CLEF. A Linked Open Data Native System for Crowdsourcing

MARILENA DAQUINO, University of Bologna
MARI WIGHAM, The Netherlands Institute for Sound and Vision
ENRICO DAGA, The Open University
LUCIA GIAGNOLINI and FRANCESCA TOMASI, University of Bologna

Collaborative data collection initiatives are increasingly becoming pivotal to cultural institutions and scholars, to boost the population of born-digital archives. For over a decade, organisations have been leveraging Semantic Web technologies to design their workflows, ensure data quality, and a means for sharing and reusing (Linked Data). Crucially, scholarly projects that leverage cultural heritage data to collaboratively develop new resources would benefit from agile solutions to simplify the Linked Data production workflow via user-friendly interfaces. To date, only a few pioneers have abandoned legacy cataloguing and archiving systems to fully embrace the Linked Open Data (LOD) paradigm and manage their catalogues or research products through LOD-native management systems. In this article we present *Crowdsourcing Linked Entities via web Form (CLEF)*, an agile LOD-native platform for collaborative data collection, peer-review, and publication. We detail design choices as motivated by two case studies, from the Cultural Heritage and scholarly domains respectively, and we discuss benefits of our solution in the light of prior works. In particular, the strong focus on user-friendly interfaces for producing FAIR data, the provenance-aware editorial process, and the integration with consolidated data management workflows, distinguish CLEF as a novel attempt to develop Linked Data platforms for cultural heritage.

CCS Concepts: • **Software and its engineering** → **Object oriented frameworks**; • **Applied computing** → **Arts and humanities**;

Additional Key Words and Phrases: Crowdsourcing, provenance, Linked Open Data, Wikidata

1 INTRODUCTION

Collaborative data collection initiatives are increasingly becoming pivotal to cultural institutions and scholars, to boost the population of born-digital archives. However, scholarly projects and applications in the cultural heritage domain have to comply with several requirements to ensure the FAIRness of their data [43], in order

to support the reuse of such data and their long term availability. A number of scholarly data management workflows have been designed [32] to cope with issues related to data collection, update, and publication. To facilitate the task, user-friendly interfaces are required to collect and integrate data with external sources, to validate data quality, and to publish results that can be reused by a broad, diverse audience [22]. In addition, storing provenance information along with content data is deemed essential to prevent inconsistencies when integrating sources, to emphasize content responsibility, and eventually to foster trust in data [2, 6, 39]. Moreover, provenance is fundamental for project management purposes, e.g., to monitor the editorial process and to keep track of data versions. Finally, the usage of repositories for dissemination (e.g., GitHub[1]), and preservation (e.g., Zenodo,[2] Internet Archive[3]) are suggested to improve findability, accessibility, and long-term availability of data.

To cope with these requirements, cultural heritage institutions and scholars have been increasingly leveraging Semantic Web technologies. Consortia of museums, libraries, and archives [18, 23, 30] adopt **Linked Open Data** (**LOD**) as a *lingua franca* to develop data aggregators, promote crowdsourcing campaigns [20], and serve high-quality data to scholars. Several solutions for provenance management have been discussed by the Semantic Web community [27, 36], including the PROV Ontology [33] and named graphs [5], which have been largely evaluated in production environments [37, 40]. In recent years, a few content management systems have been introduced to facilitate scholarly data publication and integration, such as Omeka S[4] or Semantic MediaWiki,[5] which store data in relational databases and provide means to serve Linked Data on demand.

However, it has been argued that several issues affect Linked Data management and publishing systems themselves, spanning from storage to indexing and query-related issues [26]. First, some solutions do not leverage provenance in data management workflows (e.g., Omeka S) and do not serve provenance data as Linked Open Data. In particular, provenance management in database approaches assumes a strict relational schema is in place, whereas RDF data is by definition schema free [26]. Second, mechanisms to facilitate Linked Data collection and integration from external sources are not always implemented in user-friendly interfaces (e.g., Semantic MediaWiki), therefore time-consuming, error-prone, manual reconciliation tasks are still delegated to the user. Lastly, existing systems do not always offer tools for version control, continuous integration, and integration with consolidated data management workflows, which must be ensured by data providers separately.

As a matter of fact, only a few pioneers have abandoned legacy cataloguing and archiving systems to fully embrace the Semantic Web paradigm and manage their catalogues through LOD-native management systems [34]. Institutions seem to prefer to maintain legacy systems for managing the data life-cycle (addressing aspects such as data entry, review, validation, and publication), and to provide dedicated services to access their 5-star data, whether these represent complete collections [21], subsets [13, 17], or project-related data [15]. In contrast to institutions, scholarly projects leverage cultural heritage Linked Data to develop new digital resources since the inception of their projects (see [14] for a recent survey). Such projects often enrich descriptions of cultural heritage artefacts with novel contributions, and in turn they become precious sources of information for cataloguers, professionals, and scholars. A comprehensive Linked Open Data management environment would facilitate their tasks, such as data collection, data quality validation, and data publication, according to FAIR data requirements, so as to ensure seamless integration between diverse stakeholders' data sources. While a few solutions have been developed by scholars to support Linked Data production with user-friendly, provenance-aware interfaces [1], these solutions are not portable or reusable, and are difficult to maintain. This can hinder the use of such solutions, in particular for smaller institutions or projects that have limited technical support.

In this article, we introduce *CLEF, Crowdsourcing Linked Entities via web Form*, an agile, portable, LOD-native platform for collaborative projects [12]. The objective of CLEF is threefold, namely: to facilitate Linked Open

---

[1]https://GitHub.com.
[2]http://zenodo.org/.
[3]https://archive.org.
[4]https://omeka.org/s/.
[5]https://www.semantic-mediawiki.org/wiki/Semantic_MediaWiki.

Data creation, integration, and publication using user-friendly solutions; to leverage provenance information stored in named graphs in data management and editorial workflows; and to integrate Linked Data production with existing data management workflows. CLEF has been designed with a bottom-up approach, taking into account requirements highlighted by two case studies, respectively representative of (i) crowdsourcing campaigns promoted by cultural institutions and (ii) collaborative data collections in scholarly projects. Unlike most existing solutions, CLEF manages Linked Open Data from the beginning of a project, provides means to build user interfaces easily, and manages provenance information as first-class entities. In particular, CLEF is integrated with GitHub, which allows fine-grained provenance information to be traced and to be harmonized with common data management workflows.

The remainder of the article is as follows. In Section 2 we present the two exemplary case studies motivating the design of CLEF. In Section 3 we introduce CLEF's requirements, architecture overview, provenance management, data integration features, and sustainability model. In Section 4 we review existing solutions for Linked Data creation and publication, highlighting gaps, and opportunities with respect to the requirements we identified in the case studies. In Section 5 we discuss benefits, opportunities and limitations derived from the development of CLEF in the light of existing solutions. Finally, in Section 6 we conclude with future work.

2 BACKGROUND AND CASE STUDIES

The landscape of collaborative projects in the Cultural Heritage domain is broad. Several participation models are in place, such as crowdsourcing campaigns promoted by cultural institutions and companies, scholarly collaborative projects, platforms for peer-reviewing content, and so on [9]. Campaigns may require limited contributions from users, like *social tagging*, or can require transcription of texts and correction of metadata. Projects may collect and host contents provided by users, like stories about cultural objects preserved by the institution, or may accept new objects and descriptions curated by users.

