

Embracing Semantic Technology for Better Metadata Authoring in Biomedicine

Attila L. Egyedi, Martin J. O'Connor, Marcos Martínez-Romero, Debra Willrett, Josef Hardi, John Graybeal, and Mark A. Musen

Stanford Center for Biomedical Informatics Research
Stanford University, Stanford, CA 94305, USA
attila.egyedi@stanford.edu

Abstract. The Center for Expanded Data Annotation and Retrieval (CEDAR) has developed a suite of tools and services that allow scientists to create and publish metadata describing scientific experiments. Using these tools and services—referred to collectively as the CEDAR Workbench—scientists can collaboratively author metadata and submit them to public repositories. A key focus of our software is semantically enriching metadata with ontology terms. The system combines emerging technologies, such as JSON-LD and graph databases, with modern software development technologies, such as microservices and container platforms. The result is a suite of user-friendly, Web-based tools and REST APIs that provide a versatile end-to-end solution to the problems of metadata authoring and management. This paper presents the architecture of the CEDAR Workbench and focuses on the technology choices made to construct an easily usable, open system that allows users to create and publish semantically enriched metadata in standard Web formats.

Keywords: Metadata, Metadata Management, Ontologies, Semantic Web.

1 Introduction

In the life sciences, high-quality metadata are important for discovering experimental datasets, for understanding how the associated experiments were carried out, and for reproducing those experiments. Funding agencies and journals increasingly demand that descriptive metadata accompany published datasets, which has led to a dramatic increase in the volume of available metadata. Unfortunately, this increasing volume of metadata has not been matched with an equivalent quality increase. The quality of public metadata continues to be very poor. As a result, the data curation and preprocessing effort can form a significant portion of knowledge discovery costs.

There is a growing awareness that metadata quality needs to be significantly improved [1, 2]. The literature on improving metadata quality generally focuses on the need for better practices and infrastructure for authoring metadata. Infrequent use of ontologies to control metadata field names and values and lack of validation have been identified as key problems [3]. The recently defined FAIR data principles [4]

specify a set of desirable criteria that metadata and their corresponding datasets should meet to enhance their discovery and reusability. Use of controlled terms and Linked Open Data technologies are central to the FAIR principles.

The biomedical community has developed several tools to address the challenge of producing high-quality metadata. These tools typically focus on supporting so-called *minimal information* metadata guidelines [5]. These guidelines specify the minimum information about experimental data necessary to ensure that the associated experiments can be reproduced. One of the first minimum information-focused systems was ISA Tools [6], which provides a desktop application that allows users to construct spreadsheet-based submissions for metadata repositories. The linkedISA [7] evolution of this software added mechanisms to annotate submissions with ontology terms. RightField [8], an Excel-based plugin, also allows users to embed ontology terms in spreadsheets, and to restrict cell values to terms from ontologies. A similar desktop application called Annotare [9] focuses on submissions to ArrayExpress [10].

These tools, while powerful, often require a significant amount of complex configuration by specialists to generate spreadsheet-based metadata submissions. They also do not provide a collaborative platform to support Web-based metadata management. There is a need for a solution that can be used by non-specialist users and that addresses the metadata problem in a holistic manner. The system should support the creation and submission of metadata that is based on open Web-based standards, conforms to the FAIR recommendations, and interoperates with Linked Open Data.

The Center for Expanded Data Annotation and Retrieval (CEDAR) [11] has developed such a system. The system—referred to as the CEDAR Workbench (see Fig. 1)—is focused on creating *templates* that define the structure and semantics of metadata specifications. CEDAR provides a metadata workflow from template creation, to metadata authoring, and to final submission to public databases. We outline the architecture of the CEDAR Workbench and illustrate how we combine standard Web-tier technologies with semantic technologies to produce an end-to-end platform for creating and publishing high-quality, semantically enriched metadata.

