICKMEM data. The combined outcomes of both alignment activities enabled the robust linking from items of EMS programme data to SLICKMEM resources.

In terms of the within-SLICKMEM alignment, the musicologist confirmed 2564 matches between SLICKMEM composers and authors, involving a total of 266 distinct authors (i.e. 9.6 works per author on average).

The musicologist created match decisions between 68 EMS composers and SLICKMEM authors. Thus, 68 out of 362 distinct composers (19%) featured on the EMS programme are mapped to digitised resources and further metadata via SLICKMEM. At least one of these composers features in 317 of the 507 episodes available at the time of analysis (63%). Thus, just under two thirds of all EMS episodes can be augmented with additional metadata in a unified view of the datasets (Sect. 8), a respectable number given the narrow chronological scope of two centuries in the SLICKMEM dataset, compared with the broader musical

timeline, from mediaeval times to the baroque and beyond, presented on EMS.

7 User evaluation

A user study was conducted in order to evaluate the data model, alignment framework, and SALT user interface beyond the context of the musicologist performing the alignments detailed in Sect. 6.

7.1 Sampling frame

Eight academic musicologists, including one based at an international music library institution, one from a mediaeval manuscript archive, and another with formal background in library and information sciences, participated in the user study. All participants possessed extensive domain knowledge in early music. They were recruited using snowball sampling [31], whereby initially contacted participants were asked to recommend others with similar expertise.

In a post-evaluation questionnaire, each participant self-reported considerable expertise in early music: on a scale of 1 ("*I have never heard of early music*") to 7 ("*I am an expert in early music*"), the median response was 6.5 (minimum: 5). Further, each participant indicated a high degree of familiarity with the author and composer names encountered during the evaluation: on a scale of 1 ("*I did not recognise any of the names*") to 7 ("*I recognised almost all of the names*"), the median response was also 6.5 (minimum: 6). In terms of technical background, responses were slightly more varied: rating their technical expertise in using computers, on a scale of 1 ("*I am a novice computer user*") to 7 ("*I am an expert computer user*"), the median response was 6 (minimum: 3); and rating their familiarity with digital musicology on a scale of 1 ("*I have never heard of digital musicology*") to 7 ("*I am very familiar with digital musicology*"), the median response was 7, with two individuals indicating 1 and 3, respectively. Our sample thus consisted of individuals with strong expertise in early music, varying in terms of their technical background.

7.2 Design and procedure

The evaluation consisted of a practice session and two evaluation tasks, each presenting a subset of the SLICKMEM author-to-composer alignment task. This was followed by a questionnaire investigating participants' familiarity with digital musicology and with early music, and posing several questions relating to their user experience of the evaluation and of the alignment tool. Participation took place remotely using participants' own computers. Evaluation sessions were scheduled to ensure the researcher overseeing the evaluation was available for immediate clarification of questions about the task via e-mail or through video conferencing.

Having indicated their consent for voluntary participation in the evaluation study, participants first read an instructions page, detailing the task objectives and the functionality of the alignment tool. Participants then completed a practice task presenting a subset comprising 25 SLICKMEM authors against the complete list of SLICKMEM composers. The practice task lasted for 10 min, with a timer indicating the remaining time at the top of the interface. During the practice task, participants were encouraged to try out the various functionalities of the tool, as described in the instructions page. After the practice session, participants completed two experimental tasks, in sessions lasting 20 min each. In both tasks, participants were presented with a distinct subset of 50 SLICKMEM authors (i.e. 100 authors across both sessions, all distinct from those presented during the practice session). In both cases, as in the practice session, the authors were presented against the complete list of SLICKMEM composers. All SLICKMEM authors had been previously matched by the musicologist involved in the development of the tool (see Sect. 6), ensuring the existence of matching composers. In task 1, participants completed the evaluation with a limited subset of the tool's modes, working with either: only unmatched lists mode; suggestions based on string similarity, plus unmatched lists mode; or suggestions based on contextual similarity, plus unmatched lists mode. Participants were randomly assigned to one of these conditions. During the practice session and in task 2, intended as our control condition, participants each had access to the full capabilities of the tool.

7.3 Analysis and results

An analysis of the generated match decisions was conducted, employing the evaluation principles outlined by Paulheim et al. [32] (see Sect. 3). The F-measure was determined by calculating precision and recall as defined against a reference set of match decisions generated by the musicologist performing the SLICKMEM author-to-composer alignment in the case study (Sect. 6). Two cost functions were calculated: *matches per interaction*, where the number of distinct match confirmation actions (clicks on a single instance or bulk confirmation button) was considered in terms of the match decisions generated, and *matches per second*, where the average time required for each generated match decision was considered.

Over the course of the user evaluation, our participants discovered 2969 out of a possible 3747 valid author–composer matches (recall: .79), generating a further 246 "erroneous" matches (according to the judgement of the musicologist responsible for the early music use case alignments; precision: .89), giving an overall F-measure of .84; this cor-

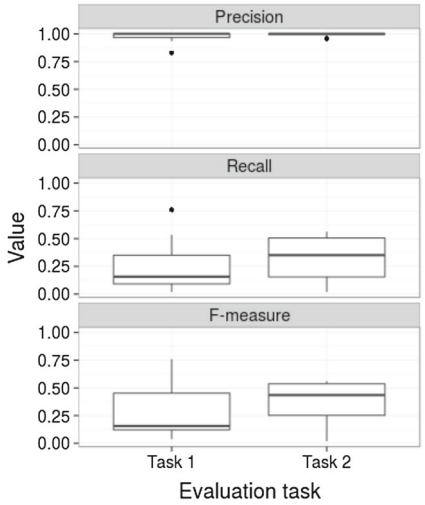

Fig. 7 Precision, recall, and F-measure for each evaluation task

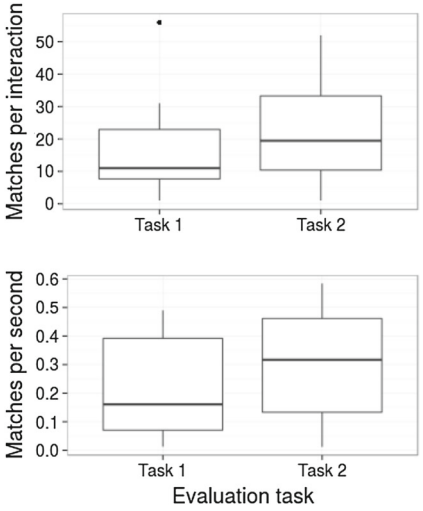

Fig. 8 Cost measures for each evaluation task. *Top* Number of matches generated per confirmation interaction. *Bottom* Number of matches generated per second

responds to *distinct* author–composer matches identified by the *combined* efforts of our participants.

