Unleashing the value of Common Data Elements through the CEDAR Workbench

Martin J. O'Connor, MSc[1], Denise B. Warzel, MSc[2], Marcos Martínez-Romero, PhD[1], Josef Hardi[1], Debra Willrett, MS[1], Attila L. Egyedi[1], Aras Eftekhari[3], John Graybeal[1], and Mark A. Musen, MD, PhD[1]

[1]Center for Biomedical Informatics Research, Stanford University, Stanford, CA, USA
[2]Cancer Informatics Branch, National Cancer Institute, Bethesda, MD, USA
[3]Attain LLC, McLean, VA, USA

Abstract

Developing promising treatments in biomedicine often requires aggregation and analysis of data from disparate sources across the healthcare and research spectrum. To facilitate these approaches, there is a growing focus on supporting interoperation of datasets by standardizing data-capture and reporting requirements. Common Data Elements (CDEs)—precise specifications of questions and the set of allowable answers to each question—are increasingly being adopted to help meet these standardization goals. While CDEs can provide a strong conceptual foundation for interoperation, there are no widely recognized serialization or interchange formats to describe and exchange their definitions. As a result, CDEs defined in one system cannot be easily be reused by other systems. An additional problem is that current CDE-based systems tend to be rather heavyweight and cannot be easily adopted and used by third-parties. To address these problems, we developed extensions to a metadata management system called the CEDAR Workbench to provide a platform to simplify the creation, exchange, and use of CDEs. We show how the resulting system allows users to quickly define and share CDEs and to immediately use these CDEs to build and deploy Web-based forms to acquire conforming metadata. We also show how we incorporated a large CDE library from the National Cancer Institute's caDSR system and made these CDEs publicly available for general use.

Introduction

The use of ontologies and controlled terminologies has become pervasive in biomedicine. Dozens of large ontologies and terminologies and hundreds of specialized smaller ones have been developed to cover many biomedical domains. For example, the National Center for Biomedical Ontology's BioPortal ontology repository[1] serves over 700 biomedical ontologies and terminologies, which are used throughout biomedicine. While ontologies and controlled terminologies provide a common vocabulary to refer to biomedical concepts, additional detailed specifications are typically needed to satisfy data collection and reporting needs.[2] Data collection requirements for clinical studies, for example, mandate precise specifications of questions and the range of possible answers to those questions. Common data elements (CDEs), which provide the means to link the specification of the question to a range of possible answers, are increasingly being adopted to satisfy these data collection and reporting needs.[2–4]

A CDE is effectively an agreed-upon question specification, precisely defining how a particular question should be asked and what values should be presented to users for their selection.[5,6] An answer is represented as a single typed value. Answers can be strings, in which case simple parameters such as string length and encoding can be specified. They may be numeric, in which case parameters such as valid ranges, units, and precision may be specified. Answers can also come from controlled terminologies, where the answer may be specified as a terminology code or an encoded value. Often a permitted value set, or value domain, is built to define the allowed answers for a question. Such value sets are often built from one or more standards or terminologies.[7,8] CDEs are commonly used to define case report forms (CRFs) for clinical trials, though they can be used in any situation where it is important to meet rigorous data collection or reporting requirements. They have been adopted most widely in cancer research,[9–12] but they are also used in other domains, such as epilepsy,[13] brain injury,[14] stroke,[15] phenotyping,[16] and radiology.[17]

CDE definitions can provide a strong foundation for interoperation because they allow data descriptions to be recorded in a registry. Such registries can help to standardize the way data are collected, stored, transferred, and reported. One of the largest CDE registries has been developed by the U.S. National Cancer Institute (NCI) with the goal of facilitating multidisciplinary, multi-institutional cancer research. This registry is called Cancer Data Standards Repository (caDSR)[9,18] and it contains over 60,000 CDEs that cover many aspects of cancer research. The U.S.

National Institutes of Health (NIH) are also developing a multi-discipline registry that aims to unify the range of biomedical CDEs that have been produced by a variety of NIH and other organizations (https://cde.nlm.nih.gov).

While these registries provide a strong conceptual foundation for the definition and use of CDEs, there are a number of practical challenges that must be addressed when attempting to reuse CDEs. The common structure is provided by the ISO/IEC 11179 standard,[19] which is used as a basis of the caDSR and other repositories, is helpful but the standard does not specify implementation-level details. As a result, CDEs cannot be easily used across systems. Reflecting the strong regulatory requirements of the domains they are used in, and ISO standard conformance requirements, building new CDEs tends also to be a laborious task that involves complex workflows. In addition, CDEs must be fully specified before data collection forms can be built, adding additional complexity. Building data collection forms from CDEs can thus be an onerous task. An additional issue is that ISO/IEC 11179-based systems typically aim to provide faithful implementations of large parts of the standard, significantly increasing their complexity.

There is thus a technological barrier to developing and reusing CDEs in new systems that we believe is limiting their adoption by the broader biomedical community. To address this problem we extended an existing Web-based metadata management platform called the CEDAR Workbench[20,21] to provide a core representation of CDEs suitable for specifying questions in a metadata acquisition system. Rather than aiming to provide CDE definitions that reflect a comprehensive implementation of the ISO/IEC 11179 standard, we instead concentrated on providing the functionality in core parts of the standard that relates to the precise specification of questions and the values used to answer those questions. A key focus is on interoperation with Linked Open Data by providing a direct mapping of CDE-described data to RDF. We describe the functionality provided by the system and show how the resulting system allows users to easily use CDEs to build and deploy Web-based forms to acquire conforming data. We also show how we incorporated the large CDE library from NCI's ISO/IEC 11179-based caDSR system and made these CDEs publicly available for general use.