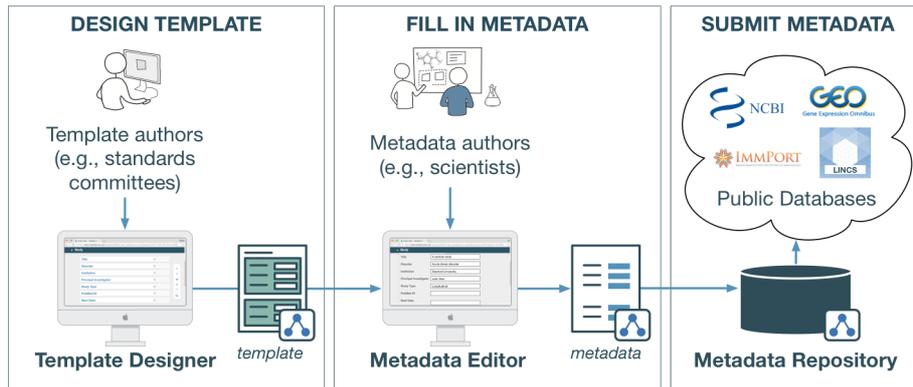


Fig. 1. CEDAR’s template-based metadata authoring workflow. Template authors use the Template Designer to create templates. These templates are used by the Metadata Editor to automatically generate form-based interfaces that scientists use to create metadata. The Metadata Repository stores the acquired metadata prior to their submission to public databases.

2 Architecture of the CEDAR Workbench

The CEDAR Workbench is implemented as a modular microservice-based system (see Fig. 2). A formal model to represent metadata artifacts—referred to as the Template Model—serves as a foundation for the system [12]. Using this model, the system provides services for creating and managing these artifacts. A collection of Web-based tools then uses these services to provide a user-friendly metadata management platform. We now present the architecture of this system. We first outline the model and then describe how the various system components use this model to provide services that can be used to create and publish semantically annotated metadata.

2.1 CEDAR Template Model

CEDAR’s primary goal is to generate high-quality metadata describing scientific data sets that are semantically enriched with terms from ontologies. As mentioned, CEDAR uses templates to define metadata specifications. Templates are structural specifications of metadata and define the attributes (called *template fields* or *fields*) needed to describe scientific experiments. For example, an *Experiment* template may have a *disease* field containing the name of the disease studied by an experiment. To facilitate reuse, the model allows templates to be composed from existing templates. The goal is to support the development of libraries of templates that can be reused by template authors. The model also specifies a set of provenance fields for templates that provide support for attribution and auditing.¹

For interoperability on the Web, we designed an open standards-based model for representing templates and metadata that can be serialized to widely accepted Web-based formats [12]. We identified two key Web-centric technologies that can be combined to meet this goal: JSON Schema (<http://json-schema.org/>) and JSON-LD (<https://json-ld.org/>). JSON Schema is used to represent all structural aspects of CEDAR’s Template Model. A JSON Schema-based CEDAR template effectively provides a structural specification for metadata. These metadata are encoded using JSON-LD. JSON-LD provides mechanisms to add semantic annotations to JSON documents that can restrict the types and values of fields to terms from ontologies. The use of JSON-LD provided a bridge between the model and semantic technologies. JSON-LD is effectively an RDF serialization, so CEDAR can use off-the-shelf tools to export metadata in a variety of RDF formats. CEDAR’s JSON Schema–and JSON-LD–based model is used by all CEDAR services and front end tools.

2.2 CEDAR Open Services

All CEDAR services are implemented as microservices. The services are written in Java and use the Dropwizard framework (<http://www.dropwizard.io/>) to provide REST-based APIs. These APIs² are used by all CEDAR front end components and can also be used directly by third-party applications. CEDAR services can be broadly

¹ A full model specification is available at <http://metadatascenter.org/cedar-template-model>.

² CEDAR REST APIs are documented at <https://resource.metadatascenter.org/api/>.