The distribution of individuals' performances in terms of precision, recall, and F-measure is summarised in Fig. 7. Precision was high throughout, as would be expected given participants' domain expertise in early music. Encouragingly, precision remained consistently strong regardless of matching modes employed, suggesting that a greater use of match suggestions and bulk confirmation does not negatively affect the accuracy of the alignment activity.

Recall varied consistently among individuals, corresponding to the variation in alignment efficiency (cost per match decision) discussed above; note that participants were limited to 20 min per interaction task and thus that less efficient use of the tool necessarily resulted in a lower recall score.

The number of matches generated per interaction, and per second, are visualised in Fig. 8. There is a considerable degree of variability between participants, ranging from 1 to 56 matches per interaction, and 0.004–0.6 matches per second. This variability is expected for task 1, given the differences in modes available to participants. The retention of this variability into task 2 was due to a tendency of several participants to remain in the unmatched lists mode that reduced opportunities for greater efficiency via match candidate suggestions and row-wise bulk confirmation. The variation may also reflect differences in the analytical approach between participants, perhaps due to certain participants being more thorough in their confirmation of match candidates.

One participant's performance particularly demonstrates the value of our approach. In task 1, the participant only had access to the unmatched lists mode and thus could not benefit from the advanced functionalities of the tool. During this task, the participant generated 762 matches, at 6 matches per interaction, and 0.3 matches per second. In task 2, the participant made extensive use of the full functionality of the tool, generating 1402 matches, at 36 matches per interaction, and 0.6 matches per second. The participant was thus able to roughly double alignment efficiency, while decreasing six-fold the number of interactions required, demonstrating the value of this approach when the capabilities afforded by the tool are used to the full.

7.4 User experience

Paulheim et al. explicitly placed assessment of the user experience out of scope in their evaluation guidelines, as the aim is to fully automate the evaluation procedure for the interactive matching track of the Ontology Alignment Evaluation Initiative, making measurements of the user experience difficult. However, they note that, for interactive matching tools providing a user interface, measuring user experience is a useful complement to the measures they outline. We attempted to capture these aspects by asking participants to reflect on their user experience in the post-evaluation questionnaire. Participants were asked to rate their perception of the clarity of the task, and usability of the tool, by responding to the statements "*I found the instructions and objectives for this task easy to understand*" and "*I found this tool to be easy to use*" on five-point Likert scales, ranging from "*strongly disagree*" via "*neutral*" to "*strongly agree*". Participants responded to the statement, "*I could usefully incorporate such a tool into my research*", by choosing a response from "*No*", "*Uncertain*", or "*Yes*". Participants were able to elaborate their responses to each question via free-text fields and were given the opportunity to include any further comments at the end of the questionnaire.

As one of the goals of this evaluation, we intended to investigate the relative utility of the different matching modes, hence the distinction between the different modes available, according to experimental condition. Unfortunately, participants tended to remain in the default unmatched lists mode, thus not benefiting from SALT's ability to suggest match candidates, even when other modes would have been available to them—one participant explicitly stated in the post-evaluation questionnaire that the other modes *would* have been explored if there had been more time available. This tendency to remain within the default mode was particularly common among participants with lower self-reported computer literacy and familiarity with digital musicology. Cases where participants fully utilised the SALT functionality did indeed result in the greatest alignment efficiency (Sect. 7.3).

Responses regarding task clarity were mixed, with five participants indicating agreement that the instructions and objectives were easy to understand, one remaining neutral, one participant disagreeing, and a further disagreeing strongly. Similarly, views on the tool's usability were variable, with five participants agreeing that the tool was easy to use, two disagreeing, and one disagreeing strongly. It is possible that some of this variability may relate to differences in the participants' technical backgrounds and experience with digital scholarship; the participant strongly disagreeing in both cases also reported a lack of familiarity with digital musicology and indicated in comments a confusion about the tool's purpose. This participant only completed the practice task of the evaluation, successfully generating 24 match decisions that all correctly matched our "ground-truth" set.

Four participants indicated that they could see a role for this sort of tool in their own research, elaborating responses detailed applicability in other alignment contexts, arising when building digital resources from original sources, and when mapping potentially noisy user input against authority records, as well as a means of handling attribution questions. The remaining participants indicated concerns about scalability, or simply stated that they saw no applicability to their own work.

Although all participants were able to generate match decisions with the tool, most had suggestions for improved usability. These included requests for increased font size (currently, the contextual item view panes use very small font sizes in order to fit more information on-screen); the inclusion of keyboard short cuts, to reduce reliance on mouse clicks; the ability to explicitly select multiple items in either list for one-to-many and many-to-many instance matches (currently, this type of functionality is achieved implicitly, by a combination of filtering, unlisting, and bulk confirmation); and an explicit undo function for single and bulk confirmation operations. It is clear from these responses that there is a learning curve to the current user interface that must be overcome, particularly if the tool is to target users lacking technical expertise. These insights will provide useful guidance to future development work on the user interface (Sect. 9).

7.5 Limitations

Our use case in early music necessarily narrowed the pool of domain experts available for the evaluation of our system. A larger-scale evaluation on a broader knowledge domain, employing a correspondingly greater number of participants in order to obtain a more fine-grained understanding of the utility of the system and its different matching modes, is envisaged for future work. Nevertheless, the present evaluation serves to demonstrate the value of the data model and design underlying our approach; when the tool's functionalities are fully exploited, highly efficient and precise alignment progress can be achieved.

8 Providing unified access to the matched corpora

As the match decisions generated by SALT users are published as Linked Data, the combined information available within the aligned datasets can be queried using SPARQL. However, given a target audience of musicology scholars and laypersons with interests in early music, a more familiar means of access that does not assume knowledge of Linked Data technologies as a prerequisite is clearly required. Accordingly, we now present the Semantic Linking of BBC Radio (SLoBR) demonstrator, a web application inspired by the look and feel of the existing EMS web resource while providing access to biographical information, bibliographical catalogue data, and digitised musical score available via alignment to the SLICKMEM dataset, and via further external datasets made available by this alignment. In designing the architecture to support this demonstrator, we aim to divorce aspects catering to this particular use case from the generic aspects involved in the outcomes of any application of SALT, regardless of alignment context. To support this, we have developed tooling that facilitates the creation of unified views across any saltset combinations linked by match decision structures generated by SALT. This tooling serves to demonstrate the flexibility of our model (Sect. 4), and the reusability of the data it produces, within and beyond the domain of our present use case.

8.1 Implementation

The demonstrator consists of an HTML/CSS front-end with a design loosely based on the BBC GEL website design specification,[25] and a back-end server using the Flask web

[25] http://www.bbc.co.uk/gel.

application framework. The Flask server handles requests by performing template filling using the results of parameterised SPARQL queries to generate the front-end views.