Methods

Although there are no widely adopted standards defining CDEs in the biomedical domain, the ISO/IEC 11179 specification[19] has been used as the underpinning of many CDE-based systems. This specification, which is formally known as the ISO/IEC 11179 Information-Technology Metadata Registry standard, is divided into six parts and covers a wide range of requirements for developing and deploying metadata registries. Part 3 of the standard describes a *Data Element*, which is the fundamental information component in the standard. An ISO data element is designed to support the description of an atomic piece of data. It has been adopted by many systems to model common data elements.

In addition to defining the core specification of a particular question and answer, the standard also outlines a rich model describing many aspects of a data element. Such information includes provenance information (e.g., who developed the data element), workflow (e.g., its development status, such as whether it is under development or is released), possible relationships with other data elements, and detailed descriptions about the context and the domain in which the data element is to be used. While useful in some situations, this information is generally not needed if the goal is to produce an operational implementation of a system that uses CDE-based question specifications. In this paper we, ignore these contextual metadata. Instead, our analysis and implementation restrict themselves to the core features needed to deploy CDEs in a metadata acquisition system.

To provide a detailed set of requirements for supporting the resulting types of CDEs, we analyzed a 11179-based CDE system. In particular, we used the cancer Data Standards Repository (caDSR)[22,23], which was developed by the U.S. National Cancer Institute. The system was designed to support the development and deployment of CDEs in data collection forms in cancer research. It adheres very closely to the ISO/IEC 11179 Edition 2 metadata standard, with extensions to support terminologies and ontologies by data elements. It provides a rich and comprehensive implementation of that standard and has been used for over two decades by clinical-trials data management systems.

Requirements

Using the caDSR system as the source, our requirements analysis aimed to identify the core set of features necessary to support CDEs in a metadata acquisition system. In addition to faithfully representing the core requirements of a CDE itself, a system must be able to ingest libraries of CDEs, be able to use them to build Web-based forms, and, finally, to be able to deploy those forms to acquire data meeting the specifications of the CDEs used in the forms. An additional requirement is to produce an RDF representation of data collected using CDEs.

The analysis identified three main sets of requirements for representing CDEs: (1) Descriptive information about a CDE must be represented. This information includes a public identifier for a CDE, its name, and a definition. CDEs

can be versioned, allowing them to evolve over time. CDEs can also be tagged with basic provenance information, which can be used to, for example, indicate their domain or to divide them into categories. (2) CDE question specifications must be represented. The caDSR system provides a number of ways to indicate the questions presented to users when they are filling in CDE-specified fields. In addition to the primary question text, CDEs may also contain an set of alternative questions. These alternatives questions can be used to customize deployed CDEs for different domains. (3) Finally, CDE value domains must be represented. An array of complex features is provided by caDSR to define the value domain of a CDE. Broadly speaking, values may be string-based, dates, times, Boolean, numeric, or from a controlled term source. String-based CDEs support fairly simple minimum and maximum length specifications. Numeric values may come from a variety of different datatypes. Units and ranges may also be specified for numeric values. Controlled term value domains specifications are the most complex. A system must allow value sets to be built to define sets of controlled values. These value sets may be customized during deployment to allow reordering of values so that the most useful values are presented to users for particular needs. Similarly, some values from a value set can be excluded for particular deployments. Again, these features are to allow customization of CDEs to meet different deployment needs.

A final set of requirements that we identified relates to the management of CDEs in a system. Users must first be able to search for CDEs for use in forms. Searching may be by CDE public identifier, name, description, category, and value in value set. Once a CDE is found, users must be able to add the selected CDE to a form, and possibly to customize the CDE. Customizations include selecting a question from a set of alternative questions and reordering or excluding values in a value set. Web-based forms must then be generated from these form specifications to acquire data from end-users. The deployed forms must enforce the value domain restrictions specified by CDEs.

Implementation

We decided to provide CDE support on top of an existing platform (Figure 1), rather than to implement the required features from scratch. This platform, which was developed by the Center for Expanded Data Annotation and Retrieval,[21] is called the CEDAR Workbench. It provides a collaborative, Web-based environment for managing metadata resources. The platform is centered on the creation of *metadata templates* (or simply *templates*) to describe biomedical experiments. These templates define the data attributes—termed *template fields* or *fields*—needed to precisely describe these experiments. For example, an experiment template may have an organism field containing the name of the organism being studied by the experiment (e.g., *Homo sapiens*).