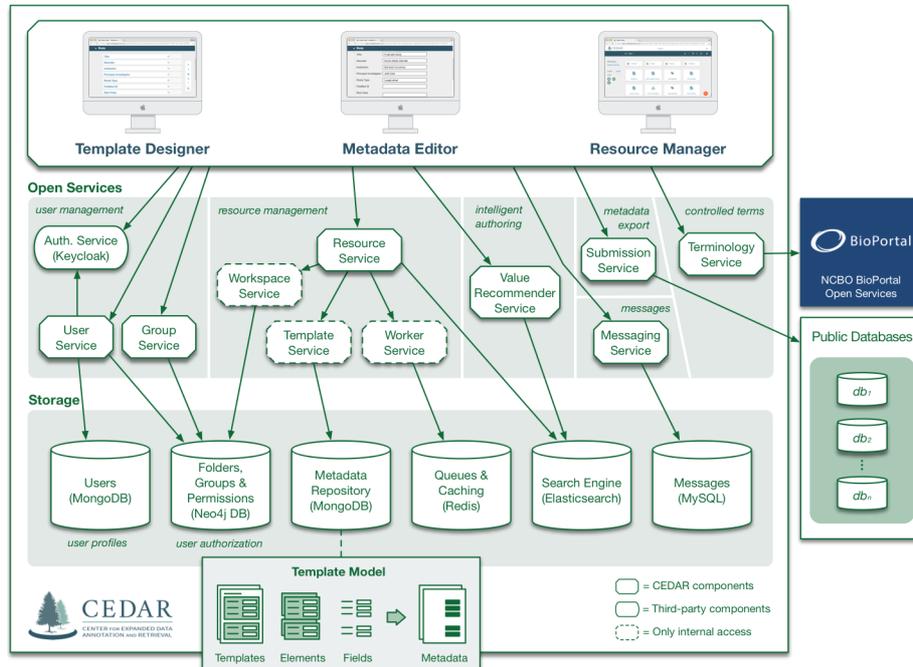


Fig. 2. Architecture of the CEDAR Workbench. All metadata resources adhere to the CEDAR Template Model. A *Storage* layer provides persistence services for metadata resources. These resources are stored in the Metadata Repository. An *Open Services* layer features components for managing resources, including a Resource Service for managing metadata resources, groups, and permissions, a Submission Service that allows users to upload metadata to external databases, and a Terminology Service that provides a link to the BioPortal ontology repository. The *Front End* layer includes a Template Designer for creating templates, a Metadata Editor for entering metadata, and a Resource Manager for managing templates and metadata.

divided into two functional groups: (1) metadata repository services, which provide storage and management functionality for templates and metadata, and (2) metadata enrichment and submission services, which assist in generating semantically rich metadata and submitting the generated metadata to public databases.

Metadata Repository Services. Three microservices—the Template, Workspace, and Resource services—provide CEDAR’s metadata repository functionality.

Template Service. The Template Service acts as the main entry point to the Metadata Repository. It is responsible for managing templates and metadata content. Since the CEDAR Template Model is serialized as JSON by default, a JSON-based database was the natural choice for the data persistence layer of this component. We used MongoDB (<https://www.mongodb.com/>) because of its proven record, though any equivalent JSON-based database could be used. Templates and metadata are stored directly as model-conforming JSON Schema and JSON-LD artifacts, respectively.

The Template Service publishes several REST endpoints that provide standard operations for these artifacts. Templates and metadata are effectively passed through the REST layer as is with minimal bookkeeping transformations that mainly involve setting provenance information for the artifacts. Apart from this transformation, the incoming resource is stored almost verbatim in the persistence layer. This minimalist approach allows for a small, lightweight microservice.

Workspace Service. The Workspace Service is a Metadata Repository service responsible for providing management functionality for the templates and metadata resources stored in the Template Service. We decided to create a filesystem-like structure to organize these resources. This structure was loosely modeled on the Unix file system and on the resource organization functionality provided by Google Drive. The Workspace Service is responsible for providing resource management based on this structure, and is also responsible for providing permissions and resource sharing functionality. Users can organize their resources using folders and those folders can be shared with other users. CEDAR also supports the creation of groups, which can be used for resource sharing. Separate User and Group services provide REST-based operations on users and groups, respectively.

We decided to use a graph database to implement the Workspace Service, since this technology natively offers the graph traversal and recursion queries necessary for working with hierarchical information. We considered other NoSQL solutions, but decided that the elegant, native support for graph-based queries offered by graph databases would allow us to naturally represent a variety of resources and the relationships among them. From the various available graph database solutions, we picked Neo4j, primarily because of its broad popularity but also because preliminary tests convinced us that it offers full coverage for the types of queries we require in our system. The example in Fig. 3 shows how the system uses Neo4j and MongoDB to represent a scenario with users, groups, permissions, folders, templates, and metadata.