Basic navigational vectors are supported by a generic match chain walking query (SPARQL Query 1). This query returns all information associated with the URI of a specified "source" entity, as well as all information associated with any entities linked to the source entity by a chain of trusted match decisions (see Sect. 4.5). In the early music demonstrator, these entities may each represent a person (EMS composer, SLICKMEM author or composer) or a work (EMS work or SLICKMEM work). The query stitches the datasets together according to the SALT user's alignment activity, enabling all related information to be displayed in aggregated, unified views. We now step through SPARQL Query 1 to explain the process in detail.

Block A sets up the query parameters, specifying input and output variables, and as well as the relevant datasets the query will be confined to in order to improve efficiency. Line 1 describes the variable bindings produced by the successful execution of this query, i.e. the shape of the results set. `?uri` refers to the unique identifier of a particular entity in the match chain; this entity is retrieved from the named graphs listed as possible `VALUES` of the `?contentGraphs` variable in lines 3–6. `?p` and `?o` refer to the predicates and objects associated with these entities; that is, the directly related information that we wish to retrieve. The `{sourceUri}` parameter encased in curly brackets on line 2 specifies the entity URI that serves as an entry point to the match chain; it is filled by the Flask server in response to the user's actions on the web front-end using standard Python string formatting prior to query execution and bound to the `?source` variable when the query is run.

Block B performs the match chain walking operation. The `{trustedGraph}` parameter encased in curly brackets on line 7 specifies the named graph of trusted match decisions, as configured on the server and supplied prior to query execution. Line 8 retrieves all entities that share a match chain with the specified `?source` URI. This is achieved using a SPARQL property path[26] that constrains the query to patterns where the relationship between `?source` and `?uri` is such that `?source` is a match participant in a match decision that also has a match participant `?uri`, or which has an intermediary node that is involved in match decisions with both `?source` and `?uri`. The `*` operator allows for an arbitrary number of repetitions of this pattern,[27] including zero, in which case, `?uri` simply takes the value of `?source`, as a property path of length zero connects a node to itself.

Block C now extracts all information directly associated with the entities contained in the match chain, i.e. all properties (`?p`) and objects (`?o`) of any `?uri` retrieved in *Block B*. This part of the query is constrained to the graphs containing the datasets specified in *Block A*, supporting efficient performance by avoiding the need to search the entire triplestore.

```
Block A: Specify datasets and source URI
1 SELECT DISTINCT ?uri ?p ?o WHERE {
2   BIND({sourceUri} AS ?source) .
3   VALUES ?datasets {
4     :EMS
5     :SLICKMEM
6   }
Block B: Find all URIs in the match chain
7   graph {trustedGraph} {
8     ?source (:matchParticipant/^:matchParticipant)* ?uri .
9   }
Block C: Retrieve all associated information
10   graph ?datasets {
11     ?uri ?p ?o .
12   }
13 }
```

SPARQL Query 1: Retrieve all information associated with a given entity (e.g., an author, a composer, a work) by unified query of the aligned datasets, via match chain walking. *Block A*: Supply source entity URI, and specify dataset graphs. *Block B*: Specify graph containing trusted match decisions, and retrieve the URI of any entity sharing a trusted match chain with the source entity. *Block C*: Capture information directly associated with each URI in the match chain.

An example results set is provided in Table 1. Here, the EMS URI for the composer Orlande de Lassus is supplied as the starting point of the match chain walk. The query returns information on this EMS composer, as well as on the matched SLICKMEM author, Orlando di Lasso, and on 48 distinct SLICKMEM composers—one associated with each SLICKMEM work attributed to the composer—with labels exhibiting 5 variant spellings of his name. As each `?uri` in the results set is part of the chain of match decisions, any one of them could serve as the input `?source` variable to generate identical results.

The set of resulting triples is stored as a simple JSON object storing the predicates (`?p`) and objects (`?o`) associated with the entities in the match chain (`?p` as keys, and `?o` as values). Where there are multiple instances of a certain predicate, potentially with different values—e.g. an EMS composer, SLICKMEM author, and various SLICKMEM

[26] http://www.w3.org/TR/sparql11-query/#propertypaths.

[27] While the subsection on arbitrary length path matching in the property path specification section of the W3C Recommendation on the SPARQL 1.1 Query Language states that "Connectivity matching is defined so that matching cycles does not lead to undefined or infinite results", complex alignment contexts involving very long match chains, or erroneous matches resulting in longer than expected chains, may significantly impact query performance; in such situations, the number of hops can be constrained using the property path syntax.

Table 1 Match chain walking: Example results set produced by SPARQL Query 1

?source	?uri	?p	?o
ems:p012dzzq	ems:p012dzzq	mo:musicbrainz_guid	mbz:853f1c0a-4b59-4957-9c05-c3a8d3a4d5ef
ems:p012dzzq	ems:p012dzzq	salt:in_saltset	saltsets:ems_composers
ems:p012dzzq	ems:p012dzzq	slobr:contributor_role	Composer
ems:p012dzzq	ems:p012dzzq	rdfs:label	Orlande de Lassus
ems:p012dzzq	ems:p012dzzq	rdf:type	dct:Agent
ems:p012dzzq	slickmem:0-33a03aa0855a012e8330fd1cbfbf31ec	salt:in_saltset	saltsets:slickmem_authors
ems:p012dzzq	slickmem:0-33a03aa0855a012e8330fd1cbfbf31ec	rdfs:label	Orlando di Lasso
ems:p012dzzq	slickmem:0-33a03aa0855a012e8330fd1cbfbf31ec	rdf:type	dbpedia:Person
ems:p012dzzq	slickmem:20040806000036_creator	salt:in_saltset	saltsets:slickmem_composers
ems:p012dzzq	slickmem:20040806000036_creator	rdfs:label	Orlan. di Lassus
ems:p012dzzq	slickmem:20040806000137_creator	salt:in_saltset	saltsets:slickmem_composers
ems:p012dzzq	slickmem:20040806000137_creator	rdfs:label	Orlandi di Lassus
ems:p012dzzq	slickmem:20040806000035_creator	salt:in_saltset	saltsets:slickmem_composers
ems:p012dzzq	slickmem:20040806000035_creator	rdfs:label	Orlando di Lasso
ems:p012dzzq	... 90 further entries associated with 45 other saltsets:slickmem_composers ...		

`?source` is an input variable, included here for illustrative purposes. Any of the resulting `?uri` values could equally serve as the `?source` to produce the same results set, as they are all part of the same match chain

composers may form a match chain, each with their own rdfs:label—all distinct values are stored against the predicate as an array. This representation strips out semantic context inherent in the result set, but makes the development of web interfaces significantly simpler. Where the association of the predicates and objects to their source subject (i.e. the entity bound to `?uri` in the result set) must be retained—for example, if only the rdfs:label of the SLICKMEM author is to be displayed as the authoritative name—a secondary JSON object, keyed first according to `?uri` and then by `?p` and `?o`, is also available. Information about the saltset membership of each entity in the result set is made available through these objects using the `salt:in_saltset` property, further facilitating entity class-specific view decisions.

