JCIM JOURNAL OF CHEMICAL INFORMATION AND MODELING

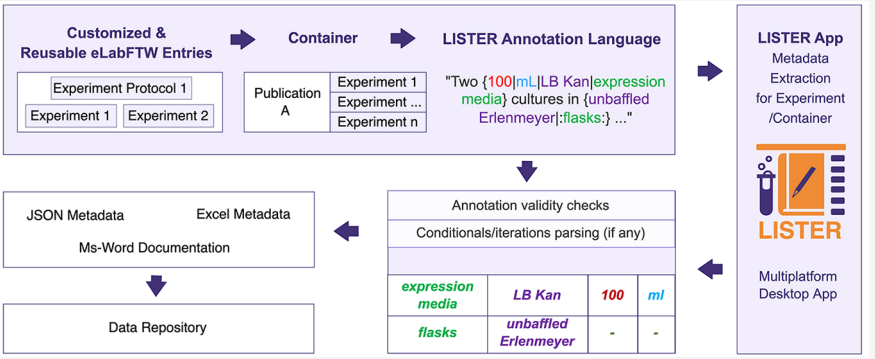
pubs.acs.org/jcim Article

LISTER: Semiautomatic Metadata Extraction from Annotated Experiment Documentation in eLabFTW

Fathoni A. Musyaffa, Kirsten Rapp, and Holger Gohlke*

Cite This: J. Chem. Inf. Model. 2023, 63, 6224−6238 Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information



ABSTRACT: The availability of scientific methods, code, and data is key for reproducing an experiment. Research data should be made available following the FAIR principle (**f**indable, **a**ccessible, **i**nteroperable, and **r**eusable). For that, the annotation of research data with metadata is central. However, existing research data management workflows often require that metadata be created by the corresponding researchers, which takes effort and time. Here, we developed LISTER as a methodological and algorithmic solution to create and extract metadata from annotated, template-based experimental documentation using minimum effort. We focused on tailoring the integration between existing platforms by using eLabFTW as the electronic lab notebook and adopting the ISA (**i**nvestigation, **s**tudy, **a**ssay) model as the abstract data model framework. LISTER consists of four components: annotation language to support metadata extraction; customized eLabFTW entries using specific hierarchies, templates, and tags to structure reusable scientific documentation; a "container" concept in eLabFTW, making metadata of a particular container content extractable along with its underlying, related experiments via a single click; a Python-based app to enable easy-to-use, semiautomated metadata extraction from eLabFTW entries. LISTER outputs metadata in machine-readable .json and human-readable .xlsx formats, and Material and Methods (MM) descriptions in .docx format that could be used in a thesis or manuscript. The metadata can be used as a basis to create or extend ontologies, which, when applied to the published research data, will significantly enhance its value. DSpace is used as a data cataloging platform for hosting the extracted metadata and research data. We applied LISTER to computational biophysical chemistry, protein biochemistry, and molecular biology, and our concept should be extendable to other life science areas.

1. BACKGROUND

In a survey conducted in 2016 involving more than 1500 researchers, most respondents agreed that a reproducibility crisis exists in various scientific domains, including chemistry, biology, physics, engineering, and medicine.[1] Besides inherent factors related to the situation in academia, such as selective reporting, pressure to publish, poor analysis, and insufficient mentoring, extrinsic factors, such as the unavailability of methods, code, and primary data (i.e., data needed for reproducing an experiment), contribute to this crisis. The data unavailability led funding agencies and science publishers to require researchers to publish their data. For example, the German Research Foundation (DFG) expects research data to be made available for at least ten years.[2]

Conscious efforts to build a better data infrastructure and standards have received wide attention. For example, the National Research Data Infrastructure (NFDI) was established in Germany as an association comprised of many German institutions whose purpose is to create a permanent digital knowledge repository.[3] In the United States, the National Science Foundation (NSF) requires research data obtained under the NSF grants to be shared and to adhere to guidelines specifically designated for each NSF directorate field.[4] As a publisher, the American Chemical Society (ACS) also provides a guideline regarding the data policy: each journal under the

Received: May 16, 2023
Published: September 29, 2023

ACS umbrella follows one of four policy levels concerning data availability.[5]