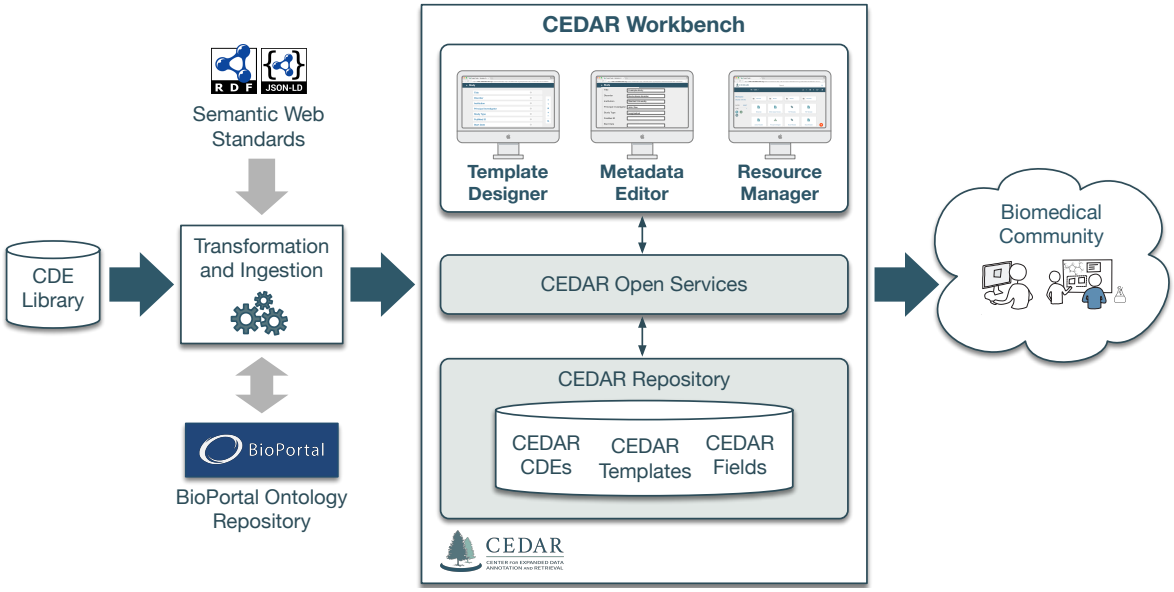

Figure 1. High-level overview of the workflow of ingesting libraries of CDEs into CEDAR. CDEs from an external library are transformed to the CEDAR model and uploaded to the CEDAR Workbench via the CEDAR REST APIs. The controlled term value sets used by these CDEs are stored in BioPortal. The biomedical community can easily access and reuse the CDEs when building Web-based metadata acquisition forms. The CDEs can also be accessed via the CEDAR REST APIs.

The template fields defined by CEDAR are analogous to CDEs in that they model an atomic piece of data. CEDAR fields effectively define a question specification and an allowed answer. In addition to basic datatypes, such as strings,

numbers, and dates, field values can also be defined using standardized terms from ontologies and other sources of controlled terms. CEDAR works in concert with the BioPortal ontology repository[1] to supply these values. BioPortal is a popular platform for accessing and sharing biomedical ontologies and hosts over 700 ontologies and 8.9 million classes. This combination of CEDAR and BioPortal provides the ability to create value set descriptions that nicely align with the requirements of CDE value domains.

A final set of features provided by CEDAR supplies the ability to deploy CDEs in Web-based forms defined by templates. These are: (1) a Template Designer, which supports interactive template creation; (2) a Metadata Editor, which allows end-users to fill in templates with metadata; and (3) a Metadata Repository, which manages the storage and retrieval of both templates and the metadata created using those templates. CEDAR defines a standardized metadata model, together with Web-based services to store, search, and share metadata.[24] This model is based on the JSON Schema and JSON-LD specifications. It allows users to publish their metadata as both JSON-LD and RDF, thus facilitating interoperation with Linked Open Data. Users can quickly create forms using the Template Designer and deploy the forms using Metadata Editor to produce semantically annotated data.

CEDAR thus provides a set of core features that are suited to supporting the deployment and use of CDEs. Based on the requirements outlined earlier, we identified the set of existing CEDAR features and the set of extensions that would meet these requirements. The primary extensions involved significantly enhancing CEDAR's field-level capabilities to support the rich value restrictions specified by CDEs. In addition to this field-level functionality, CEDAR must also be able to ingest a CDE library and map the library's CDE representation to the CEDAR model. With this functionality, CDE-based fields can be added to templates and deployed to collect metadata from users using CEDAR's Template Designer and Metadata Editor tools.

Field Extensions We performed a detailed analysis of the caDSR implementation of ISO/IEC 11179 standard to identify the core functionality that a CDE system must support. The first task was to find commonalities between the caDSR model and the CEDAR metadata model, as well as the limitations of the latter to support CDE-based fields. Note that, reflecting the ISO/IEC 11179 standard, caDSR stores rich provenance information and domain-level semantic descriptions for individual CDEs. Since we are aiming only to represent the relevant operational specifications of CDEs, we excluded these from our analysis. We identified four classes in the caDSR model that contain information that represents this operational information. The classes are *Data Element*, *Value Domain*, *Permissible Values*, and *Reference Document* (Figure 2).