8.2 Early music demonstrator interface

A web interface developed for the SLoBR early music demonstrator provides access to EMS programme data, SLICKMEM catalogue data and digitised score images, as well as further biographical data obtained by federated querying of DBPedia via LinkedBrainz. The interface presents four interlinked views: *episode view*, *episode listing*, *contributor view*, and *work view*.

The *episode view* (Fig. 9) provides access to full details for a particular episode, as available from the EMS programme resource's JSON feed. This includes a synopsis of the content of the episode, an illustrative image (generally of the episode's presenter, or of the featured composer, performer, location, or musical instrument), as well as a listing of the works and composers featured. The items in this listing link to the *work* and *composer* views, respectively. This linking makes use of the EMS composer and work URIs retrieved from the BBC's feed; the combined data then becomes available using the match chain walking technique detailed in Sect. 8.1.

The *episode listing* (Fig. 10) provides a short summary of multiple EMS episodes, ordered chronologically. By default, all episodes are summarised. The list may also be filtered from links situated on the *episode*, *contributor*, and *work views*, to "all episodes featuring" these contributors, this composer, or this work. These links lead to filtered instances of the *episode listing* showing only those EMS episodes featuring at least one of the indicated composer(s) or work(s), a navigational means unavailable from the BBC's EMS programme resource.

The *contributor view* (Fig. 11) provides access to biographical data and a depiction of particular composers, as extracted from structured information of Wikipedia articles via DBPedia. It is worth noting that neither the EMS nor the SLICKMEM datasets include DBPedia identifiers directly. However, these can be obtained via the MusicBrainz database of crowd-sourced music metadata, accessible as Linked Data via the LinkedBrainz project. This information is obtained on page load via a federated query, ensuring that the presented data reflect the latest versions of the corresponding information; alternatively, the data could be cached locally in the triplestore, in order to increase robustness against potential downtime of the external services' SPARQL endpoints. By comparing the life and death dates of the composer, obtained

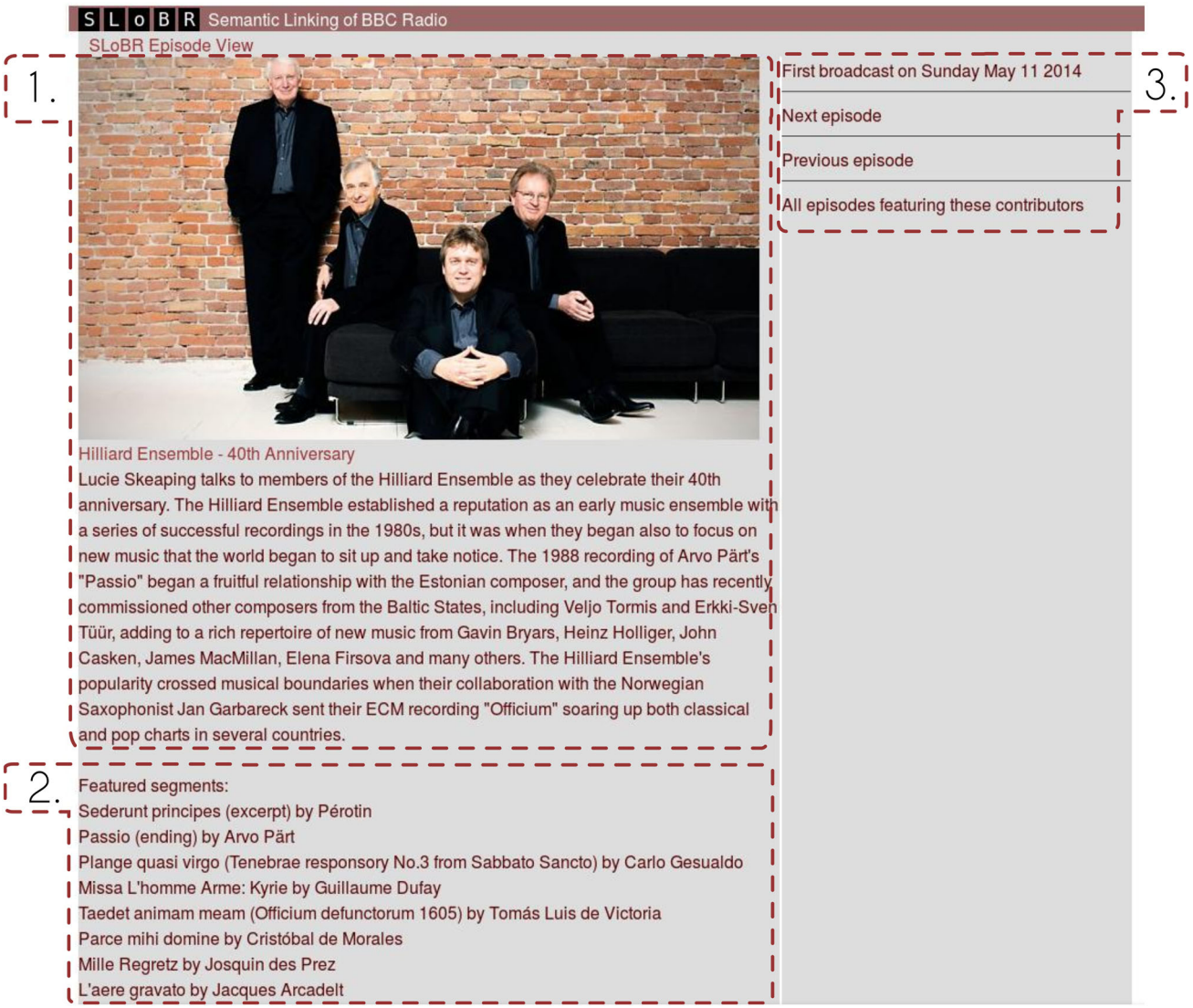

Fig. 9 Episode view. *1*. Full episode details, including image associated with the episode in the BBC EMS programme data. *2*. List of works (with composers) featured in the episode. Work names and composer names are clickable links that reference the corresponding work/contributor view pages using the match chain walking SPARQL query. *3*. BBC broadcast data and inter-episode navigation. `All episodes featuring these contributors` links to the episode listing view, filtered to only show episodes presenting works by composers featured in the current episode (Fig. 10)

from DBPedia, with the publication dates associated with the books described within the SLICKMEM dataset, we arrive at a rough conception of the composer's contemporaries—people who were involved in the creation of music books published during the composer's lifetime. The list of corresponding names is displayed as part of the contributor view, with links to the respective contemporary's contributor view page that enable a novel navigation vector according to temporal proximity. Future work could usefully include a geographical element, defining "contemporariness" along spatial as well as temporal dimensions. Finally, the contributor view also includes a list of works by the composer that have been featured on EMS, linking to the respective work views by virtue of the EMS to SLICKMEM works alignment, as well as the broadcast dates associated with each work's appearance on the show, linking to the corresponding EMS episode view.