Resource Service. We described how the Template Service handles resource storage and the Workspace Service adds a management layer for those resources. An aggregator service called the Resource Service provides a unified interface to these components. The goal of this service is to act as a main Metadata Repository entry point for REST operations. It ensures data consistency by orchestrating the various operations performed by the Template Service and Workspace Service. A typical REST operation executed by the Resource Service is performed according to the following steps: (1) user authentication and authorization (via the User Service); (2) input validation; (3) preconditions checking; (4) calls to the Template and Workspace services; and (5) response assembly. If any of these steps fail, the REST call will fail.

The Resource Service also provides search capabilities. This service supports complex searches on template field names and values and it also allows users to find template and metadata resources that are shared with them. The index-based Elasticsearch engine (<https://www.elastic.co>) was used to support search. The Resource Service is responsible for supplying index data to Elasticsearch and for ensuring that the search index is kept up to date. This service considers resource permissions when performing

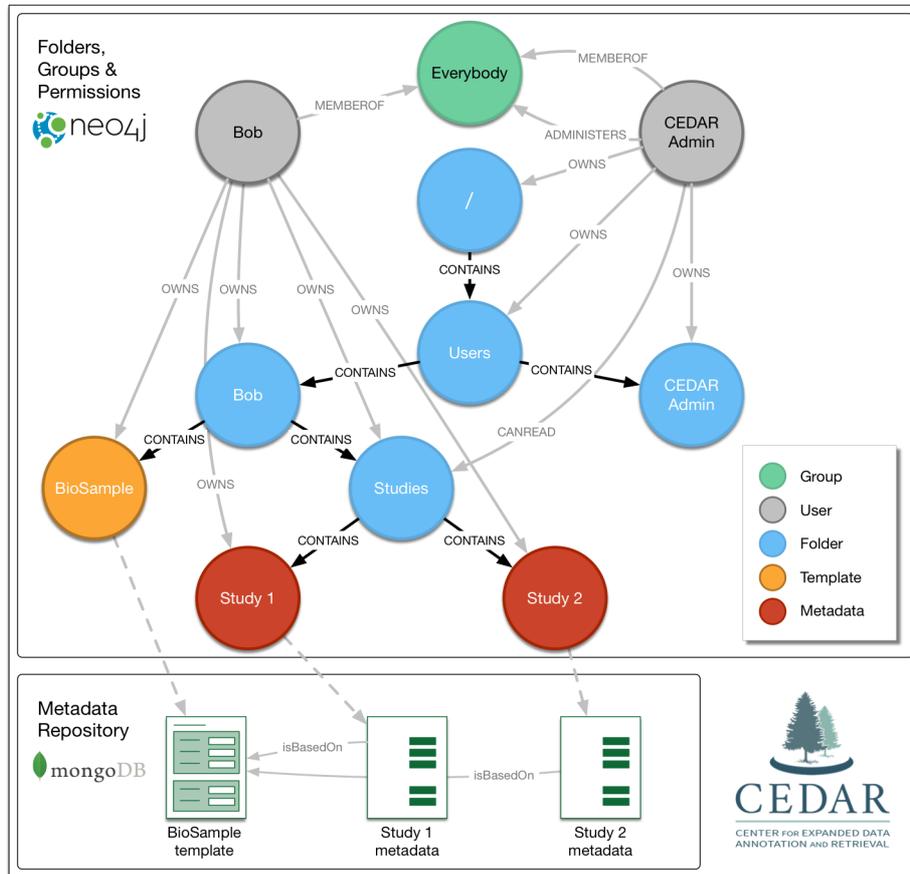


Fig. 3. Illustration of CEDAR's representation of users, groups, folders, templates, and metadata in the Neo4j graph database, together with the linkage of those resources with their serializations in the MongoDB JSON-based database. The figure shows a simple folder hierarchy for a user called Bob. Bob has a template called *BioSample* in his home folder and a subfolder *Studies* that contains metadata resources *Study 1* and *Study 2*.

searches. To ensure rapid searches, a separate permission index is stored in Elasticsearch to hold the user and group permissions. By joining the content index with the permission index, the CEDAR Workbench can execute the search queries and take permissions into account in one step. The maintenance of this permission index adds some complexity but helps ensure rapid permissions-based search.

Metadata Enrichment and Submission Services. CEDAR provides several services to help authors semantically enrich metadata and submit them to public repositories.