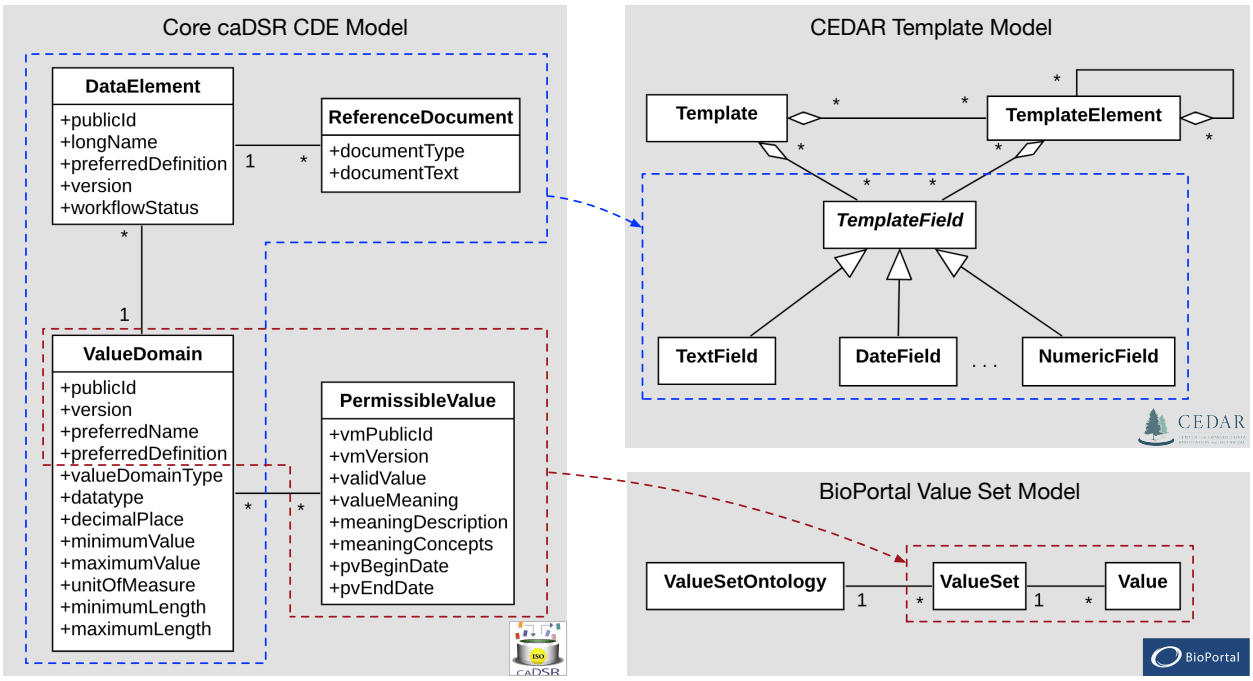

Figure 2. Visual representation of the mappings between the core caDSR CDE model and the CEDAR and BioPortal models. The figure shows the subset of attributes of the caDSR model that are mapped to the CEDAR Template Model (dashed blue line) and to the BioPortal Value Set model (dashed red line). The attributes *publicId*, *version*, *preferredName*, and *preferredDefinition* from the *ValueDomain* caDSR entity are mapped both to CEDAR and BioPortal.

The core CDE entity in caDSR model is the *Data Element*, which represents the smallest unit of data that can be represented and exchanged between systems. It contains attributes that capture descriptive information for a CDE, such as its public identifier, name, and version. The characteristics of the values accepted by a data element are defined by the *ValueDomain* entity, which contains attributes to specify the data type and the different value constraints. Each data element and value domain can contain terminology references. The attribute *ValueDomainType* of the *ValueDomain* entity specifies whether the domain is enumerated—when it is specified by a predetermined list of permissible values (e.g., *male* and *female* for the element *Sex*)—or non-enumerated—when the domain is specified by a description or range (e.g., positive integers for the element *Number of Months Stayed Off Cigarettes*). When the value domain is enumerated, the accepted values are represented by the *PermissibleValue* entity, which specifies the exact names, codes, and textual labels that can be stored for the CDE.

Table 1. Field-level mappings between the core caDSR model and the CEDAR and BioPortal models. The table shows the core attributes of a CDE in the caDSR model and the corresponding attributes in a CEDAR model. For example, the first row shows that the attribute *PUBLICID* from the element *Data Element* in the NCIs caDSR model file is mapped to the field *schema:identifier* in the CEDAR model. The right column shows the attributes used to represent value sets and their values in BioPortal. Fields added to the CEDAR model to support the representation of CDEs are indicated by an asterisk.

NCI caDSR CDE (XML)		**CEDAR field (JSON Schema, JSON-LD)**	**BioPortal caDSR Value Sets (JSON)**
DataElement	PUBLICID	schema:identifier*	-
	LONGNAME	schema:name title description	-
	PREFERREDDEFINITION	schema:description	-
	VERSION	pav:version	-
	WORKFLOWSTATUS	bibo:status	-
ReferenceDocument	DocumentType	Used to map DocumentText	-
	DocumentText	skos:prefLabel skos:altLabel*	-
ValueDomain	PublicId, Version	_valueConstraints.valueSets.uri _valueConstraints.actions.sourceUri*	ID, prefLabel
	PublicId	-	identifier
	Version	-	hasVersion
	PreferredName	_valueConstraints.valueSets.name	altLabel
	PreferredDefinition	-	comment
	ValueDomainType Datatype	_ui.inputType _valueConstraints.numberType*	-
	DecimalPlace	_valueConstraints.decimalPlace*	-
	MinimumValue	_valueConstraints.minValue*	-
	MaximumValue	_valueConstraints.maxValue*	-
	UnitOfMeasure	_valueConstraints.unitOfMeasure*	-
	MinimumLength	_valueConstraints.minLength*	--
	MaximumLength	_valueConstraints.maxLength*	-
PermissibleValue	VMPUBLICID, VMVERSION	_valueConstraints.actions.termUri*	ID, prefLabel
	VMPUBLICID	-	identifier
	VMVERSION	-	hasVersion
	VALIDVALUE	skos:notation*	notation
	VALUEMEANING	rdfs:label	prefLabel
	MEANINGDESCRIPTION	-	comment
	MEANINGCONCEPTS	@id	relatedMatch
	PVBEGINDATE	-	startTime
	PVENDDATE	-	endTime

When comparing the caDSR and CEDAR models, we noticed a direct correspondence between the caDSR *DataElement*, *ReferenceDocument*, and *ValueDomain* entities and the CEDAR *TemplateField* entity. We observed that the *TemplateField* entity contained attributes to cover some of the core information for CDEs, such as name, definition, and version. However, it lacked support for some other crucial CDE features. We identified five main limitations in the CEDAR model and associated software and developed functionality to provide them:

Public Identifiers The CEDAR model lacked a way to store a public identifier of a field. This information complements the field CEDAR identifier and it is crucial to accurately find and refer to the CDE. As a solution to this limitation, we added the `schema:identifier` property to the model, which can now be used to store public identifiers for any CEDAR field. The Template Designer and Metadata Editor were also extended to handle this new field.

Question Text The caDSR *ReferenceDocument* entity makes it possible to specify, in addition to the preferred question text, alternate questions that can be presented to the user when filling out CDE-specified fields. An example of preferred question text for a CDE is *Has the disease relapsed?* while an alternate question is *Was the status considered a disease relapse?* We extended CEDAR to allow entering an alternate question text and stored it in the model using the property `skos:altLabel`. The Template Designer and Metadata Editor were also extended to work with alternate questions.

Datatypes The caDSR *ValueDomain* entity contains a *dataType* attribute that specifies the type of the value accepted by the CDE. Our analysis of a set of 60,409 caDSR CDEs revealed 190 different data types. Most of these datatypes can be directly mapped to equivalent data types supported by CEDAR. To limit the implementation effort, we decided to initially map the 10 most used datatypes (CHARACTER, java.lang.String, ALPHANUMERIC, ISO21090CDv1.0, NUMBER, java.lang.Long, java.lang.Integer, java.lang.Double, java.util.Date, DATE). These 10 types are present in 53,175 CDEs (88%). We are currently working on mappings the remaining data types to existing CEDAR types. In a few cases, we will need to extend CEDAR to support less common data types.