The *work view* (Fig. 12) revolves around the display of digitised musical score pages from the book containing the respective work, obtained by following links in the SLICKMEM dataset to images hosted by the EMO Digital Repository at Royal Holloway University of London. A lazy loading technique is used to only load images for pages that currently need to be visible to the user, as well as the next few pages down the scroll list, in order to minimise server load; further images are loaded dynamically as the

Fig. 10 Episode listing, displaying a multiepisode view; either all EMS episodes, or a subset determined by user interaction context—here, the three episodes of the EMS to have featured works by Jacques Arcadelt at time of writing

user scrolls down the list. Navigation to the contributor view for the work's composer, as well as to various filtered episode list views, is supported.

Certain functionalities presented here—the determination of contemporaries, and the retrieval of supplementary detail describing the composer from DBPedia—are driven by parameterising specialised SPARQL templates on the server, and thus, their implementation requires a degree of familiarity with semantic technologies; these functionalities are included in the demonstrator as illustrations of the kinds of added value that is made available by interlinking with external Linked Data resources such as DBPedia. However, the navigational hyperstructure enabling the exploration of the unified corpus, from an EMS episode, to the composers and works featured on that episode, to digitised images of the pages of books featuring those works, is entirely based around applying the match chain walking query (SPARQL Query 1) operating over a collection of match decisions generated by a domain expert using SALT. This process is generic and abstracted from the types of entities involved in the particular alignment context—the process is the same operating over persons as it is operating over works. Its implementation allows users to benefit from the advantages of Linked Data without requiring proficiency in semantic web technologies.

9 Conclusions and future work

In this paper, we have detailed the design of a data model and framework to align and provide unified access to complementary datasets lacking common identifiers. Tackling the ambiguity inherent in automatic alignment processes, and issues of scalability in fully manual alignment, our approach takes a middle path: automatically generating candidate match suggestions based on textual and contextual alignment cues, which are confirmed or disputed by manual application of human insight and domain expertise.

This is accomplished by the definition of *saltsets* comprising sub-graphs of the available RDF datasets. These structures describe alignment anchor entities whose textual labels are of relevance for match candidate generation based on textual similarity. We associate these entities via *contextual paths* to user-configurable *contextual items*, forming weighted graph traversals that provide the cues informing contextual match candidate generation. User *match decisions*, realised through additional RDF structures incorporated into the knowledge graph, confirm or dispute the generated match candidates, capturing provenance information from the responsible user, including their reasoning behind the decision. These match decisions may themselves serve as contextual items, driving iterative alignment activity; further, their transitive nature may be exploited in *match chain walking* to provide unified views of the aligned data.

We have presented the Semantic Alignment and Linking Tool (SALT) and SLoBR (Semantic Linking of BBC Radio) toolsets that implement this design, motivated by a use case in early music combining catalogue metadata and digitised score images from the British library and other sources with programme data from the BBC3 Early Music Show. We have evaluated our approach in a user study employing eight musicologists with expertise in early music, determining highly significant increases to the efficiency of the alignment process when taking full advantage of the semantic affordances of our model.

The domain expert-verified Linked Data generated by SALT can form the basis of novel music digital library systems with user interfaces presenting the underlying datasets