Research data needs to be made available following the FAIR data principle, which provides guidelines as to four main quality aspects: research data should be **f**indable, **a**ccessible, **i**nteroperable, and **r**eusable.[6] Further specifications have been recommended for other dedicated research fields. In machine learning, Heil et al. categorize three research data *reproducibility* standards based on which data quality factors are satisfied.[7] Besides reproducibility, *repeatability* and *replicability* are being discussed as aspects of research data, although there is no universally accepted definition for the terms.[8] As one example, the Association for Computing Machinery[9] (ACM) differentiates between these three terms such that 1) *repeatability* refers to achieving the same precision through an identical experimental setup performed by the same team, 2) *replicability* is achieving the same precision with the same experimental setup but executed by a different team, and 3) *reproducibility* is the ability to achieve the stated precision by a different team using a distinct experimental setup. This definition, however, is not always aligned with other publications, for example, Prasad et al.[10] and Bollen et al.[11] emphasize the recollection of new data as the requirement of a replicable study.

In addition, these interpretations are not always straightforward. For instance, in computational science, the question arises whether the use of the same software but a divergent hardware setup (such as different CPU or GPU configurations and specifications) is permissible. Concerning replicability vs reproducibility, Goodman et al.[12] suggested two new terms: "*Methods reproducibility*" and "*Results reproducibility*", which align with the term *replicability* and *reproducibility* from ACM.[8]

In the long run, we aim for the reproducibility of research data. This aim is heavily influenced by the extent of provided metadata and documentation, requiring standardization of which organized terms are necessary to make experiments reproducible from each research domain. At this point, we have not established a standard set of metadata that would classify an experiment as either replicable or reproducible; this remains beyond the scope of our current work. The LISTER (**L**ife **S**cience Experimen**t**s Metadata Pars**er**) workflow proposed here allows the gathering of the relevant metadata fields, which will later provide the minimum set of metadata fields for a reproducible experiment.

Besides the question of reproducibility, one can envision several use cases that can benefit from good research data management (RDM), ranging from influences on the local research environment to global scales:

(I) *Data preservation, preparation, and collection* can be streamlined for researchers before tasks such as thesis, article, or grant proposal preparation, provided that regular Research Data Management (RDM) practices are employed.

(II) *Access, discovery, and filtering* of data produced by fellow lab members or ex-lab members, which could be useful for example during the extension of the research series.

(III) *Data cataloging* that ascertains contextual information regarding the research activities, with little effort both from the involved researchers and for developing/setting up the digital infrastructure.

(IV) *Data alignment* with other data repositories and catalogs, by associating highly structured metadata from one data repository to another repository/data catalog using different levels of granularity and technology stacks.

(V) *Persistent storage* of data for access even a long time after the research has been conducted.

We reasoned that several requirements need to be satisfied to support such use cases:

1 Research data should be preserved, as well as findable, accessible, and downloadable openly by anyone that needs the data when the license is not restrictive, which is mostly the case in scientific publications. (Use case I–II.)

2 The metadata of these research data should be extracted using minimum efforts to allow researchers to focus on the experiment design, execution, and analysis instead of manually extracting the metadata. (Use case III.)

3 Whenever possible, an existing and proven technology stack or standard should be adopted, which allows faster implementation and wider acceptance compared to building digital infrastructures from scratch. (Use case IV.)

4 Both research data and metadata should be stored on a long-term basis with regular maintenance and backup operations. The data repository should allow for scaling to increasing amounts of research data, and the metadata should be searchable. (Use case V.)

The annotation of research data with metadata is central to any RDM workflow. Some specifications and frameworks exist to standardize metadata both structurally and terminologically. As one, the ISA Model provides a community-driven metadata-tracking framework for providing rich descriptions of heterogeneous experiments in the life science, biomedical and environmental domains.[13] The model is based on three fundamental concepts surrounding *Investigation* to describe the project context, *Study* to explain the unit of research, and *Assay* to provide analytical measurements.[13] Each fundamental concept contains different metadata. A more detailed view of the ISA model is discussed in the Overview section.

Good RDM practices include keeping sufficient documentation, organizing and naming files consistently, versioning the files, creating a security plan when applicable, defining roles and responsibilities for user and access management, backing up the data, identifying tool constraints, archiving the research data upon project closing, putting the data into repositories, and documenting the conventions in a data management plan.[14] This can be facilitated by the use of electronic Lab Notebooks (ELNs) instead of traditional paper-based notebooks. ELNs offer additional features, such as facilitating auditing and enabling collaborations.[15] They are particularly well-suited for modern life science and engineering experiments that generate extensive data sets and demand high data integrity.[16] ELNs benefit researchers by facilitating long-term storage, reproducibility, and easy access to experiment records across multiple devices, as well as ensuring compliance with standard operating procedures, protecting intellectual property, fostering collaboration, and supporting open science initiatives.[17]