Terminology Service. The CEDAR Workbench provides an interactive lookup service that makes it possible to enrich biomedical metadata with ontology terms selected

from BioPortal. BioPortal, developed by the National Center for Biomedical Ontology (NCBO) [13], is a popular platform for hosting biomedical ontologies that provides more than 650 ontologies and terminologies, with over 8 million classes and 64,000 properties. CEDAR uses the Terminology Service to search for ontology terms to annotate templates—that is, to add type and property assertions using ontology classes and properties [14]. Users can also specify that the possible values of fields must correspond to ontology terms. The system effectively allows template designers to constrain field values to any combination of (1) all classes in an ontology branch, (2) all classes from a specific ontology, (3) specific classes, and (4) value sets. We are studying how to enhance the Terminology Service with intelligent term suggestion capabilities based on the NCBO Ontology Recommender service [15], which will help users to find the most appropriate ontology terms to annotate their templates. When appropriate terms do not exist, users can create new terms and value sets dynamically at template design-time.

While BioPortal is currently the only ontology repository supported by the Terminology Service, we plan to extend it to work with third-party repositories. Users may also upload domain-specific ontologies to BioPortal, thus making them available for use by CEDAR. A CEDAR deployment can also be configured to use other BioPortal installations, which can contain custom user-managed collections of ontologies.

Value Recommender Service. CEDAR provides an intelligent authoring functionality designed to decrease metadata authoring time and improve metadata quality. A Value Recommender service uses ontology-based metadata specifications combined with analyses of previously entered metadata to generate suggestions for filling out metadata templates [16]. These suggestions are context-sensitive, meaning that the values predicted for a field are generated and ranked based on the values entered for other fields in the template. During metadata entry, the recommender provides the user with a ranked list of suggested values for each template field.

For example, suppose that a *Study* template contains the fields *tissue* and *disease*, and that the user fills out the *tissue* field with the value *liver*. Then, when filling out the *disease* field, the Value Recommender would suggest diseases that affect the liver, such as *cirrhosis* or *hepatitis A*. For plain text metadata, the recommender suggests textual values. For ontology-based metadata, it suggests ontology term identifiers supplied by the Terminology Service. These suggestions are presented using a user-friendly label defined in the source ontology (e.g., *hepatitis A* is the preferred label for the class http://purl.obolibrary.org/obo/DOID_12549 in the Human Disease Ontology). The suggestions are generated in real time as template fields are being filled in. Additionally, we are developing new methods that will use the analyses performed by the Value Recommender to identify potential mistakes during metadata entry. For example, if the user enters *liver* for the *tissue* field and then enters *colorectal cancer* for the *disease* field, the system will warn of a possible inconsistency.

Submission Service. The Submission Service supports submission of metadata from CEDAR to external repositories. Repository-specific code is provided in this service to achieve submissions since there is no global standard for metadata submission. A

The screenshot shows a web-based interface for editing metadata. At the top, there are navigation options for 'Project and Samples', 'Project', and 'Sample'. Below this is a table with the following columns: 'Tissue', 'Phenotype', and 'Cell type'. The table contains 10 rows of data. A dropdown menu is open over the 'Tissue' column of the second row, showing a list of suggestions: 'amoebocyte', 'B95-8 cell', 'basophil', 'blast cell', 'blood', and 'blood cancer cell'. The first row has 'peripheral blood' in the Tissue column, 'AchR' in the Phenotype column, and 'B cell' in the Cell type column. The second row has an empty Tissue field, 'AchR' in the Phenotype column, and 'B cell' in the Cell type column. The third row has 'amoebocyte' in the Tissue column, 'MuSK' in the Phenotype column, and 'B cell' in the Cell type column. The fourth row has 'B95-8 cell' in the Tissue column, 'MuSK' in the Phenotype column, and 'B cell' in the Cell type column. The fifth row has 'basophil' in the Tissue column, 'MuSK' in the Phenotype column, and 'B cell' in the Cell type column. The sixth row has 'blast cell' in the Tissue column, 'MuSK' in the Phenotype column, and 'B cell' in the Cell type column. The seventh row has 'blood' in the Tissue column, 'MuSK' in the Phenotype column, and 'B cell' in the Cell type column. The eighth row has 'blood cancer cell' in the Tissue column, 'MuSK' in the Phenotype column, and 'B cell' in the Cell type column. The ninth row has 'peripheral blood' in the Tissue column, 'not applicable' in the Phenotype column, and 'B cell' in the Cell type column. The tenth row is partially visible and has 'peripheral blood' in the Tissue column.