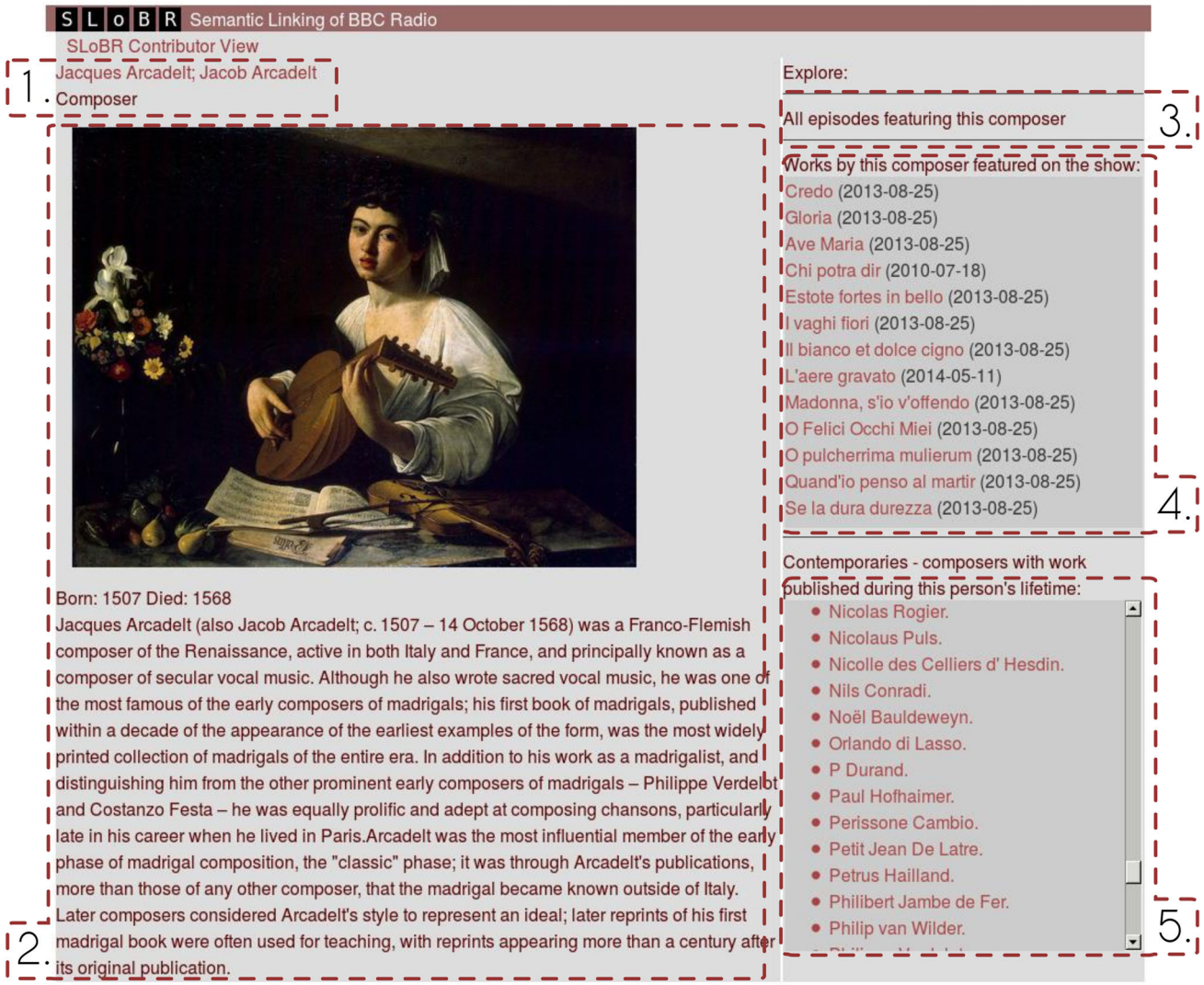

Fig. 11 Contributor view. *1*. Composer name labels associated with the entities within this match chain: the BBC use `Jacques Arcadelt`, the British Library (via SLICKMEM) use `Jacob Arcadelt`. *2*. Where any entity in the match chain is associated with a MusicBrainz ID, we can query LinkedBrainz to retrieve a DBPedia ID. This enables the retrieval of a depiction, birth and death dates, and a biographical blurb. *3*. Link to the episode listing, filtered to show only episodes featuring this composer (Fig. 10). *4*. List of work titles and broadcast dates retrieved from BBC programme. Titles link to corresponding work view (Fig. 12) via match chain walking; dates link to corresponding episode view (Fig. 9). *5*. Links to contemporaries' contributor view pages. Contemporaries are authors that have published books with publication dates that fall within this composer's lifetime

as one union corpus, demonstrated here by the SLoBR web application. The immediate value of such views is in the availability of new connections, e.g. between works presented during an episode of the radio programme, pages of corresponding digitised musical score from the British Library, and biographical information about the composers of the works extracted from sources such as DBPedia. By publishing these connections as Linked Data, we expect further value to accrue as reuse of the data in other contexts is facilitated.

For all the benefits of Linked Data, there are significant barriers to uptake: two of the greatest are the difficulties of publishing pre-existing data in a usefully linked way and, on the other hand, the complexity of exploring a semantic web dataset. In the latter case, the problem is often that, where the representation is rich enough to reflect the data meaningfully, the graph generated is complex and full of indirect paths that limit the use of generic browsers. We believe that both of these barriers can be greatly reduced by the use of shared semantics implicit in the underlying graph structure, used to perform data reduction offered as contextual views to the user.

While our tooling makes profound use of semantic technologies, it is desirable to minimise or eliminate any obligation on the user's familiarity with such techniques. The

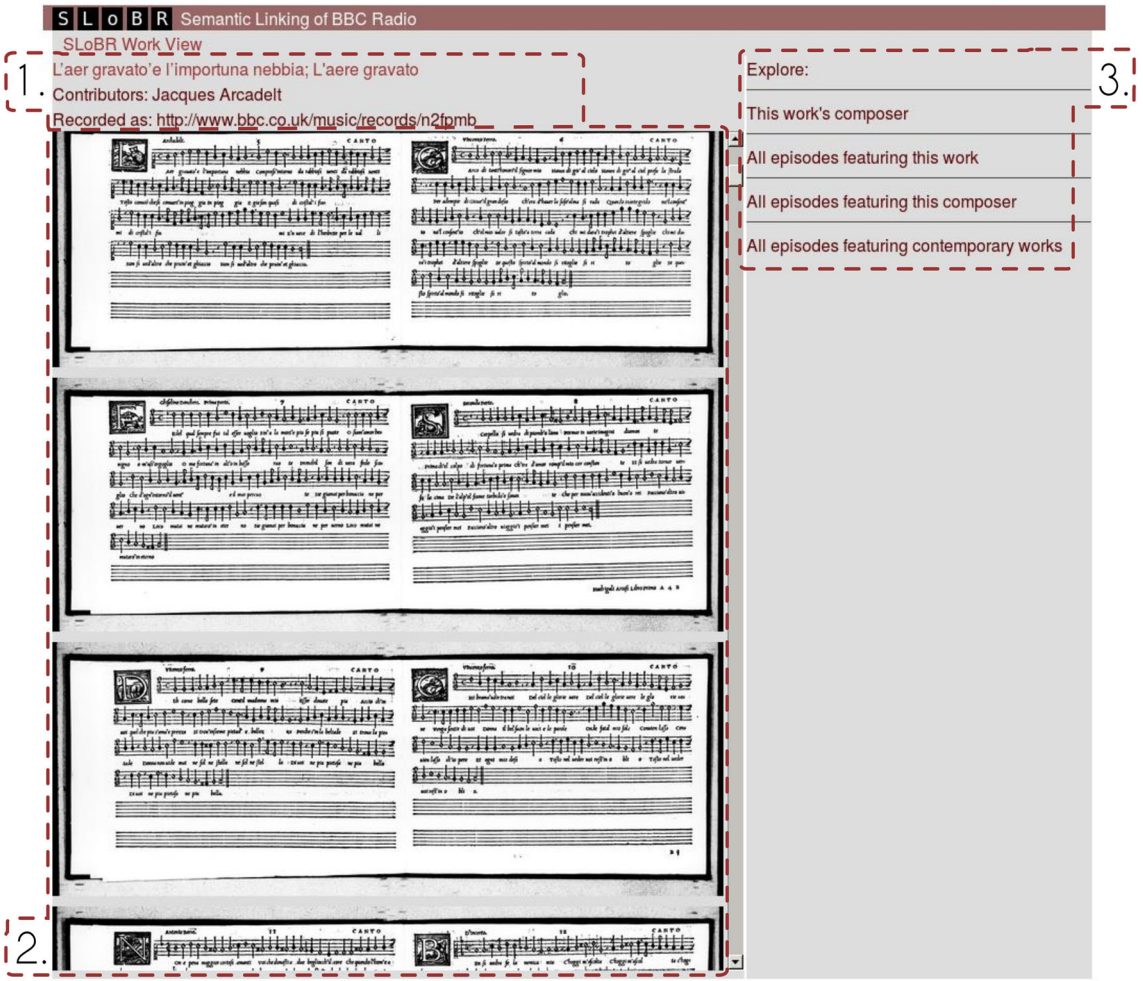

Fig. 12 Work view. *1*. Titles associated with work entities in the match chain. Contributor name and `recorded as` link as per associated EMS programme data. `2`. Score images (click for full-screen viewer) served by Early Music Online via SLICKMEM data associated with this match chain. `3`. Navigational links to work composer's contributor view (Fig. 11), and to filtered episode listings (Fig. 10)

match chain walking mechanism employed by our model to create unified views of the data accomplishes this goal by reducing the required technical knowledge to the much more widespread JSON syntax. Technical expertise at this level is sufficient for the configuration of SALT, which is achieved through the use of JSON-LD; however, a basic knowledge of the semantic schema underlying the data is currently required in order to appropriately set up the saltsets and their associated contextual paths. Algorithms for efficient path finding along a directed graph (the topological form of RDF data) are well studied in computer science [12]. In future development, we plan to make use of such an algorithm in order to automate the configuration of contextual paths over the graph structure of the dataset, given two endpoints (i.e. the focal entity of the saltset, and a contextual item). Further, we will tie specification of the endpoints and the weighting of contextual paths into the UI, facilitating iterative refinement by simplifying the interactive reconfiguration of the system as the alignment process unfolds.