For our implementation within the LISTER framework, we utilize eLabFTW[18] as the preferred ELN. eLabFTW is a web-based, open-source electronic lab notebook software that can be installed on a server and features common ELN functionalities such as experiment tracking, lab asset management, timestamping (as legal proof in the case of patent

https://doi.org/10.1021/acs.jcim.3c00744
J. Chem. Inf. Model. 2023, 63, 6224−6238

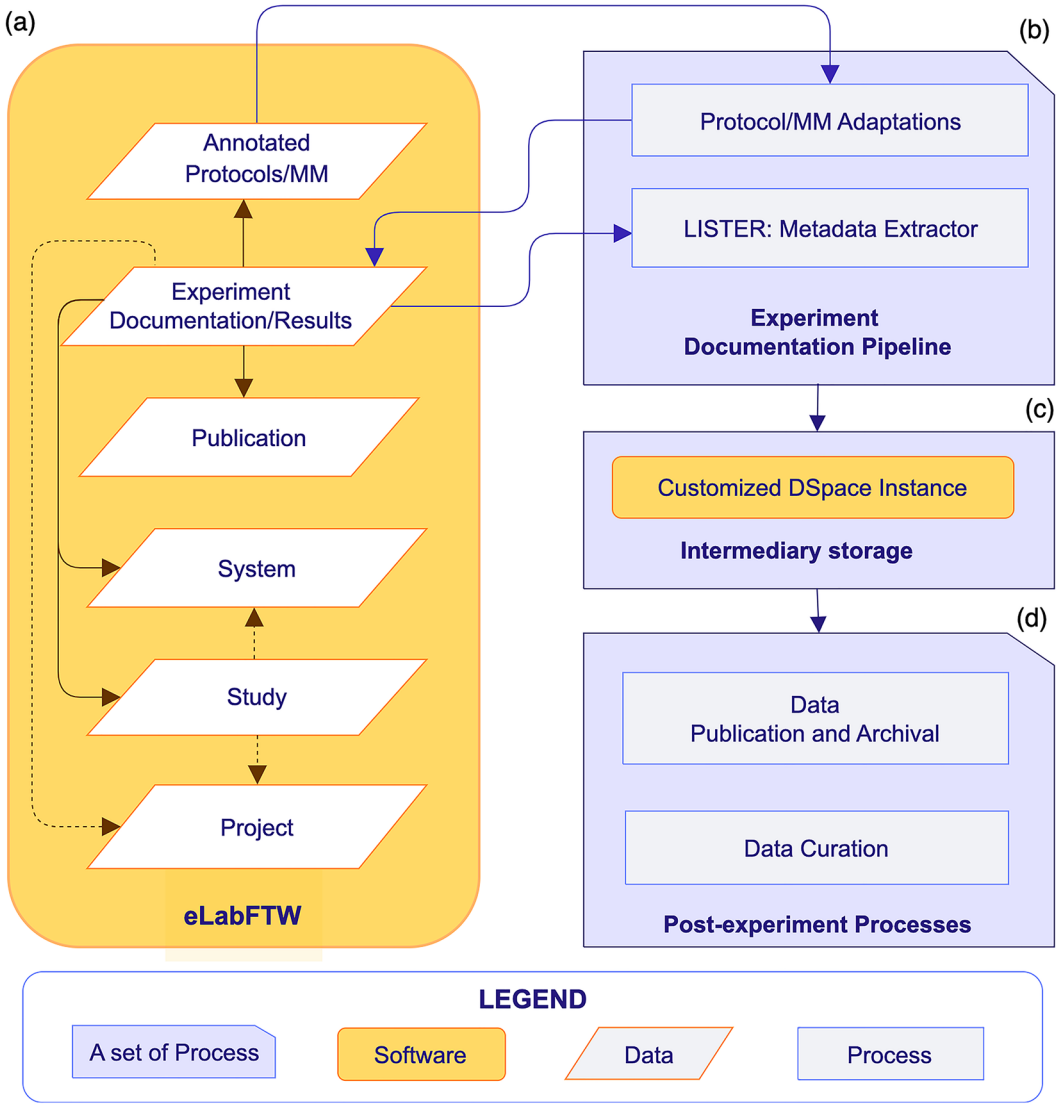


Figure 1. Interplay of the components of the LISTER-based RDM workflow. (a) eLabFTW contains a hierarchy of specific Resource classes as containers to map an (extended) ISA model; stores annotated protocol and MM templates; and holds adapted annotated experiment documentation (see below) and results. (b) The user adapts annotated Protocol/MM entries and invokes metadata extraction with LISTER (see below). (c) Extracted metadata and corresponding research data are stored in a data repository platform, such as a customized DSpace instance as intermediary storage. (d) From the intermediary storage, metadata and research data can be prepared for archival (such as in Zenodo and bioRxiv) and curation (such as re3data) or retrieved for publication purposes (such as publisher websites).