Moreover, crowdsourcing campaigns may target a specific community or can be a short-term activity, focused on a particular cultural heritage collection. Investing in updating a legacy data management solution to fit the needs of a focused activity would be overkill. Often, such systems are developed *ad hoc* (e.g., Google Forms) and can only satisfy the data collection requirement. In contrast, existing full-fledged data management solution would necessarily require major changes for satisfying the needs of a local project (further discussed in Section 4).

In this work, we consider two common scenarios, namely: (i) a crowdsourcing campaign promoted by cultural institutions to aggregate metadata and produce new information sources for qualitative and quantitative analysis (Section 2.1), and (ii) a scholarly project to collaboratively collect information on digital heritage resources and provide an online catalogue for discovery purposes (Section 2.2). We collected requirements from two real-world scenarios, respectively the *ARTchives* and *musoW* projects, during interviews with stakeholders (ARTchives, musoW) and project meetings (musoW). The added value of such type of projects to the economy of cultural heritage data is of great importance for institutions, which foster co-curation practices as a way to engage with patrons, to disseminate high quality contents, and to promote cultural awareness and, partly, participation [3, 8]. In the following sections, we present the case studies, highlighting the categorization of user requirements in parentheses.

2.1 ARTchives: A Crowdsourcing Campaign for Art Historians' Archival Collections

ARTchives[6] is a collaborative project supported by six institutions (Bibliotheca Hertziana, Rome; Federico Zeri Foundation, Bologna; Getty Research Institute, Los Angeles; Kunsthistorisches Institut in Florenz, Florence; Scuola Normale Superiore, Pisa; Università Roma Tre, Rome) which aims at surveying the heritage of art historians' archival collections [11]. The objective of ARTchives is to provide scholars with an online tool to retrieve information about archival collections relevant to their studies, gather bibliographic sources, and answer

---

[6]http://artchives.fondazionezeri.unibo.it/.

research questions related to Art historiography with quantitative methods–such as historians' network analysis, topic analysis of debates, interlinking collections and so on.

Since a wealth of information is already available on the Web, the project aims at referencing existing data sources as much as possible and creating new data only when needed (**reusability**). Existing data include bibliographic references from the Open Library,[7] historical information from Wikidata,[8] technical terms from the Getty Art and Architecture Thesaurus,[9] and biographical information from the Dictionary of Art Historians.[10] In particular, a few sources (e.g., biographies) are available as long natural language texts, which cataloguers would copy-paste, and which would require knowledge extraction mechanisms to identify important entities and relations (**enhancement**). Cataloguers must be able to accept or reject automatic suggestions (**accuracy**). The latter can be considered the main contribution of ARTchives, which allows users to build bridges between well-known data sources in Art history, thus simplifying interlinking operations. Moreover, the description of archival collections complies with archival metadata standards, for which existing ontologies shall be reused, and must conform to user-specified templates (**validation**).

Contributors in ARTchives are historians, cataloguers, and archivists belonging to cultural institutions preserving art historians' collections. Members of an internal editorial board review data entered by guest cataloguers before publication. Since several institutions may contribute collections from the same art historian, thereby providing contrasting information on the same people, the peer-review process plays an important role in ensuring data quality and consistency (**consistency**). Existing platforms do not allow keeping track of competing contributions to the same artefact, which are usually incompatible, and only one version (the latest) is stored. In ARTchives, cataloguers would need to prevent data duplication, e.g., by informing cataloguers of existing duplicates, and allowing competing information to be temporarily stored in separate records. Reviewers could validate the one to be shown in the final application (**accuracy**).

Already at the inception of the project, it was clear that the sustainability of data created by ARTchives would be hampered in the long run, since the maintenance of the project is affected by time and resource constraints. To this extent, ARTchives is representative of many small-medium research projects that struggle to maintain their infrastructure in the long run (**preservation**) and would need to donate their data to other projects or initiatives.

2.2 musoW: A Collaborative Catalogue of Music Resources on the Web

Musical data play a key role in the every-day life of musicologists, music teachers/learners, and creative industries, who often need to combine diverse resources (e.g., music scores, audiovisual materials, data) from digital music libraries and audiovisual archives for their purposes. In musoW[11] stakeholders can gather information on musical resources available on the Web and incrementally populate an online catalogue to be used for discovery purposes, e.g., to collect online sources relevant to musicology enquiries or music teaching [10]. Records in musoW include descriptions of online resources, such as digital libraries, repositories, datasets, and relevant software solutions. For each resource bibliographic data, scope insights, and technical information are recorded, e.g., responsible people or projects, relevant musicians or music genres, data licenses, APIs, and other services. Unlike ARTchives, also non-experts can contribute to musoW, who may not be familiar with terminology and existing records in the musoW catalogue. In order to facilitate interlinking records and data consistency, auto-completion suggestions should be provided from controlled vocabularies and existing records (**reusability**).

musoW is part of Polifonia, an EU-funded project which aims at interlink musical heritage resources and producing tools to effectively support scholars, professionals, and people passionate about music in knowledge

---

discovery. Members and accredited contributors of the Polifonia organization repository[12] can validate crowd-sourced records and publish them (**accuracy**). To maximise data reuse, data should be easily accessible and securely backed up (**accessibility**). To track fine-grained provenance of records (**reusability**), every change occurring in records should be registered.

Moreover, to ensure services built on top of musoW can rely on continuous data availability, once a record is published it cannot be unpublished. Rather, records must be flagged as *drafts*, so that it is clear to both users and applications which publication stage the data is at (**persistency**).

musoW is meant to increase the findability of collected resources, which would be otherwise only known to hyper-specialized communities (**findability**). To do so, indexing of resources in well-known search engines is deemed a priority (**discoverability**).

Moreover, to simplify data exploration and discovery, browsing solutions including indexes, filters, and aggregations of records should be generated according to preferences specified in record templates (**exploration**). Since such preferences may change over time along with the population of new data, the definition of new filters and data views should be easy to do and have immediate effect. Lastly, part of the project mission is to contribute to the long-term preservation of catalogued resources (**preservation**).

3   LINKED OPEN DATA NATIVE CATALOGUING WITH CLEF

To design CLEF we started from the user requirements collected from the aforementioned scenarios in a bottom-up fashion. We realized that most user requirements highlighted in the case studies correspond to well-known requirements in Linked Data publishing practices [7], as well as challenges in the development of Linked Data Platforms, as specified (in a top-down approach) by prior works [35]. Our research interest in developing CLEF is to test the strength of such parallelism, and understand the benefits and limitations of LOD-native approaches for solving common problems in the Cultural Heritage domain. Therefore, where applicable, we addressed requirements as design issues of a Linked Open Data management workflow.

3.1   Requirements

For the sake of readability, we grouped requirements into four categories, previously acknowledged as FAIR principles (Findability, Accessibility, Interoperability, and Reusability) [43]. In so doing we want to stress one of the main strengths of CLEF, i.e., producing high-quality, reusable data. This aspect is particularly relevant to the scholarly domain, and in recent years it has been increasingly addressed in the Cultural Heritage domain too [28]. Along with requirements, we propose one or more solutions to drive development. Where applicable, such requirements have been translated into Linked Open Data requirements or functionalities.