	Tissue	Phenotype	Cell type	
1	peripheral blood	AchR	B cell	MK05
2		AchR	B cell	MK05
3	amoebocyte	MuSK	B cell	MK04
4	B95-8 cell	MuSK	B cell	MK04
5	basophil	MuSK	B cell	AR05
6	blast cell	MuSK	B cell	AR05
7	blood	MuSK	B cell	AR02
8	blood cancer cell	MuSK	B cell	AR02
9	peripheral blood	not applicable	B cell	AR04
10	peripheral blood			

Fig. 4. Metadata Editor screenshot showing value suggestions for a spreadsheet-based *Sample* field. Here, the *Tissue* column is restricted to controlled terms from the BRENDA Tissue and Enzyme Source Ontology (BTO). Suggestions are retrieved in real time from the BioPortal ontology repository via the Terminology Service. The IRIs of selected terms are then stored in the final metadata and encoded using JSON-LD.

uniform interface is presented through the server's REST APIs. Currently, the CEDAR Workbench has initial submission pipelines for NCBI's BioSample and SRA repositories, together with custom pipelines for several collaborating groups of biomedical investigators. In addition to metadata submission, the submission server also provides a data pass through service that supports the incremental upload of large data files. The Submission Service uses the Messaging Service to provide asynchronous event notifications to users for these long-running data uploads.

2.3 Front End Tools

We developed several highly interactive Web-based tools to manage CEDAR templates and metadata. The Template Designer allows users to create templates. The Metadata Editor tool (see Fig. 4) uses these templates to automatically generate a forms-based acquisition interface for entering metadata. Entered metadata are stored in CEDAR's Metadata Repository. The Resource Manager tool can be used to organize resources into folders and to manage resource permissions and sharing.

A key focus is on interoperation with ontologies. Using interactive Terminology Service-based look-up services, the Template Designer allows template authors to find terms in ontologies to annotate their templates and to restrict the values of template fields. Users entering metadata in the Metadata Editor are prompted in real time with drop-down lists, auto-completion suggestions, and verification hints, significantly reducing their errors while speeding metadata entry. This lookup is driven by the value constraints specified in templates. Semantic markup acquired from users is represented in the generated metadata using standard JSON-LD constructs.

3 Discussion

This paper outlines a system that combines standard Web-centric software development approaches with semantic technologies to provide an environment for creating and submitting semantically enriched metadata. The system is built on a standards-based model that defines a common format for describing metadata using JSON Schema and JSON-LD. The use of JSON-LD provides a robust bridge between semantic technologies, such as ontologies, and the practical advantages of widely-available Web-centric tooling. JSON-LD also facilitates publishing metadata on the Web in variety of RDF serializations. Similarly, JSON Schema provides a standard technology to represent all structural aspects of CEDAR's Template Model. The use of JSON-based representations also has many practical advantages for system development. Both formats can be easily published and consumed directly via REST APIs based on lightweight microservices. These microservices could directly serialize CEDAR templates and metadata to a JSON-based database. The use of a JSON-based format also eased front end development since JSON is the native format for JavaScript, the dominant language in front end Web development. Finally, the Neo4j graph database provided a very natural representation of the complex relationships needed to organize and share resources. While Neo4j is not currently used to represent semantic relationships between resources, we plan to use it to capture an array of semantically rich relationships, such as version and provenance links.

CEDAR is used by several communities to develop metadata submission pipelines. These groups include (1) the Library of Integrated Network-Based Cellular Signatures (LINCS, <http://www.lincsproject.org>), which is using CEDAR to build an end-to-end metadata management solution; (2) the AIRR community (<http://airr-community.org>), which is developing standards for describing datasets acquired using sequencing technologies; and (3) the Stanford Digital Repository (SDR, <http://sdr.stanford.edu>) in the Stanford University Libraries, which is testing the use of CEDAR templates for creating RDF-encoded metadata describing digital artifacts. These groups have integrated CEDAR into their metadata workflow in a variety of ways. For example, the AIRR submission process involves submitting the generated metadata to the public NCBI BioSample and SRA repositories whereas the LINCS and SDR projects target internal metadata repositories. A common approach for all groups is the use of CEDAR to encode semantically-enriched templates describing their metadata. The resulting template-described metadata is then used in their submission pipelines.