Several improvements to the user interface have been identified based on feedback during the user evaluation (Sect. 7.4). We plan to address these concerns in future development in order to ease the learning curve of the interface, and address the current absence of convenient features including multiple selection and undo functionalities. These developments will be guided by further iterative user evaluation sessions in order to ensure the tool's usefulness to our target audience of domain expert users, while minimising requirements for additional technical expertise. Additionally, some optimisations are required to handle datasets of significantly greater size than those in the early music deployment, including on-demand loading of data (e.g. using web sockets) rather than a complete load on client initialisation, and dataset segmentation or indexing when calculating string distances to avoid a combinatorial explosion in computation.

Further, non-textual representations may be envisioned. We anticipate that multimodal information representations will be of particular interest in the context of digital musicology, for instance in the alignment of audio recordings with musical score. Further plans involve the incorporation of feature vectors obtained from symbolic or audio representations of musical works, using techniques from Music Information Retrieval and available from Linked Data sources such as the Computational Analysis of the Live Music Archive (CALMA) [43] project, to serve as contextual cues in the alignment task.

Our work provides an illustration of the power of tooling that assists, rather than fully automates, the process of digital scholarship, respecting that alignment involves both groundwork in gathering and structuring data, combined with judgement which must always be elevated beyond the groundwork to the purview of the musicologist. As digital resources continue to expand in scope and quantity, the development of tools such as SALT is imperative to overcome the increasing scale and complexity of this groundwork to ensure that the resources within remain accessible to the insight of scholarship. In doing so, we accept and reinforce the observation that the act of study is iterative and ongoing; our data model can provide both a means for capturing the provenance of judgements over complex information structures, and of incorporating these judgements in new and dynamic data structures that can, in turn, provide the foundation for further insight.

For musicologists interrogating the aligned datasets, the simplest benefit comes in the form of clearer, richer explorations. With composers and places linked to external resources, it becomes possible to construct lines of enquiry based on chronology and geography without the information having been entered separately into each database. By accessing the linked EMS and SLICKMEM datasets and similar resources, scholars investigating recent performance history and practice can explore a wider variety of research questions—for instance, the extent to which music programmes by London-based ensembles shape their repertory to the sources that are readily available in the British Library. Consumers and interested laypersons are provided with simplified access and novel navigation vectors that support exploratory browsing and serendipitous discovery. To generalise, linked datasets facilitate the study of the spread of music in historical and contemporary periods with far greater detail and depth than would otherwise be possible without the models and tooling presented here.

Epilogue: Linked Data (re)use

When first designing the RDF structures representing the user's match decisions, our considerations revolved around encapsulating inter-entity links and match decision provenances, in order to drive the iterative alignment process, facilitate unified access to the combined data, and provide an addressable handle to allow the collection of match decisions generated by a particular musicologist to function as a coherent object of scholarly output. This last property of our approach ended up greatly facilitating the analysis of the user evaluation of the alignment tool, reported in Sect. 7. Using the same SPARQL endpoint that drives the various tools presented in this paper, along with some simple set relationship logic, it was easy to determine evaluation measures including precision and recall by determining the differences and overlaps between the collections of match decisions generated by each of our participants against the "ground-truth" set created by our resident domain expert responsible for the alignments in our case study (Sect. 6). The cost measures were also determined with the aid of SPARQL by reference to the captured provenance information associated with each match decision, allowing us to trivially compute the number of match decisions generated per second and per confirmation interaction. Pleasingly, the flexibility and utility of the Linked Data approach in promoting and facilitating data reuse was thus reaffirmed in the process of writing this paper.

Acknowledgements This work was undertaken through the Semantic Linking of BBC Radio (SLoBR) project, a subaward of the EPSRC funded Semantic Media Network (EP/J010375/1), with additional support from the AHRC Transforming Musicology Project (AH/L006820/1), part of the Digital Transformations theme, and continued as part of the EPSRC Fusing Semantic and Audio Technologies for Intelligent Music Production and Consumption (FAST IMPACt) Project (EP/L019981/1). We gratefully acknowledge the support of our colleagues within these projects and our institutions, particularly Graham Klyne for his advice on RDF and SPARQL, and Terhi Nurmikko-Fuller for user feedback during the development of the SALT tool. We thank the musicologist participants in the user evaluation of SALT for generously volunteering their time, and the anonymous reviewers for their thoughtful comments and suggestions on this article.

References

1. Bacciagaluppi, C.: Classifying misattributions in Pergolesi's sacred music. Eighteenth Century Music **12**(02), 223–229 (2015)
2. Barthet, M., Dixon, S.: Ethnographic observations of musicologists at the british library: Implications for music information retrieval. In: Proceedings of the 12th International Society for Music Information Retrieval Conference, pp. 353–358. Citeseer (2011)
3. Beckett, D., Berners-Lee, T., Prud'hommeaux, E., Carothers, G.: RDF 1.1 Turtle: Terse RDF Triple Language. Recommendation, W3C, Feb. 2014. http://www.w3.org/TR/turtle/
4. Bennett, R., Hengel-Dittrich, C., O'Neill, E., Tillett, B.B.: VIAF (Virtual International Authority File): Linking die Deutsche Bib-