Value Constraints The *ValueDomain* entity contains a rich set of attributes to define some advanced value constraints, which the CEDAR model did not support. We extended the CEDAR model with fields to specify the minimum and maximum values, the number of decimal places, and the unit of measure accepted by numeric fields. We also the added fields to store the minimum and maximum length of string fields. The Template Designer and Metadata Editors were also enhanced to handle these value constraint extensions.

Value Sets In caDSR value sets can be used to define the range of possible values for a CDE. Value sets are versioned, first-class entities. They can be reused by several CDEs and can evolve over time. Typically, the values in a value set are selected from controlled term sources. Most value set values in caDSR come from the National Cancer Institute Thesaurus (NCIT). In collaboration with the BioPortal team, we extended BioPortal to support the representation of caDSR value sets. The enumerated value set specified by a *ValueDomain* is mapped to a BioPortal value set, while all the permissible values are stored as values in the value set. In CEDAR, the CDEs with enumerated values are linked to BioPortal using a URI that identifies the value set in BioPortal. We also extended the CEDAR model to support value set reordering and value exclusions. The Metadata Editor was also modified to handle this reordering and exclusion functionality.

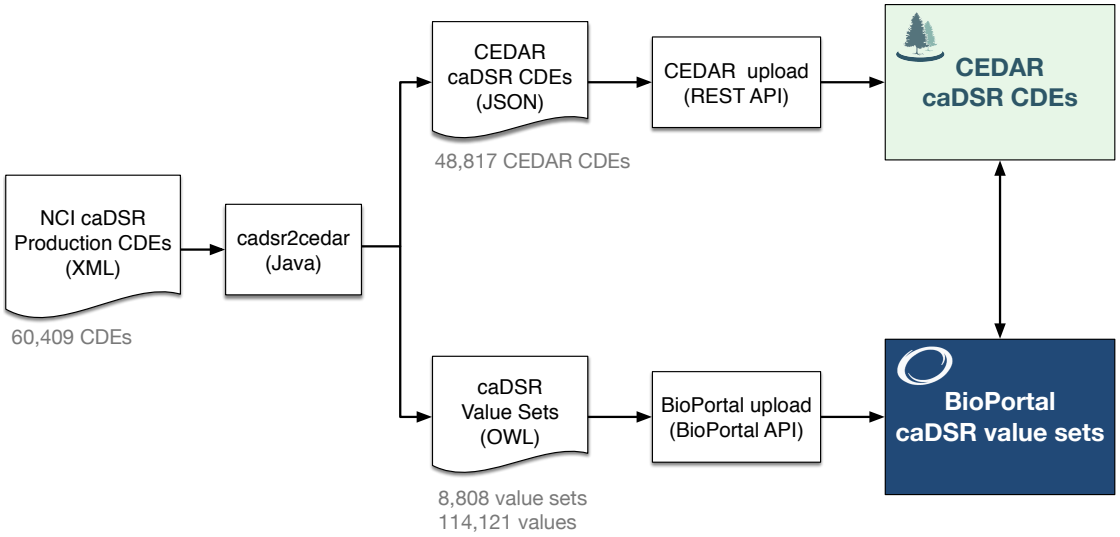

Figure 3. Schematic showing the caDSR CDE ingestion workflow. The *cadsr2cedar* tool takes an XML file with a set of caDSR CDEs and transforms each CDE to a CEDAR field. The transformed CDE fields are uploaded to the CEDAR Workbench via the CEDAR REST API. Any associated value sets and their values are transformed into an OWL and then uploaded to BioPortal via the BioPortal REST API.

Ingestion Pipeline We developed a pipeline to ingest the set of public caDSR CDEs into the CEDAR Workbench. The process converts XML-encoded caDSR CDEs to JSON Schema-encoded fields in the CEDAR model (Figure 3). We used a set of 60,409 CDEs that we downloaded from caDSR[23] on August 2018. We developed a tool called *cadsr2cedar* to transform the CDEs in the XML file to the CEDAR model. Table 1 shows the primary mappings used

in this conversion. The conversion process excluded currently unsupported datatypes and also excluded CDEs that were not marked as released by the caDSR system. As a result, we obtained 48,817 CDEs, which we uploaded to the CEDAR Workbench using its REST API. The *cadsr2cedar* tool also generated a total of 8,808 different value sets and 114,121 values used by the CDEs. We uploaded these to BioPortal using the BioPortal REST API.

Currently, we run the conversion process manually, but we plan to enhance it to run to immediately reflect CDE updates in the caDSR system. Extensions will be needed to the caDSR system to notify 3rd-party systems of such updates. The ingestion tools will also need to quickly identify the CDEs or value sets that have changed since the last update and selectively update only the corresponding CEDAR or BioPortal entities.

All the CDEs ingested are publicly available in the CEDAR Workbench (https://goo.gl/SggjQB).

Results

The main result of this work is a new version of the CEDAR Workbench that supports CDEs and that has been equipped with a large library of CDEs defined and maintained by the NCI. In addition to the field types natively available in the CEDAR Workbench (e.g., text, date, numeric, ontology values), CEDAR users can now search and select from a large number of CDEs ingested from the NCI's caDSR registry to build Web-based data acquisition forms. These forms can be used to collect data based on standard values from ontologies and terminologies, and easily share both the forms and the collected data with the broader biomedical community.

We used a CDE ingestion pipeline to incorporate 49,280 NCI caDSR CDEs into the CEDAR Workbench, as well as to upload the corresponding value sets to the BioPortal ontology repository. Our CDE ingestion pipeline can be reused to ingest new NCI caDSR CDEs into CEDAR. It can also be adapted to include CDEs from other sources.