disputes), lab equipment scheduler, experiment documentation, and the possibility to create custom-type entries in its *Resource*[19] (in previous versions of eLabFTW termed *Database*). eLabFTW is widely used in the scientific community (in GitHub, it has >700 stars and ∼1800 issues have been submitted, with ∼1700 issues resolved/closed),[20] and freely available for download and use and has been translated into 17 languages. eLabFTW also facilitates metadata storage that can be attached to each experiment, allowing custom JSON data in experiment entries through the metadata attribute. This enables adding JSON content or utilizing specific keys to enhance the customization of the entry such as incorporating additional fields. In addition, eLabFTW allows for fetching its contents via an API, which is a necessity for the LISTER workflow.

2. APPROACH

2.1. Overview. We provide methodological and algorithmic solutions coined LISTER to satisfy the requirements mentioned in the *BACKGROUND* section, focusing on the field of life sciences, which includes molecular modeling/simulations and wet lab experiments as pilot cases. The LISTER workflow extracts metadata from experiment documentation with minimum effort from the researcher and makes the research data findable, accessible, and downloadable. Our developments are guided by the necessities of users whose scientific focus should not get distracted by manually annotating experiments from scratch to conform with RDM and, in particular, metadata creation and extraction. This aspect makes LISTER unique compared to alternative approaches (see below in the *Related Works* section).[21]

We used existing RDM standards and platforms whenever possible. That way, we can focus on tailoring integration

https://doi.org/10.1021/acs.jcim.3c00744
J. Chem. Inf. Model. 2023, 63, 6224−6238