  — **Findability.** Data are identified with persistent identifiers (URIs), described with rich metadata, and are findable in the Web.
    – *Discoverability. Allow search engines to leverage structured data for indexing purposes.* Every record is served as HTML5 documents including RDFa annotations.
    – *Exploration. Automatically generate data views to facilitate retrieval.* Operations for automatically generating views, such as filtering, grouping, and sorting, are available. These are ontology-driven, i.e., the result of SPARQL queries.
  — **Accessibility**. Data are accessible via the HTTP protocol, and are available in the long-term via a plethora of solutions for programmatic data access.
    – *Preservation (sources). Request digital preservation of user-specified resources.* Rely on established services like the WayBack Machine[13] for web archiving.

---

[12]https://GitHub.com/polifonia-project/.
[13]https://archive.org/web/.

Preservation (ontologies). Allow direct reuse of up-to-date schemas and data of existing projects. Retrieve information on the user-defined data model from the Linked Open Vocabularies initiative [42].
Preservation (data). Integrate the system with established data management workflows. Bind changes in Linked Data to commits in GitHub and release versions in Zenodo.

- *Persistency. Ensure continuity of services built on top of generated data.* Prevent deletion of published records identified with persistent URIs.

— **Interoperability.** Data are served in standard serializations, include references to standard or popular ontologies, and links to external Linked Open Data sources. While this was not an explicit requirement highlighted from use cases, it is a natural consequence of the usage of Linked Open Data, which makes it easier to work with data (e.g., in data integration over multiple sources).

— **Reusability.** Data are released as open data with non-restrictive licenses, are associated with detailed provenance information and follow well-known data sharing policies.

- *Enhancement. Generate structured data from natural language texts.* Perform Named Entity Recognition over long texts on demand, extract structured data, and reconcile to Wikidata.
- *Consistency. Ensure interlinking of records and correct usage of terminology.* Suggest terms from selected Linked Open Data sources and user-specified controlled vocabularies while creating new records. Allow contradictory information to be recorded as named graphs. Ensure peer-review mechanisms are enabled to supervise contributions from non-experts, and prevent inconsistencies in the final user application. Allow restriction of access, and give privileges to a group of users that share ownership of data on GitHub.
- *Accuracy. Allow fine-grained curatorial intervention on crowdsourced data.* Represent records as named graphs and annotate graphs with provenance information according to the PROV ontology (including contributors, dates, and activities/stages in the peer-review process). Update annotations every time a change happens in the graphs. Track changes and responsibilities in GitHub commits.
- *Validation. Allow automatic validation of data.* Along with manual curation, perform schema and instance level checks to ensure created data conform to user-generated (ontology-based) templates.

Moreover, while not in scope in FAIR principles, the use cases highlighted that in order to prevent error-prone operations, guarantee high-level data quality standards, and serve easy-to-find data, **user-friendly interfaces** are necessary or highly recommended. Therefore, the provision of easy-to-use interfaces becomes a fundamental user requirement of CLEF to ensure (1) reusability of data and easiness of exploration for the final user, and (2) simplicity and error avoidance for editors and administrators.

In summary, the interaction with stakeholders highlighted three important research areas, namely: (1) the need of user interfaces to manage most data management processes, which would otherwise require complex or time-consuming operations to be performed manually (e.g., data reconciliation, data quality validation, data exploration) (UF); (2) the importance of provenance management in the editorial process (PM); and (3) the compliance with reusability and sustainability requirements and the integration with data management workflows for scholarly data (DMI). Managing data natively as Linked Data allows us to address all three aspects and to fully comply with FAIR principles.

3.2   CLEF Overview

CLEF is a highly configurable application that allows digital humanists and domain experts to build their own crowdsourcing platform, to integrate the data management workflow with Linked Open Data standards and popular development and community platforms, and immediately enjoy high-quality data with exploratory tools. CLEF is a web-based application in which users can describe resources (e.g., real-world entities, concepts, digital resources) via intuitive web forms. To help with entering descriptions, users are offered auto-complete suggestions from vocabularies, existing records, and terms automatically extracted from text. Administrative users
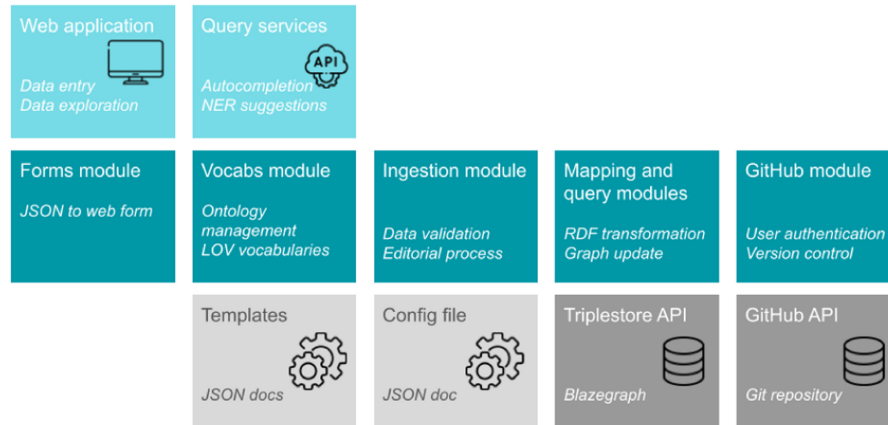
Fig. 1. CLEF overview.

have full control of the setup of their CLEF application and of the definition of templates for describing information about their resources. The templating system of CLEF is configurable via a web interface, in which each form field for describing the resource is mapped to an ontology predicate chosen by the user. Templates are the main drivers of the application, since they ensure consistency in data entry and data validation, they guide the peer-review of records, and are fundamental in retrieval and exploration via actionable filters. It's worth noting that the template setup, in which the ontology mapping is manually (at present) curated, is the only input that requires expert users.

Both authenticated and anonymous contributions can be enabled. A simple peer-review mechanism allows users to curate their records, publish them, and continuously populate the catalogue of contents, which can be immediately browsed via automatically generated interfaces and filters. The tool is particularly suitable for collaborative projects that need to restrict access to members of one or more organizations, to share data and code on dissemination repositories, and that need an environment to discuss project issues. In fact, CLEF is designed to be easily integrated with GitHub and simplify the data management workflow, naturally supporting several of the FAIR requirements.

Figure 1 presents an overview of the CLEF data management system. In detail, CLEF allows an administrator to configure and customize the application via user-friendly interfaces. In the **configuration** setup, users can specify information relevant to their dataset, e.g., URI base, prefix, SPARQL endpoint API (with default configuration), and optional mechanisms for version control and user authentication (via GitHub). The setup of dereferencing mechanisms is delegated to the adopter, who can choose and set up redirection rules by means of their favorite persistent URI provider (e.g., w3id).[14]

For each type of resources to be collected and described, a **template** is created in the form of a JSON mapping document. This includes form field types (e.g., text box, checkbox, dropdown), expected values (literals, entities), services to be called (e.g., autocomplete based on Wikidata and the catalogue, Named Entity Recognition in long texts), the mapping between fields and ontology terms or controlled vocabularies, and whether the field should be used in a default web page, called *Explore*, as a filter to aggregate data.

Ontology terms and terms from controlled vocabularies specified by users are managed via the **vocabulary** module. It's worth noting that, while users can specify their own ontology terms, CLEF fosters reuse of popular and standard vocabularies. The module updates the CLEF triplestore with user-specified terms (which may be

---

[14]https://w3id.org/.

new terms or terms belonging to existing ontologies) and calls the APIs of *LOV Linked Open Vocabularies* [42] to retrieve original labels and comments associated with reused terms. The resulting data model is shown in a dedicated web page called *Data Model* along with information retrieved from LOV.