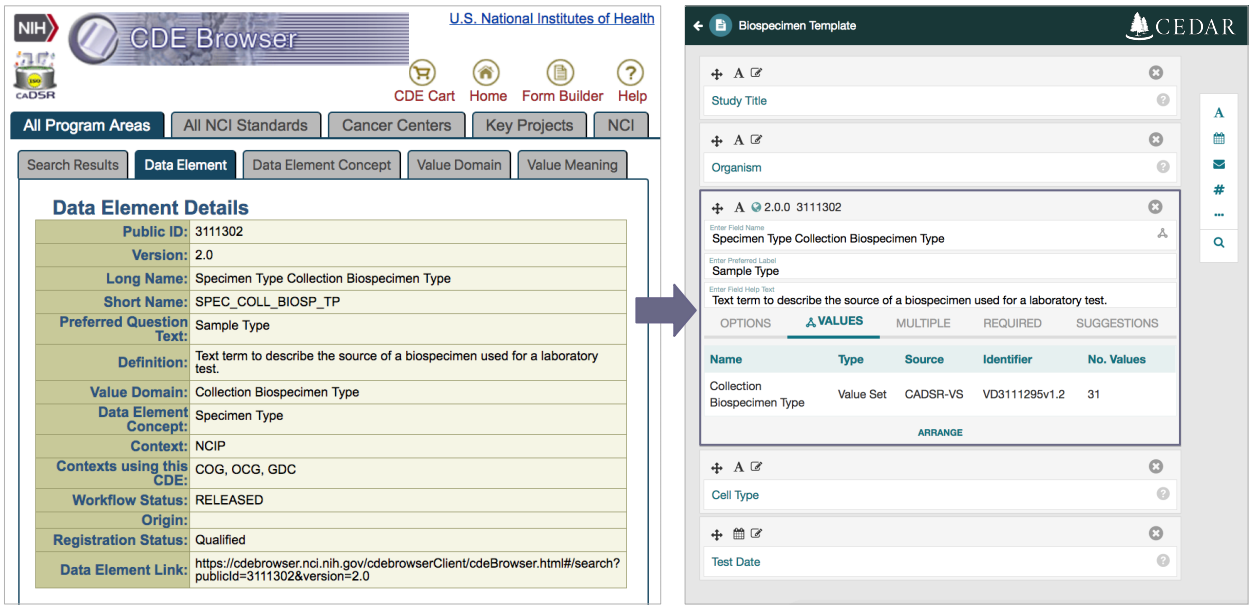

Figure 4. Parallel representation of a CDE in the caDSR and CEDAR systems. The left side of the figure shows a screenshot of NCI's CDE Browser with details of a CDE, including its public identifier (*3111302*), version (*2.0*), and long name (*Specimen Type Collection Biospecimen Type*). The right side of the figure shows how that particular CDE can be used in the CEDAR Template Designer to build a Biospecimen template in combination with other non-CDE fields (*Study Title*, *Organism*, *Cell Type*, and *Test Date*), which were created on-the-fly. The Template Designer displays the most relevant information for the CDE, including its version, public identifier, long name, preferred question text, definition, and value set. This value set can be explored interactively.

CDEs can now be used when building CEDAR templates (Figure 4). The left side of the figure shows a screenshot of the NCI's CDE Browser (https://cdebrowser.nci.nih.gov/cdebrowserClient/cdeBrowser.html) for the CDE *Specimen Type Collection Biospecimen Type*. The CDE Browser provides relevant details for the selected CDE, including its identifier (*3111302*), version (*2.0*), and the preferred question text that should be used when incorporating it into a form (*Sample Type*). The right side of the figure shows how that CDE can be used in the CEDAR Template Designer to build a *Biospecimen* template, in combination with either other CDEs or with other fields that can be created on-the0fly, such as *Study Title* and *Organism*.

Figure 5 shows a screenshot of the Metadata Editor for the *Biospecimen* template, with the list of allowed values for the *Sample Type*. The right side of the figure shows the entered values in JSON-LD format. For the *Sample Type* field, CEDAR's metadata contain not only the label of the value selected (*Blood*) but also the URI of the corresponding term in the NCI Thesaurus (http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C12434).

Our current implementation allowed us to transform 88% of the set of 60,409 caDSR CDEs available in August 2018. We are currently working on extending CEDAR to reach 100% coverage.