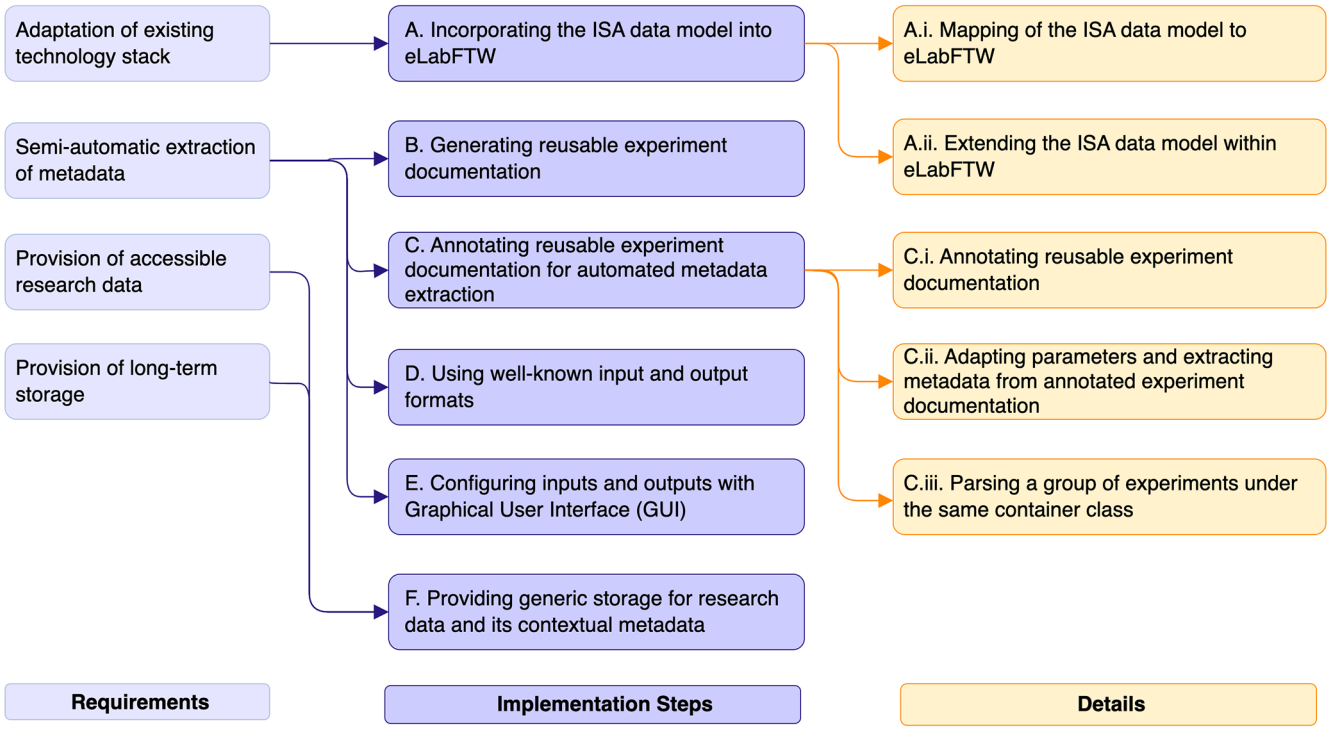


Figure 2. Requirements (left) and developed implementation steps (middle). The goal is to make research data accessible and downloadable as well as to extract metadata using minimal effort. Detailed steps are given on the right.

between existing platforms and developing necessary applications as a critical contribution to implementing good RDM practices. We use eLabFTW as the ELN and adopt the ISA model as the abstract data model framework, which is implicitly supported by our eLabFTW adaptation. DSpace, an open-source repository software package typically used for creating open-access repositories for scholarly and/or published digital content,[22,23] is used as a data cataloging platform. Since DSpace is open-source, we can customize DSpace so that it allows searching metadata extracted by the LISTER workflow.

The choice of eLabFTW as ELN reflects its design for the life sciences, which fits our user community, widespread use, active development, and facilitated access via the API, which allows us to access experiment documentation and to use container class entries for the metadata parsing implementation. An alternative ELN, Chemotion,[24,25] is more focused on the chemistry domain. Note, though, that the ELN Consortium,[26] involving ELNs and RDM-related works including Chemotion, eLabFTW, Chemedata,[27,28] Herbie,[29] Juliabase,[30] Kadi4Mat,[31] PASTA-ELN,[32] and SampleDB,[33,34] aims at making ELN entries interchangeable via the ELN data format.[26]

The choice of the ISA Model reflects its support in the communities of life sciences, biomedicine, and environmental research[8] and its straightforward and realistic structuring of heterogeneous experiments. We extended the model with container classes for *Project*, *Publication*, and *Protocol/Materials and Methods (MM)* (see *Extending the ISA model* below within this section, as well as later in the *Details of the Implementation Steps* section). This extension was motivated to map complex research environments, where more than one (funded) project addresses an overarching research question and to summarize those experiments that have been entered into a publication and store annotated *Protocol/MM* sections to facilitate experiment documentation.

While LISTER is independent of a specific data repository platform (i.e., the metadata output by LISTER can be attached to any type of data repository), we use DSpace[22] to store, manage, and catalog research data and metadata extracted by LISTER. DSpace has been specifically designed to provide storage, access, and preservation of digital archives on a long-term basis and is used by more than 2000 organizations worldwide.[35] Since DSpace is free and open-source software, it can be customized to create an adapted data preservation strategy. We run a customized DSpace instance that allows users to search the content of the contextual, LISTER-generated metadata along with the primary research data by its key, value, or key-value pair values as well as measure and unit when applicable. At present, this instance is intended for research data storage and cataloging within the scope of our University but will be extended to long-term storage and (hierarchical) public access (as mentioned in the fourth requirement mentioned above). Note that specifications of long-term storage may vary between laboratories and research environments because of differences in computing infrastructure and conventions, such as storage size and data management practices. LISTER-extracted metadata is not limited to use in connection with our own DSpace-based archival but can be adapted to or uploaded to other data repositories.

In the following three sections, we concisely introduce the main concepts behind LISTER, which will be described in more technical detail later.