The **form** for data entry is generated according to settings specified in templates. While editing (creating, modifying, or reviewing) a record, both CLEF triplestore and external **services** like Wikidata APIs are called to provide suggestions. Every time a record is created/modified, data are sent to the **ingestion** module. The latter performs a first validation of the form based on the associated templates, and calls the **mapping** module, which transforms data into RDF according to ontology terms specified in the template and updates the named graph created for the record.

CLEF supports a compliant SPARQL 1.1 [24] endpoint as back-end, therefore, it is not dependent on a specific implementation. However, current running instances use Blazegraph [41]. In particular, named graphs are extensively used to annotate and retrieve provenance information needed to manage the peer-review process, and to efficiently serve record-related information in the exploratory interfaces.

A module is dedicated to the interaction with **GitHub**. GitHub was chosen for its popularity as a dissemination platform for versioned code and data, which fosters visibility of project results, and for its services—i.e., APIs for read-write operations, OAuth mechanisms. Users may decide to bind their application to a GitHub repository, which allows them (1) to store a backup of data in a public/private repository, (2) to keep track of every change to data via commits, and (3) to enable user authentication to the web application via GitHub OAuth.[15]

Lastly, to increase findability of collections, a few automatically generated web pages serve browsing and search interfaces over the catalogue. Currently, CLEF provides the following templates: a homepage; the backend controller from which to access the list of records, the setup configuration form and the templates forms; records creation/modification/review and publication forms; a *Documentation* page with instructions on the usage of forms; the *Explore* page, where views on collected data are shown and filtered; a template to display the records wherein Linked Open Data are also served as RDFa annotations; a template to display controlled vocabulary terms and statistics on their usage in the catalogue; a *Data model* page, collecting ontology terms and definitions from LOV; and a GUI to query the SPARQL endpoint.

The software has been developed in two phases. An initial data management system was developed for ARTchives.[16] In a second phase, the code base has been extended and adapted to be customizable and reusable as-is in other crowdsourcing projects. CLEF is developed in Python, based on Webpy,[17] a simple and small-size framework for web applications. The source code of CLEF is available on GitHub[18] and Zenodo [12].

CLEF is a production-ready solution. It is under continuous development to become a flexible tool for a wider range of collaborative scholarly projects. Potential scalability issues have been improved in recent SPARQL/quad store implementations and there is a continuing effort in the community to support analysis of performance [29, 38].

The initial version of the system was tested with around 15 cataloguers of the six institutions promoting the ARTchives project, namely: the Federico Zeri Foundation (Bologna), Bibliotheca Hertziana (Rome), Getty Research Institute (Los Angeles), Kunsthistorisches Institut in Florenz (Florence), Scuola Normale Superiore (Pisa), and Università Roma Tre (Rome). Currently, user tests are continuously performed by Polifonia project members, who provide new requirements to foster development and research, documented in the musoW repository issue tracker.[19] User tests will soon be performed with users with different profiles and less technical experience.

---

[15]https://docs.GitHub.com/en/developers/apps/building-oauth-apps/authorizing-oauth-apps.

[16]ARTchives source code is available at: https://GitHub.com/marilenadaquino/ARTchives.

[17]https://webpy.org.

[18]CLEF source code available at: https://GitHub.com/polifonia-project/clef/releases/latest, ISC license.

[19]https://GitHub.com/polifonia-project/registry_app/issues.

3.3 The Editorial Process: Provenance Management and User Authentication

In CLEF, every record is formally represented as a named graph [5]. Named graphs enable us to add RDF statements to describe those graphs, including their provenance, such as activities, dates, and agents involved in the creation and modification of a record. Provenance information is described by means of the well-known W3C-endorsed PROV Ontology [33]. Moreover, named graphs allow us to prevent inconsistency caused by competing descriptions for the same entities, for instance when different cataloguers describe the same creator of multiple collections. While this scenario is allowed, users are informed of existing potential duplicates when creating a new record, which prevents involuntary inconsistencies.

The editorial process in CLEF addresses three phases: record creation, record modification, and review and publication. When creating a record, the corresponding named graph is annotated with the identifier of the responsible user (anonymous user if no authentication method is set), the timestamp, and the publication stage unmodified. When modifying a record, additional provenance information is added, including the identifier of the (new) responsible user, the new timestamp, and the new stage (modified). Lastly, when publishing the record, the stage changes to published. A published record can be browsed from the *Explore* page, searched from a text search, and can be retrieved as Linked Open Data from the SPARQL endpoint, via the REST API at <APP-URL>/sparql. While a published record can be modified, and therefore moved back to the stage modified, it cannot be unpublished. While this may be inconvenient in some scenarios, this prevents applications relying on records (and related persistent URIs) from getting unexpected, inconsistent responses.

We chose GitHub to manage user authentication, fine-grained provenance tracking, and version control. In general, CLEF allows both authenticated and anonymous users to create new records. However, records can be modified and published only by accredited users. CLEF is optimized to authenticate users that have a GitHub account. To enable GitHub authentication in the initial setup of CLEF, the admin must perform a one-off operation and specify (1) their GitHub credentials, (2) a GitHub repository they own, and (3) must have created an OAuth App connected to their repository, so as to enable read-write operations on the repository and to confirm the identity of collaborators.

Every time a change is made to a record, content data, and provenance information are updated on the triplestore via its REST API, on the file system, and—if enabled—also on GitHub. To avoid spamming, only records that have been reviewed are stored on GitHub, thereby initializing the versioning. All changes to records are identified by a commit on the repository, and it is possible to track which information (i.e., field of the resource template) has been modified. While such information is currently not stored as Linked Open Data, auxiliary tools such as git2PROV [16] can be used to generate PROV-compliant RDF data. In so doing, we prevent the development from scratch of features that are anyway available on GitHub, and we intertwine the two platforms for a better data management workflow.

In case a user decides not to enable the GitHub synchronization, data are stored in the local triplestore, changes in data are recorded with minimal provenance information (date of changes and publication stage), and only anonymous contributions can be made to the platform. The latter scenario is particularly handy if the application runs only locally (e.g., because contributors do not have the possibility to run the application on a remote server). Indeed, users may decide to create data via their own private instance of CLEF (which runs as a web application in localhost), to store data on their local triplestore, and to manage the publication as they prefer. Moreover, if the application runs locally and user authentication is not enabled, but local users have a GitHub account and collaborate on a GitHub repository with other users, they may decide to keep working locally with CLEF, and backup their data on the shared repository. While publishing a remote instance of CLEF without any user authentication method is discouraged, CLEF implements anti-spamming mechanisms, in order to limit contributions from IP addresses, and to disable write operations on the triplestore.

3.4 Support Data Collection: Reconciliation and Enhancement

When creating or modifying a record, contributors are supported in certain tasks relevant to the reusability of their data, namely: (1) data reconciliation, (2) duplicate avoidance, (3) keyword extraction, (4) data integration.

In detail, when field values address real-world entities or concepts that can appear in other records, autocomplete suggestions are provided by querying external selected sources (live) and the SPARQL endpoint of the project at hand. Suggestions appear in the form of lists of terms, each term including a label, a short description (to disambiguate homonyms) and a link to the online record (e.g., the web page of Wikidata entity or a record already described in the project). If no matches are found, users can create new entities that are added to the knowledge base of their project and these will appear in the list of suggestions in new records. Currently, CLEF is optimized to work with Wikidata, but implementations of entity linking from the Open Library, the Getty AAT, and the Getty ULAN are available too.