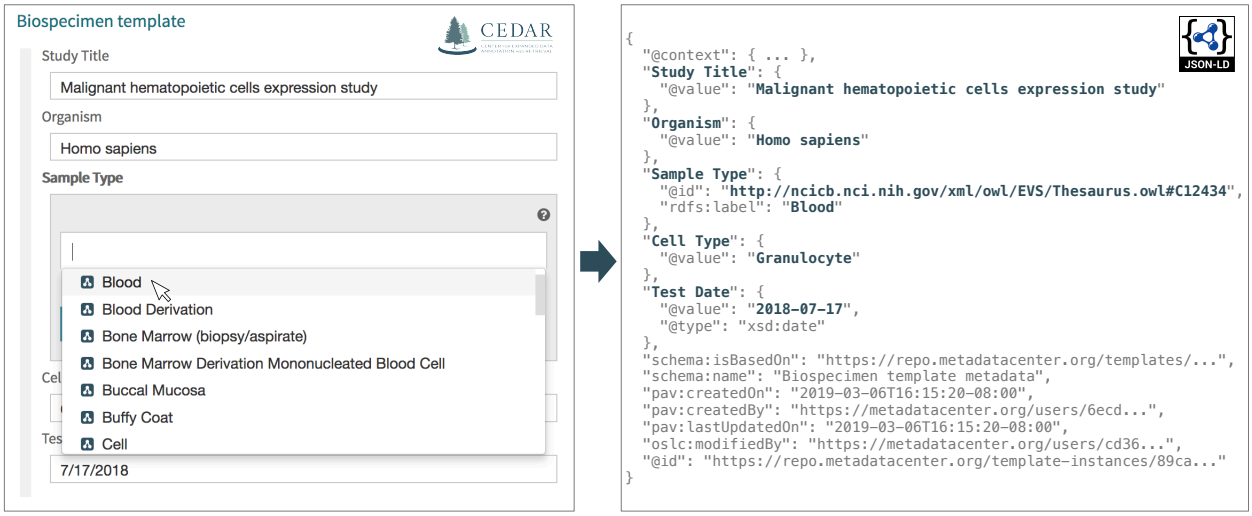

Figure 5. Screenshot of CEDAR's Metadata Editor displaying a data acquisition form generated from the *Biospecimen* template shown in Figure 4. Here, the user is about to select the value *Blood* from the list of allowed values for the *Sample Type* field. The right side of the figure shows CEDAR's JSON-LD representation for the entered values. For the *Sample Type* field, the JSON-LD representation contains not only the label of the selected value (*Blood*), but also the URI of that term in the NCI Thesaurus (*http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C12434*). CEDAR's JSON-LD metadata also contains the values entered for the other fields, as well as provenance information such as the author and the date when the values were entered. This JSON-LD representation can be automatically transformed to RDF.

Discussion

Over the past few decades, there has been a strong emphasis on standardizing data collection and reporting requirements to facilitate data discovery, data interpretation, and data reuse. Different communities have adopted a variety of approaches to address these standardization needs. The cancer community, for example, has adopted CDEs as a mechanism for defining reusable question specifications that can be employed when collecting and reporting data in clinical research studies. Large libraries of CDEs have been built, which provide a strong foundation for interoperation. The social sciences community often uses data collection tools such as REDCap[25] for their reporting needs. REDCap allows users to construct data acquisition forms. These forms can be built from reusable data collection instruments, which are used to standardize data-collection requirements over different studies. In the biological sciences, controlled terminologies are widely used. Many comprehensive ontologies have been developed, providing a common vocabulary to refer to biological entities.

While these solutions address the needs of their respective communities, they do not interoperate with each other, and each has shortcomings. For example, approaches such a REDCap make little use of controlled terms. The CDE-based solutions developed by the cancer community typically involve systems that are not easily reusable outside the specialized task of constructing case report forms. And, while controlled terms are useful in themselves, they do not address the full needs of specifying data-collection and reporting requirements. There is a need for an interoperable approach that meets the needs of these different communities.

We believe that the CDE-based approach outlined in this paper can provide such a solution. The system we developed supports the easy form construction capabilities of tools such as REDCap and the precise data specification advantages of CDEs, together with the central use of controlled terms. The system is built using the principled definition of CDEs provided by the ISO/IEC 11179 standard. However, the goal was not to exhaustively represent the wide array of provenance information and conceptual metadata that is specified by the standard; such information would be needed,

for example, when developing a specification-conforming metadata registry, or when trying to identify or analyze CDEs or data that are semantically related. Instead, the system restricts itself to the core parts of the standard necessary to represent standalone question specifications. This restricted interpretation of the ISO/IEC 11179 standard leverages its power while simplifying the specification of robust CDEs. We have done some preliminary testing to demonstrate the feasibility of our approach with NCI users, though more empirical testing is required to fully evaluate our claim.

Conclusion

In this paper, we describe how we extended the CEDAR Workbench to natively support CDE-based question specifications. The primary goal of this work is to provide an open platform that dramatically simplifies the use and deployment of CDEs by supplying intuitive and highly interactive Web-based interfaces. A key focus is to support interoperation, both by allowing third-party CDEs to be incorporated into the system, and by representing CDEs using ontologies and Semantic Web standards. Support for this markup was provided by extending the BioPortal ontology repository to natively facilitate the creation of value sets of controlled terms and then allowing these values sets to be interactively linked to CDE definitions. The resulting functionality supports the creation of robust, semantically rich CDE definitions that can be quickly deployed to collect data.

We ingested a library of over 48,000 CDEs and associated value sets from the NCI's caDSR CDE repository and made these CDEs available for public use. In particular, we validated that the system could present questions to users that accurately reflect a CDE specification and that it could also ensure that acquired answers fully meet the value requirements of those specifications. As outlined in this paper, we made several extensions to CEDAR to support caDSR CDEs, eventually reaching 88% coverage. We are currently developing additional features to fully represent the remaining caDSR CDEs. In addition to making the CDEs available for reuse, the goal of this ingestion task was to validate that CEDAR could faithfully represent and enforce a subset of the key elements of an ISO/IEC 11179-based CDE specification. This initial set of CDEs was developed over several decades for use in a large number of case report forms for clinical trials. They cover a broad range of cancer research. We plan to ingest CDEs from a variety of other sources to increase domain coverage. We also plan to develop additional features to support the management and search of multiple, large CDE collections.

The resulting system provides an open platform for sharing, managing, and deploying CDEs. We believe that the system lowers the barrier to the use of CDEs by leveraging existing CDE libraries and by supporting the easy creation of data-collection forms that allow these CDEs to be quickly provided to end-users.

Acknowledgements

CEDAR is supported by the National Institutes of Health through an NIH Big Data to Knowledge program under grant 1U54AI117925 and partially sponsored by NCI/FNLCR IDIQ Agreement 17X074. NCBO is supported by the NIH Common Fund under grant U54HG004028. The CEDAR Workbench is available at https://cedar.metadatacenter.org and on GitHub at https://github.com/metadatacenter.