All software and models described in this paper are open source and available on GitHub (<https://github.com/metadatacenter>). A Docker-based installation is also provided. We released a public version of the CEDAR Workbench (<https://cedar.metadatacenter.org>) in April 2017.

Acknowledgments

CEDAR is supported by the National Institutes of Health through the NIH Big Data to Knowledge program under grant 1U54AI117925. NCBO is supported by the NIH Common Fund under grant U54HG004028.

References

1. Bruce, T.R., Hillmann, D.I.: The continuum of metadata quality: defining, expressing, exploiting. In: *Metadata in Practice*. ALA editions (2004).
2. Park, J.-R.: Metadata Quality in Digital Repositories: A Survey of the Current State of the Art. *Cat. Classif. Q.* 47, 213–228 (2009).
3. Gonçalves, R.S., O'Connor, M.J., Martinez-Romero, M., et al.: Metadata in the BioSample Online Repository are Impaired by Numerous Anomalies. In: *1st International Workshop SemSci 2017, co-located with ISWC 2017* (2017).
4. Wilkinson, M.D., Dumontier, M., Aalbersberg, I.J., et al.: The FAIR Guiding Principles for scientific data management and stewardship. *Sci. Data.* 3, 160018 (2016).
5. Tenenbaum, J.D., Sansone, S.-A., Haendel, M.: A sea of standards for omics data: sink or swim? *J. Am. Med. Inform. Assoc.* 21, 200–203 (2014).
6. Rocca-Serra, P., Brandizi, M., Maguire, E., et al.: ISA software suite: Supporting standards-compliant experimental annotation and enabling curation at the community level. *Bioinformatics.* 26, 2354 (2010).
7. González-Beltrán, A., Maguire, E., Sansone, S.-A., et al.: linkedISA: semantic representation of ISA-Tab experimental metadata. *BMC Bioinformatics.* 15 Suppl 1, S4 (2014).
8. Wolstencroft, K., Owen, S., Horridge, M., et al.: RightField: Embedding ontology annotation in spreadsheets. *Bioinformatics.* 27, 2021–2022 (2011).
9. Shankar, R., Parkinson, H., Burdett, T., et al.: Annotare—a tool for annotating high-throughput biomedical investigations and resulting data. *Bioinformatics.* 26, 2470–2471 (2010).
10. Parkinson, H., Sarkans, U., Shojatalab, M., et al.: ArrayExpress—a public repository for microarray gene expression data at the EBI. *Nucleic Acids Res.* 33, D553–D555 (2005).
11. Musen, M.A., Bean, C.A., Cheung, K.H., et al.: The Center for Expanded Data Annotation and Retrieval. *J. Am. Med. Informatics Assoc.* 22, 1148–1152 (2015).
12. O'Connor, M.J., Martinez-Romero, M., Egyedi, A.L., et al.: An open repository model for acquiring knowledge about scientific experiments. In: *Proceedings of the 20th International Conference on Knowledge Engineering and Knowledge Management (EKAW2016)*. pp. 762–777 (2016).
13. Musen, M.A., Noy, N.F., Shah, N.H., et al.: The National Center for Biomedical Ontology. *J. Am. Med. Informatics Assoc.* 19, 190–195 (2012).
14. Martínez-Romero, M., O'Connor, M.J., Dorf, M., et al.: Supporting ontology-based standardization of biomedical metadata in the CEDAR Workbench. In: *Proceedings of the Int Conf Biom Ont (ICBO)* (in press) (2017).
15. Martínez-Romero, M., Jonquet, C., O'Connor, M.J., et al.: NCBO Ontology Recommender 2.0: An Enhanced Approach for Biomedical Ontology Recommendation. *J. Biomed. Semantics.* 8, 21 (2017).
16. Martínez-Romero, M., O'Connor, M.J., Shankar, R., et al.: Fast and accurate metadata authoring using ontology-based recommendations. In: *Proceedings of AMIA 2017 Annual Symposium* (in press) (2017).