When designing a resource template, users can flag a specific field to be used for disambiguation purposes (e.g., the field *title* for a book, the field *name* for a person). When creating a new record, the specified field is bound to a lookup service that alerts the user of potential duplicates already existing in the catalogue. The user may accept or ignore the recommendation.

Some fields may require contributors to enter long free-text descriptions (e.g., historians' biographies, scope, and content of collections), which include a wealth of information that cannot be processed as machine-readable data. To prevent such a loss, SpaCy[20] APIs are used to extract entity names (e.g., people, places, subjects) from the text. Extracted entities are reconciled to Wikidata entities and keywords (bound to Wikidata QIDs) and are shown to users to approve/discard. Approved terms are included in the data as machine-readable keywords associated to the subject entity.

When Wikidata terms are reused, the system can be configured to query the Wikidata SPARQL endpoint to retrieve and store context information in the knowledge base. For instance, in ARTchives, Wikidata entities representing artists, artworks, and artistic periods (recorded as subjects addressed by contents of archival collections) are automatically enriched with time spans, retrieved from the Wikidata SPARQL endpoint, and saved in the local triplestore; likewise, Wikidata entities representing historians are enriched with birth and death places. Finally, entities can be geo-localized via OpenStreetMap APIs.[21]

3.5 Data Sustainability: Ontologies, Data, and Long-term Preservation Strategies

Long-term accessibility of scholarly projects is often hampered by time and resource constraints. A well-known problem is the maintenance of ontologies adopted by small-medium crowdsourcing or scholarly projects [4]. While CLEF does not prevent the creation of new ontology terms, which are stored along with data, CLEF supports reuse of external ontologies. Terms from external ontologies can be directly referenced in resource templates to map form fields to predicates and to map templates themselves to classes. Where reused ontologies are popular or W3-endorsed ontologies, CLEF allows enriching referenced terms with definitions provided by **Linked Open Vocabularies** (**LOV**). Note that reused ontologies are not imported. This design choice has the evident drawback of preventing inference mechanisms, which are not applicable without manually importing ontologies in the knowledge base created by CLEF. Nonetheless, due to this design choice, we believe CLEF has the merit of complying with another debated requirements in the Semantic web community [4], namely, the ability to rely on up-to-date information on reused ontologies, provided by LOV.

Like the projects themselves, the wealth of data produced by scholarly initiatives often becomes unavailable in the mid/long-term. To prevent that, CLEF adopts several strategies. First, CLEF is optimized to reuse Wikidata as much as possible, both at schema level (users can choose classes and properties from Wikidata data model) and at instance level (autocompletion suggestions reuse individuals from Wikidata). The idea is to support stakeholders

---

[20]https://spacy.io/.
[21]https://www.openstreetmap.org/.

Table 1. Overview of Systems for Collaborative Data Collection and Linked Data Publishing

Name	User friendly (Users)	User friendly (Admin)	Provenance Mgmt.	Data Mgmt. integration	Sustainability Reusability
LED	✓	✓	✓	✓	
OmekaS	✓	✓			✓
Semantic MediaWiki	✓	✓	✓		✓
Sinopia	✓	✓			✓
ResearchSpace	✓		✓	✓	✓
Wikibase	✓	✓	✓	✓	

in producing curated metadata that can be exported and imported into Wikidata according to its guidelines for contributors.[22] While Wikidata allows users to also import non-Linked Data into the knowledge base, and to manually perform entity matching, CLEF data include entities already matched with Wikidata QIDs, avoiding the need for manual matching. Data can be retrieved via the SPARQL endpoint or via the GitHub repository.

Second, by synchronizing CLEF knowledge graphs with GitHub it is also possible to synchronize the repository with Zenodo. Zenodo is a certified repository for long-term preservation, widely recognized in the scientific community. Zenodo has recently offered the opportunity to link GitHub repositories to their platform, binding GitHub releases to new versions on Zenodo, uniquely identified with a DOI.

Lastly, the case studies highlighted the need to access and extract information from online web pages (e.g., the Dictionary of Art Historians, online music resources) and reference the source in records. Such web pages are cited as sources of information or are described as first-class entities in records, and are likely to be explored by final users of the project catalogues. Ensuring the persistence of such pages in the long-term is an important aspect, which contributes to foster trust in scholarly and cultural heritage projects. While preserving the original web sources along with data created in CLEF would be inconvenient for small-medium projects—that cannot afford to archive all the web sources they mention—CLEF allows users to specify which form fields include URLs that should be sent to the Wayback Machine,[23] which in turn takes a snapshot of the webpage and preserves it.

4 RELATED WORK

Over the years special purpose systems have been designed by cultural heritage institutions to systematically collect user-generated data and serve Linked Open Data. However, such software and initiatives do not address all our main requirements - i.e., **user-friendliness for final users (UF-U)** and **administrators (UF-A)**, **provenance management (PM)**, and **data management integration (DMI)**. Moreover, researchers have argued that solutions developed for a single institution or project turn out not to be sustainable or not motivating for users [31]. Therefore we also consider sustainability and reusability out of the original context (SR) when reviewing prior works. An overview of surveyed systems is shown in Table 1.

Among the scholarly projects that have been leveraging Semantic Web since early stages of data collection, we acknowledge the **Listening Experience Database (LED)** [1], which exceptionally adopts Semantic Web technologies to support the entire life-cycle of data management, from data collection to publication, by means of user-friendly interfaces (UF-U, UF-A). It offers user-friendly interfaces for data entry, peer-review, and exploration, leveraging several external services and sources, such as the British National Bibliography, DBpedia, and LinkedBrainz. Moreover, each record contributed by users is represented as a named graph, and provenance is accurately annotated for each graph (PM). Data are served via a SPARQL endpoint and a daily backup is provided as a link on the website (DMI). Currently, LED relies on an ad-hoc application developed to serve project-related

---

[22]https://www.wikidata.org/wiki/Wikidata:Data_donation.
[23]https://web.archive.org/web.

goals. The software relies on a heavily customized version of the Drupal CMS, which is not adaptable to support different data models and ontologies and therefore it cannot be of immediate reuse in projects with a different scope (SR).

In recent years, a few content management systems have been introduced to facilitate new projects to publish Linked Data via reusable platforms. Omeka S[24] is a popular platform for collaborative data collection and creation of virtual exhibitions (UF-U, UF-A). User groups and roles can be defined. However, editors do not have the means to supervise the peer-review process on records, including important provenance-related aspects like changes made by contributors in records, flagging records as under review or ready for publication, which must be manually included by users (e.g., records form fields) or managed separately (PM). Records (also called *items*) are served as JSON-LD documents via API. However, data cannot be accessed in any other RDF serialization and cannot be queried via a SPARQL endpoint, which may be cumbersome in some situations, e.g., the exploration of the dataset for analysis purposes. CSV data exports can be requested on demand via a dedicated plugin. However, no mechanism for automatic data versioning is active (DMI). It is also worth noting that the visibility (i.e., the availability) of records can be constantly modified, thereby hindering the continuity and reliability of services relying on data served by the application (DMI). The software is open source, actively developed and maintained by a broad community, and serves several projects in the Cultural Heritage domain (SR).