References

1. Noy NF, Shah NH, Whetzel PL, et al. BioPortal: ontologies and integrated data resources at the click of a mouse. *Nucleic Acids Res*. 2009;37(Suppl 2):W170-W173. doi:10.1093/nar/gkp440.
2. Richesson RL, Chute CG. Health information technology data standards get down to business: Maturation within domains and the emergence of interoperability. *J Am Med Informatics Assoc*. 2015;22(3):492-494. doi:10.1093/jamia/ocv039.
3. Sheehan J, Hirschfeld S, Foster E, et al. Improving the value of clinical research through the use of Common Data Elements. *Clin Trials*. 2016;13(6):671-676. doi:10.1177/1740774516653238.
4. Huser V, Amos L. Analyzing Real-World Use of Research Common Data Elements. In: *Proceedings of AMIA 2018 Annual Symposium*. Vol ; 2018:602-608. doi:10.13140/RG.2.2.10964.91529.
5. Silva J, Wittes R. Role of clinical trials informatics in the NCI's cancer informatics infrastructure. In: *Proceedings of AMIA 1999 Annual Symposium*. Vol ; 1999:950-954.
6. Jiang G, Solbrig HR, Chute CG. Quality evaluation of value sets from cancer study common data elements using the UMLS semantic groups. *J Am Med Informatics Assoc*. 2012;19(E1):129-136. doi:10.1136/amiajnl-2011-000739.
7. Williams R, Kontopantelis E, Buchan I, Peek N. Clinical code set engineering for reusing EHR data for research: A review. *J Biomed Inform*. 2017;70:1-13. doi:10.1016/j.jbi.2017.04.010.
8. Bodenreider O, Nguyen D, Chiang P, et al. The NLM value set authority center. *Stud Health Technol Inform*. 2013;192(1-2):1224. doi:10.3233/978-1-61499-289-9-1224.

9. Warzel DB, Andonaydis C, McCurry B, Chilukuri R, Ishmukhamedov S, Covitz P. Common data element (CDE) management and deployment in clinical trials. *AMIA Annu Symp Proc*. 2003:1048. http://www.ncbi.nlm.nih.gov/pmc/articles/PMC1480162/.
10. Patel AA, Kajdacsy-Balla A, Berman JJ, et al. The development of common data elements for a multi-institute prostate cancer tissue bank: The Cooperative Prostate Cancer Tissue Resource (CPCTR) experience. *BMC Cancer*. 2005;5:1-14. doi:10.1186/1471-2407-5-108.
11. Nadkarni PM, Brandt CA. The common data elements for cancer research: Remarks on functions and structure. *Methods Inf Med*. 2006;45(6):594-601.
12. Winget MD, Baron JA, Spitz MR, et al. Development of common data elements: the experience of and recommendations from the early detection research network. *Int J Med Inform*. 2003;70(1):41-48. doi:10.1016/s1386-5056(03)00005-4.
13. Loring DW, Lowenstein DH, Barbaro NM, et al. NIH Public Access. 2013;52(6):1186-1191. doi:10.1111/j.1528-1167.2011.03018.x.Common.
14. Miller AC, Odenkirchen J, Duhaime A-C, Hicks R. Common Data Elements for Research on Traumatic Brain Injury: Pediatric Considerations. *J Neurotrauma*. 2011;29(4):634-638. doi:10.1089/neu.2011.1932.
15. Saver JL, Warach S, Janis S, et al. Standardizing the structure of stroke clinical and epidemiologic research data: The national institute of neurological disorders and stroke (NINDS) stroke common data element (CDE) project. *Stroke*. 2012;43(4):967-973. doi:10.1161/STROKEAHA.111.634352.
16. Xu J, Pathak J, Mo H, et al. Developing a data element repository to support EHR-driven phenotype algorithm authoring and execution. *J Biomed Inform*. 2016;62:232-242. doi:10.1016/j.jbi.2016.07.008.
17. Rubin DL, Kahn CE. Common Data Elements in Radiology. *Radiology*. 2016;283(3):837-844. doi:10.1148/radiol.2016161553.
18. Meadows B, Abrams J, Christian M, et al. The Common Data Element Dictionary - Developing a standard nomenclature for reporting cancer clinical trial data. *Proceedings of the 14th IEEE Symposium on Computer-Based Medical Systems (CBMS 2001)*. 2001:498-502.
19. ISO/IEC 11179 metadata standard. http://www.metadata-standards.org/11179/. Accessed March 10, 2019.
20. Gonçalves RS, O'Connor MJ, Martínez-Romero M, et al. The CEDAR Workbench: An ontology-assisted environment for authoring metadata that describe scientific experiments. In: *Proceedings of the 16th International Semantic Web Conference (ISWC 2017)*. Vol 10588 LNCS. ; 2017:103-110. doi:10.1007/978-3-319-68204-4_10.
21. Musen MA, Bean CA, Cheung KH, et al. The Center for Expanded Data Annotation and Retrieval. *J Am Med Informatics Assoc*. 2015;22(6):1148-1152. doi:10.1093/jamia/ocv048.
22. Nadkarni PM, Brandt CA. The common data elements for cancer research: Remarks on functions and structure. *Methods Inf Med*. 2006.
23. NCI. caDSR Wiki. https://wiki.nci.nih.gov/display/caDSR/caDSR+Wiki. Accessed March 10, 2019.
24. O'Connor MJ, Martinez-Romero M, Egyedi AL, Willrett D, Graybeal J, Musen MA. An open repository model for acquiring knowledge about scientific experiments. In: *Proceedings of the 20th International Conference on Knowledge Engineering and Knowledge Management (EKAW 2016)*. Vol ; 2016:762-777. doi:10.1007/978-3-319-49004-5_49.
25. Harris PA, Taylor R, Thielke R, Payne J, Gonzalez N, Conde JG. Research electronic data capture (REDCap)-A metadata-driven methodology and workflow process for providing translational research informatics support. *J Biomed Inform*. 2009;42(2):377-381. doi:10.1016/j.jbi.2008.08.010.