liothek and Library of Congress name authority files. In: World Library and Information Congress: 72nd IFLA General Conference and Council. Citeseer (2006)
5. Berners-Lee, T.: Linked Data. Personal Statement: Design Issues, W3C, July 2006. http://www.w3.org/DesignIssues/LinkedData.html
6. Bizer, C., Cyganiak, R.: D2R server—publishing relational databases on the semantic web. In: 5th International Semantic Web Conference, pp. 294–309 (2006)
7. Bizer, C., Heath, T., Berners-Lee, T.: Linked Data—The story so far. In: Semantic Services, Interoperability and Web Applications: Emerging Concepts, pp. 205–227 (2009)
8. Blume, F., Finscher, L.: Die Musik in Geschichte und Gegenwart: allgemeine Enzyklopädie der Musik. Bärenreiter (2008)
9. Bretherton, D., Smith, D.A., Lambert, J., Schraefel, M.C.: MusicNet: Aligning musicology's metadata. In Music Linked Data Workshop, May 2011
10. Brown, H.M.: Instrumental Music Printed Before 1600: A Bibliography. Harvard University Press, Cambridge (1965)
11. Castano, S., Ferrara, A., Montanelli, S., Varese, G.: Ontology and instance matching. In: Knowledge-Driven Multimedia Information Extraction and Ontology Evolution, pp. 167–195. Springer, Berlin (2011)
12. Cherkassky, B.V., Goldberg, A.V., Radzik, T.: Shortest paths algorithms: theory and experimental evaluation. Math. Program. **73**(2), 129–174 (1996)
13. Crawford, T., Fields, B., Lewis, D., Page, K.: Explorations in Linked Data practice for early music corpora. In: Digital Libraries (JCDL), 2014, IEEE, pp. 309–312 (2014)
14. Crawford, T., Gale, M., Lewis, D.: An electronic corpus of lute music (ECOLM): technological challenges and musicological possibilities. In: Conference on Interdisciplinary Musicology, Graz, pp. 118–119 (2004)
15. Cruz, I.F., Antonelli, F.P., Stroe, C.: AgreementMaker: efficient matching for large real-world schemas and ontologies. Proc. VLDB Endow. **2**(2), 1586–1589 (2009)
16. Cyganiak, R., Wood, D., Lanthaler, M.: RDF 1.1 concepts and Abstract Syntax. Recommendation, W3C, Feb. 2014. http://www.w3.org/TR/rdf11-concepts/
17. Dodds, L., Davis, I.: Linked Data Patterns: A Pattern Catalogue for Modelling, Publishing, and Consuming Linked Data, chapter Label Everything (2011)
18. Elmagarmid, A.K., Ipeirotis, P.G., Verykios, V.S.: Duplicate record detection: a survey. IEEE Trans. Knowl. Data Eng. **19**(1), 1–16 (2007)
19. Inskip, C., Wiering, F.: In their own words: using text analysis to identify musicologists' attitudes towards technology. In: Proceedings of the 16th International Society for Music Information Retrieval Conference (2015)
20. Jacobson, K., Dixon, S., Sandler, M.: Linked-Brainz: providing the musicbrainz next generation schema as linked data. In: Late-Breaking Demo Session at the 11th International Society for Music Information Retrieval Conference (2010)
21. Jett, J., Nurmikko-Fuller, T., Cole, T.W., Page, K.R., Downie, J.S.: Enhancing scholarly use of digital libraries: a comparative survey and review of bibliographic metadata ontologies. In: Proceedings of the 16th ACM/IEEE-CS on Joint Conference on Digital Libraries, pp. 35–44. ACM, New York (2016)
22. Kroeger, A.: The road to bibframe: the evolution of the idea of bibliographic transition into a post-marc future. Cat. Classif. Q. **51**(8), 873–890 (2013)
23. Lambrix, P., Tan, H.: A system for aligning and merging biomedical ontologies. Web Semant. Sci. Serv. Agents World Wide Web **4**(3), 196–206 (2006)
24. Lee, J.H., Cunningham, S.J.: Toward an understanding of the history and impact of user studies in music information retrieval. J. Intel. Inf. Syst. **41**(3), 499–521 (2013)
25. Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P.N., Hellmann, S., Morsey, M., van Kleef, P., Auer, S., et al.: DBpedia-a large-scale, multilingual knowledge base extracted from Wikipedia. Semant. Web **6**(2), 167–195 (2015)
26. Leme, L.A.P., Brauner, D.F., Breitman, K.K., Casanova, M.A., Gazola, A.: Matching object catalogues. Innov. Syst. Softw. Eng. **4**(4), 315–328 (2008)
27. Levenshtein, V.I.: Binary codes capable of correcting deletions, insertions, and reversals. Soviet Phys. Dokl. **10**, 707–710 (1966)
28. Liew, C.L., Ng, S.N.: Beyond the notes: a qualitative study of the information-seeking behavior of ethnomusicologists. J. Acad. Librariansh. **32**(1), 60–68 (2006)
29. Nurmikko-Fuller, T., Dix, A., Weigl, D.M., Page, K.R.: In collaboration with in concert: reflecting a digital library as linked data for performance ephemera. In: Proceedings of the 3rd International Workshop on Digital Libraries for Musicology, DLfM 2016, pp. 17–24. ACM, New York (2016)
30. Nurmikko-Fuller, T., Jett, J., Cole, T., Maden, C., Page, K.R., Downie, J.S.: A comparative analysis of bibliographic ontologies: implications for digital humanities. In: Digital Humanities 2016: Conference Abstracts, pp. 639–642 (2016)
31. Patton, M.Q.: Qualitative Evaluation and Research Methods. Sage, Thousand Oaks (1990)
32. Paulheim, H., Hertling, S., Ritze, D.: Towards evaluating interactive ontology matching tools. In: The Semantic Web: Semantics and Big Data, pp. 31–45. Springer, Berlin (2013)
33. Philips, L.: The double metaphone search algorithm. C/C++ Users J. **18**(6), 38–43 (2000)
34. RISM-Zentralredaktion: RISM: An Overview. Brochure, Apr. 2015. http://www.rism.info/fileadmin/content/community-content/Zentralredaktion/20150410_RISM_Broschuere_NEU-1_FINAL.pdf
35. Rose, S.: Early music performer. Early Music Online **30**, 22–25 (2012)
36. Sadie, S.E.: The New Grove Dictionary of Music and Musicians. Groves Dictionaries Inc, Oxford (1980)
37. Shvaiko, P., Euzenat, J.: A survey of schema-based matching approaches. J. Data Semant. **IV**, 146–171 (2005)
38. Shvaiko, P., Euzenat, J.: Ontology matching: state of the art and future challenges. IEEE Trans. Knowl. Data Eng. **25**(1), 158–176 (2013)
39. Sporny, M., Longley, D., Kellogg, G., Lanthaler, M.: JSON-LD 1.0: A JSON-Based Serialization for Linked Data. Recommendation, W3C, Jan. 2014. http://www.w3.org/TR/json-ld/
40. The MARC 21 formats: Background and principles. Approved statement, ALA Machine-Readable Bibliographic Information Commmitte & Network Development and MARC Standards Office, LOC, Nov. 1996. http://www.loc.gov/marc/96principl.html
41. Volz, J., Bizer, C., Gaedke, M., Kobilarov, G.: Discovering and Maintaining Links on the Web of Data. Springer, Berlin (2009)
42. Weigl, D., Guastavino, C.: User studies in the music information retrieval literature. In: Proceedings of the 12th International Society for Music Information Retrieval Conference, pp. 335–340 (2011)
43. Wilmering, T., Page, K., Fazekas, G., Dixon, S., Bechhofer, S.: Automating annotation of media with linked data workflows. In: Proceedings of the 24th International Conference on World Wide Web Companion. International World Wide Web Conferences Steering Committee, pp. 737–738 (2015)
44. W.S.W. Group: SPARQL 1.1 Overview. Recommendation, W3C, Mar. 2013. http://www.w3.org/TR/sparql11-overview/