Semantic MediaWiki[25] is another popular tool used in well-known projects like Wikipedia, which allows data to be displayed in a catalogue fashion (UF-U). New records can be created via web forms and external sources can be used to populate fields when dedicated plugins are installed (UF-A). The system enables fine-grained editorial control (PM) and serves data as LOD, which are stored in a triplestore on which SPARQL queries can be performed (DMI). Like Omeka S, Semantic MediaWiki is open source, actively maintained, and supported by a broad community (SR).

Sinopia[26] is a web-based environment developed by the LD4P (Linked Data for Production) initiative, based on Library of Congress's BIBFRAME Editor and Profile Editor.[27] Sinopia potentially supports other ontologies than BIBFRAME, and users can customize templates for the creation of RDF triples via web forms (UF-A). Unfortunately, Sinopia does not provide a fully-fledged editorial workflow. In fact, provenance information must be recorded manually by cataloguers, which is otherwise not stored along with data (PM). Like in OmekaS, data are stored in a relational database and are shared via APIs that serve JSON-LD documents (DMI). The software is currently under development (SR).

ResearchSpace[28] is a semantic web platform that allows heritage institutions to create and publish collections as LOD. Indeed, the platform is optimized to build exhibitions (UF-U), and it allows the creation of sophisticated browsing interfaces via user templates. While providing flexibility and freedom to customize templates according to user-generated Linked Data patterns, unfortunately, an expert user is needed to encode such preferences in the template, using a combination of HTML, SPARQL, JavaScript, and a custom templating language (UF-A). Likewise, the entire setup of the back-end functionalities requires such a templating system to be manually setup (UF-A). Notably, when creating new records, provenance information is stored along with the data (PM). Linked Data are stored in a triplestore and a SPARQL endpoint REST API is provided. Assets like ontologies and vocabularies can be versioned by configuring a Git repository (DMI). Data can be uploaded to the triplestore via the back-end, and can be manipulated with several types of visual authoring tools (e.g., image annotators, PDF annotators, graphical network editor) (UF-U). The software is based on Metaphactory [25], a commercial enterprise product, customized to comply with the requirements of museums and libraries supporting ResearchSpace (SR).

---

[24]https://omeka.org/s/.
[25]https://www.semantic-mediawiki.org/wiki/Semantic_MediaWiki.
[26]https://sinopia.io.
[27]https://bibframe.org.
[28]https://https://researchspace.org/.

Wikibase [19] is the software that runs Wikidata. It includes several components, such as MediaWiki engine, MariaDB, and Blazegraph for data storage and Elasticsearch, and can be used in combination with other software solutions for data management, e.g., OpenRefine.[29] Several bots are available to update data automatically, as well as services simplify export, query, and visualization (UF-A), and interfaces for data entry and query (UF-U). All changes to the knowledge base are carefully tracked (PM) Unfortunately, Wikibase cannot be used with any external RDF vocabularies, and QIDs and PIDs are always assigned to terms (entities and predicates). hampering its reusability in a variety of projects wherein other types of vocabularies (custom, popular, community-based, or w3-endorsed) are necessary (SR).

5 DISCUSSION

We collected requirements to develop CLEF from two paradigmatic case studies, which address collaborations between scholars and cultural heritage institutions, an increasingly common scenario in the landscape of Linked Data and Cultural Heritage [14].

Such requirements have previously been tackled by other projects or software solutions. Each of these projects address certain of the requirements, but not all. In this section, we first discuss strengths and shortcomings of prior works, and then we show how CLEF addresses these problems. Secondly, we highlight—where applicable— the benefits and pitfalls of using Linked Open Data technologies for this task.

User-friendly ways to build interfaces. Among surveyed systems, ResearchSpace offers the most powerful, flexible templating system and engaging visual aids for data manipulation and annotation. However, it requires expert users to set up all the browsing interfaces by means of a custom markup language. This aspect may negatively affect organizations that do not have specialized personnel available for the entire duration of the project. In CLEF we minimize the need of technical expertise to the specification of ontology terms underneath the templating system, which assumes the administrator has basic knowledge of ontology specifications, and the optional customization of browsing templates. All browsing templates are provided with a default HTML configuration, which allows users to benefit from CLEF exploratory interfaces from day zero, without requiring any technical expertise. Moreover, the HTML templating system used by CLEF is implemented in pure Python+HTML. Therefore, in case users need to modify the functionalities or the look and feel of the web pages, a programmer would not require any particular knowledge of custom (enterprise) languages.

Provenance management. Despite being a popular solution, Omeka S does not shine in guaranteeing consistency, accuracy, and continuity of data produced. The lack of editorial control and the flexibility in hiding the visibility of resources identified with persistent URIs are two important shortcomings that affect organizational workflows and reliability of data sources. Likewise, Sinopia does not provide built-in methods to track provenance and support editors in guaranteeing accuracy of records. On the other hand, LED provides fine-grained provenance description, which is stored and queried along with content data, and reuses state-of-the-art Semantic web standards, such as the PROV ontology and RDF named graphs. LED was an inspiration to develop the provenance management strategy for CLEF. Moreover, the integration of CLEF with GitHub allowed us to move a step forward toward an even more detailed provenance tracking mechanism. The download of records as RDF/ttl files in a Git repository, and the commit of every change done to records modified via the web application, enables a refined editorial workflow, in which expert users can grasp changes made on individual triples/form fields.

Data management integration. Considering the sustainability of data produced by projects, we observed that none of the prior solutions really tackles data management workflows by taking into account well-known good practices. ResearchSpace allows versioning of static assets like ontologies and vocabularies, but does not provide versioning mechanisms for data. LED adopts versioning mechanisms and provides a dump of data on a daily base, but these are not citable via a DOI. Since no description is provided, we assume such mechanisms are not

---

[29]https://openrefine.org/.

integrated with the platform for data collection, therefore requiring extra effort in developing dissemination and preservation strategies. OmekaS allows data exports, but no mechanism for creating releases is active. As a matter of fact, most projects reusing such software solutions do not provide versioned, citable, open data.[30] CLEF solves this problem by integrating with common data management workflows. Users can do a one-time setup of their repository for data backup, and connect an OAuth application to the instance of CLEF to enable user authentication (if needed). In turn, GitHub and Zenodo can be synchronized, associating a versioned DOI to each release. Likewise, web sources relevant to collected data can be flagged and automatically sent to the Internet Archive Wayback Machine. These solutions allow us to close the circle of data FAIRness, ensuring long-term preservation of digital archives. Currently, the setup of GitHub and Zenodo is delegated to the user, who may decide not to publish their data, or to store data in a private repository.

Reusability and Sustainability. Solutions like LED are not available for reuse, limiting their benefits to specific projects or user groups. CLEF, on the other hand, is available as an open source project for reuse, and an active community of Polifonia project members contribute to the development. The development of fully-fledged data management systems requires a significant amount of resources, skills, and time, which is often not sustainable in the long-term maintenance of a project. Rather, CLEF economically relies on established and popular solutions, therefore minimizing the effort in maintaining the software to its core functionalities and improving its sustainability over time.

CLEF has a few known limitations. First, CLEF has so far been used by the members of ARTchives and Polifonia, and the development of CLEF has been focused on user-friendliness for editors. However, user-friendliness for end users is also important, in particular given that CLEF is intended for use by lay persons. Currently, the default configuration of CLEF allows end users to search records via text search and by means of filtering mechanisms. Tests with undergraduates are planned in the next months and will show how well it performs when used by such persons, and what are additional User Interface requirements to improve user-friendliness. Second, CLEF benefits greatly from a tight integration with GitHub. Not all cultural institutions and scholarly projects are, however, familiar with GitHub. It remains to be seen whether this is a barrier to uptake. Third, sharing CLEF as an open source project provides a valuable service to cultural institutions and projects. However, it also comes with a responsibility. We must ensure that CLEF remains in active development, to at the very least ensure that it continues to work and does not leave its users vulnerable to security issues. The project is maintained by a community of institutions involved in Polifonia, including research centres, public bodies (e.g., the Italian Ministry of Cultural Heritage), and scholars. The consortium ensures development for the next 2 years, and the Digital Humanities Advanced Research Centre (/DH.arc), University of Bologna, ensures its maintenance in the long run.

Other limitations are due to pitfalls of technological choices. A particular issue of interest is the reuse of ontologies, which are reused via links to external services, rather than by importing them. This avoids duplication and update issues, and allows CLEF to benefit from development work and expertise of third parties. But it also creates a dependency, potentially causing issues if those services become unavailable, either temporarily or permanently. Changes to externally hosted ontologies could also have negative effects on the data quality, if the meaning of terms is changed. This is a well-known problem in the ontology engineering community [4], which has not yet agreed on a shareable policy.

Lastly, there are several benefits in relying on LOD principles, e.g., it allows for decoupling with no hard dependencies and seamless integration between systems that can link to each other via data URIs. Nonetheless, mechanisms for migrating data from museums' legacy content management systems to the triplestore of CLEF are not implemented at the moment. In future works we will address this issue to allow a smooth transition.

---

[30]We reviewed the list of projects adopting ResearchSpace (https://researchspace.org/projects-and-collaborations/) and Omeka S (https://omeka.org/s/directory/), which revealed that the majority of research products do not release (and share) their data via any dissemination or long-term preservation platform, such as GitHub and Zenodo.

6 CONCLUSION

In the Introduction, we emphasized the importance to cultural heritage applications of user-friendliness, provenance management, and integration with existing data management workflows to ensure reusability and sustainability. Via the use cases, we distilled these broad principles into specific requirements, namely reusability, accessibility, continuity, findability and preservation. In the previous section, we demonstrated how CLEF answers each of these requirements. This enables CLEF to support the gathering of cultural heritage data in a user-friendly manner while ensuring good provenance. In addition, CLEF leverages the benefits of Linked Open Data to ensure FAIRness of the data, encouraging reuse. CLEF has been designed to be integrated with common data management workflows. In particular, the integration with GitHub allows adopters to maximise data reuse by offering, along with a SPARQL endpoint, an automatically generated, versioned, data dump, that can be easily linked to Zenodo and assigned a DOI, ensuring its long-term preservation and citability. Finally, CLEF is available as an open source project, making it reusable, thus enabling good quality data gathering for cultural institutions and scholarly projects not in possession of the skills or resources to develop such tools themselves.

Future work will address known limitations and design improvements, driven by user studies with a broader audience. These include tests on usability of interfaces and development of mechanisms for automatic GitHub integration with CLEF and with local storage systems. Moreover, the agile and easy to integrate code base of CLEF offers the opportunity to experiment along new research lines. In particular, methods to discover new web resources based on the ones already included in catalogues (e.g., online music resources in musoW), and methods to enrich existing records with new information (e.g., auto-fill in of fields describing contents of archival collections) will be developed to support adopters in their daily tasks. Lastly, to fill the gap in data preservation and close the data life-cycle, methods to export and donate data to Wikidata will be implemented.

REFERENCES

[1] Alessandro Adamou, Simon Brown, Helen Barlow, Carlo Allocca, and Mathieu d'Aquin. 2019. Crowdsourcing linked data on listening experiences through reuse and enhancement of library data. *International Journal on Digital Libraries* 20, 1 (2019), 61–79.

[2] Alia Amin, Jacco Van Ossenbruggen, Lynda Hardman, and Annelies van Nispen. 2008. Understanding cultural heritage experts' information seeking needs. In *Proceedings of the 8th ACM/IEEE-CS Joint Conference on Digital Libraries*. 39–47.

[3] Chiara Bonacchi, Andrew Bevan, Adi Keinan-Schoonbaert, Daniel Pett, and Jennifer Wexler. 2019. Participation in heritage crowdsourcing. *Museum Management and Curatorship* 34, 2 (2019), 166–182.

[4] Valentina Anita Carriero, Marilena Daquino, Aldo Gangemi, Andrea Giovanni Nuzzolese, Silvio Peroni, Valentina Presutti, and Francesca Tomasi. 2020. The landscape of ontology reuse approaches. *Applications and Practices in Ontology Design, Extraction, and Reasoning* 49 (2020), 21.

[5] Jeremy J. Carroll, Christian Bizer, Pat Hayes, and Patrick Stickler. 2005. Named graphs. *Journal of Web Semantics* 3, 4 (2005), 247–267.

[6] Davide Ceolin, Paul Groth, Valentina Maccatrozzo, Wan Fokkink, Willem Robert Van Hage, and Archana Nottamkandath. 2016. Combining user reputation and provenance analysis for trust assessment. *Journal of Data and Information Quality* 7, 1–2 (2016), 1–28.

[7] World Wide Web Consortium. 2014. Best practices for publishing linked data.

[8] Enrico Daga, Luigi Asprino, Rossana Damiano, Marilena Daquino, Belen Diaz Agudo, Aldo Gangemi, Tsvi Kuflik, Antonio Lieto, Mark Maguire, Anna Maria Marras, Delfina Martinez Pandiani, Paul Mulholland, Silvio Peroni, Sofia Pescarin, and Alan Wecker. 2022. Integrating citizen experiences in cultural heritage archives: Requirements, state of the art, and challenges. *ACM Journal on Computing and Cultural Heritage* 15, 1 (2022), 1–35.

[9] Marilena Daquino. 2021. Linked Open Data native cataloguing and archival description. *Linked Open Data Native Cataloguing and Archival Description* 12, 3 (2021), 91–104.

[10] Marilena Daquino, Enrico Daga, Mathieu d'Aquin, Aldo Gangemi, Simon Holland, Robin Laney, Albert Merono Penuela, and Paul Mulholland. 2017. Characterizing the landscape of musical data on the Web: State of the art and challenges. WHiSe@ISWC, CEUR-WS.org, online ceur-ws.org/Vol-2014.

[11] Marilena Daquino, Lucia Giagnolini, and Francesca Tomasi. 2021. ARTchives: A linked open data native catalogue of art historians' archives. In *Proceedings of the Theory and Practice of Digital Libraries*.

[12] Marilena Daquino, & Martin Hlosta. 2022. CLEF (v1.0.0). Zenodo. https://doi.org/10.5281/zenodo.6423933

[13] Marilena Daquino, Francesca Mambelli, Silvio Peroni, Francesca Tomasi, and Fabio Vitali. 2017. Enhancing semantic expressivity in the cultural heritage domain: Exposing the zeri photo archive as linked open data. *Journal on Computing and Cultural Heritage* 10, 4 (2017), 1–21.

[14] Edie Davis and Bahareh Heravi. 2021. Linked data and cultural heritage: A systematic review of participation, collaboration, and motivation. *Journal on Computing and Cultural Heritage* 14, 2 (2021), 1–18.

[15] Kelly Davis. 2019. Old metadata in a new world: Standardizing the Getty Provenance Index for linked data. *Art Libraries Journal* 44, 4 (2019), 162–166.

[16] Tom De Nies, Sara Magliacane, Ruben Verborgh, Sam Coppens, Paul Groth, Erik Mannens, and Rik Van de Walle. 2013. Git2PROV: Exposing version control system content as W3C PROV. In *Proceedings of the International Semantic Web Conference*. 125–128.

[17] Corine Deliot. 2014. Publishing the British national bibliography as linked open data. *Catalogue & Index* 174 (2014), 13–18.

[18] Emmanuelle Delmas-Glass and Robert Sanderson. 2020. Fostering a community of PHAROS scholars through the adoption of open standards. *Art Libraries Journal* 45, 1 (2020), 19–23.

[19] Dennis Diefenbach, Max De Wilde, and Samantha Alipio. 2021. Wikibase as an infrastructure for knowledge graphs: The eu knowledge graph. In *Proceedings of the International Semantic Web Conference*. Springer, 631–647.

[20] Chris Dijkshoorn, Victor De Boer, Lora Aroyo, and Guus Schreiber. 2017. Accurator: Nichesourcing for cultural heritage. *Human Computation* 6, 1 (2017), 101–131.

[21] Chris Dijkshoorn, Lizzy Jongma, Lora Aroyo, Jacco Van Ossenbruggen, Guus Schreiber, Wesley Ter Weele, and Jan Wielemaker. 2018. The Rijksmuseum collection as linked data. *Semantic Web* 9, 2 (2018), 221–230.

[22] María de la Paz Diulio, Juan Cruz Gardey, Analía Fernanda Gomez, and Alejandra Garrido. 2021. Usability of data-oriented user interfaces for cultural heritage: A systematic mapping study. *Journal of Information Science* 49, 2 (2021), 01655515211001787.

[23] Martin Doerr, Stefan Gradmann, Steffen Hennicke, Antoine Isaac, Carlo Meghini, and Herbert Van de Sompel. 2010. The europeana data model (edm). In *Proceedings of the World Library and Information Congress: 76th IFLA General Conference and Assembly*.

[24] The W3C SPARQL Working Group. 2013. *SPARQL 1.1 Overview*. W3C Recommendation. W3C. Retrieved from https://www.w3.org/TR/2013/REC-sparql11-overview-20130321/.

[25] Peter Haase, Daniel M. Herzig, Artem Kozlov, Andriy Nikolov, and Johannes Trame. 2019. metaphactory: A platform for knowledge graph management. *Semantic Web* 10, 6 (2019), 1109–1125.

[26] Manfred Hauswirth, Marcin Wylot, Martin Grund, Paul Groth, and Philippe Cudré-Mauroux. 2017. Linked data management. In *Proceedings of the Handbook of Big Data Technologies*. Springer, 307–338.

[27] Tom Heath and Christian Bizer. 2011. Linked data: Evolving the web into a global data space. *Synthesis Lectures on the Semantic Web: Theory and Technology* 1, 1 (2011), 1–136.

[28] Sorin Hermon and Franco Niccolucci. 2021. FAIR data and cultural heritage special issue editorial note. *International Journal on Digital Libraries* 22, 3 (2021), 251–255.

[29] José-Miguel Herrera, Aidan Hogan, and Tobias Käfer. 2019. BTC-2019: The 2019 billion triple challenge dataset. In *Proceedings of the International Semantic Web Conference*. Springer, 163–180.

[30] Craig A. Knoblock, Pedro Szekely, Eleanor Fink, Duane Degler, David Newbury, Robert Sanderson, Kate Blanch, Sara Snyder, Nilay Chheda, Nimesh Jain, Ravi Raju Krishna, Nikhila Begur Sreekanth, and Yixiang Yao. 2017. Lessons learned in building linked data for the American art collaborative. In *Proceedings of the International Semantic Web Conference*. Springer, 263–279.

[31] Zois Koukopoulos, Dimitrios Koukopoulos, and Jason J. Jung. 2017. A trustworthy multimedia participatory platform for cultural heritage management in smart city environments. *Multimedia Tools and Applications* 76, 24 (2017), 25943–25981.

[32] Anna-Lena Lamprecht, Leyla Garcia, Mateusz Kuzak, Carlos Martinez, Ricardo Arcila, Eva Martin Del Pico, Victoria Dominguez Del Angel, Stephaniei van de Sandt, Jon Ison, Paula Andrea Martinez, Peter McQuilton, Alfonso Valencia, Jennifer Harrow, Fotis Psomopoulos, Josep Ll. Gelpi, Neil Chue Hong, Carole Goble, and Salvador Capella-Gutierrez. 2020. Towards FAIR principles for research software. *Data Science* 3, 1 (2020), 37–59.

[33] Timothy Lebo, Satya Sahoo, Deborah McGuinness, Khalid Belhajjame, James Cheney, David Corsar, Daniel Garijo, Stian Soiland-Reyes, Stephan Zednik, and Jun Zhao. 2013. Prov-o: The prov ontology. W3C recommendation 30.

[34] Malmsten Martin. 2009. Exposing library data as linked data. Paper Presented at the IFLA Satellite Preconference Sponsored by the Information Technology Section Emerging Trends in Technology: Libraries Between Web 2.0, Semantic Web and Search Technology, 19–20.

[35] Nandana Mihindukulasooriya, Raúl García-Castro, and Miguel Esteban Gutiérrez. 2013. Linked data platform as a novel approach for enterprise application integration. *COLD* 1034 (2013), 146–157.

[36] Luc Moreau. 2010. *The Foundations for Provenance on the Web*. Now Publishers Inc.

[37] Fabrizio Orlandi, Damien Graux, and Declan O'Sullivan. 2021. Benchmarking RDF metadata representations: Reification, singleton property and RDF. In *Proceedings of the 2021 IEEE 15th International Conference on Semantic Computing*. IEEE, 233–240.

[38] Michael Röder, Denis Kuchelev, and Axel-Cyrille Ngonga Ngomo. 2020. HOBBIT: A platform for benchmarking big linked data. *Data Science* 3, 1 (2020), 15–35.

[39] Robert J. Sandusky. 2016. Computational provenance: Dataone and implications for cultural heritage institutions. In *Proceedings of the 2016 IEEE International Conference on Big Data*. IEEE, 3266–3271.

[40] Leslie F. Sikos and Dean Philp. 2020. Provenance-aware knowledge representation: A survey of data models and contextualized knowledge graphs. *Data Science and Engineering* 5, 3 (2020), 293–316.

[41] Bryan Thompson, Mike Personick, and Martyn Cutcher. 2016. The Bigdata® RDF graph database. In *Proceedings of the Linked Data Management*. Chapman and Hall/CRC, 221–266.

[42] Pierre-Yves Vandenbussche, Ghislain A. Atemezing, María Poveda-Villalón, and Bernard Vatant. 2017. Linked open vocabularies (LOV): A gateway to reusable semantic vocabularies on the Web. *Semantic Web* 8, 3 (2017), 437–452.

[43] Mark D. Wilkinson, Michel Dumontier, IJsbrand Jan Aalbersberg, Gabrielle Appleton, Myles Axton, Arie Baak, Niklas Blomberg, Jan-Willem Boiten, Luiz Bonino da Silva Santos, Philip E. Bourne. 2016. The FAIR guiding principles for scientific data management and stewardship. *Scientific Data* 3, 1 (2016), 1–9.