Developing the Implementation Steps. Figure 1 depicts the interplay of the components of the LISTER-based RDM workflow, both internal and external to eLabFTW. A collection of *Protocol/MM* entries, annotated with markup, is cataloged

https://doi.org/10.1021/acs.jcim.3c00744
J. Chem. Inf. Model. 2023, 63, 6224−6238

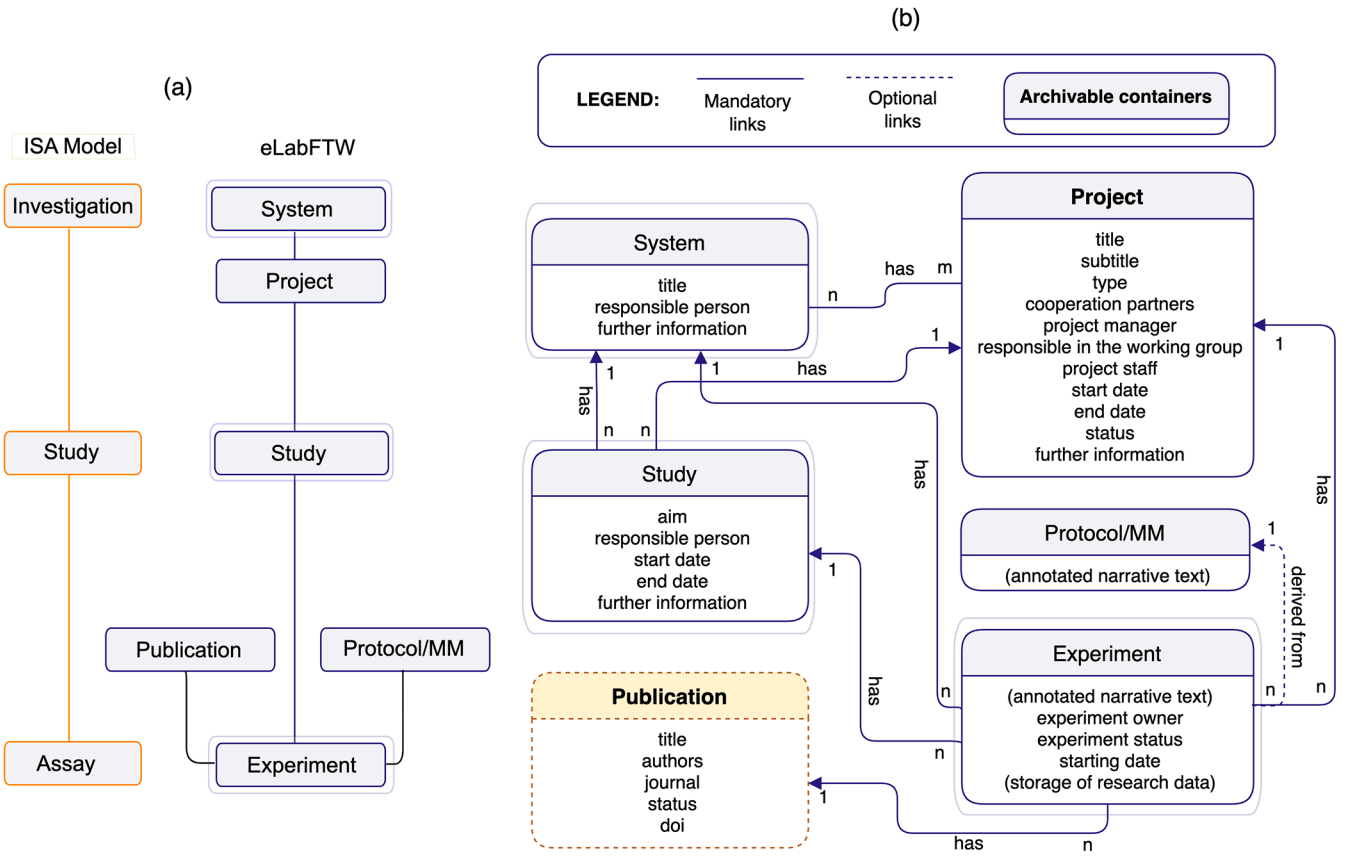


Figure 3. Incorporation of the ISA data model into the eLabFTW hierarchy of "class types". (a) Mapping between the ISA model (the orange boxes with a solid line on the left) and eLabFTW classes (boxes with gray outer line); for System and Study, specific classes were inherited from the Resource class in eLabFTW. The Resource class in eLabFTW allows the user to create their own class types with specific templates that can be defined for each created class type. In addition, the specific classes Project, Publication, and Protocol/MM were inherited from Resource. (b) Attributes of and relationships between specific class entries in eLabFTW. Protocol/MM and Experiment entries have no fixed attribute fields. Solid arrows indicate mandatory relationships, and dashed arrows indicate optional ones; "has" denotes an association, where the arrowhead points to the container. Metadata can also be parsed via its "container class", a class that can be used to group several experiments together (such as System, Study, Project, and Publication). In this paper, we illustrate the use of a container with a Publication entry (the orange rectangle with a dashed line), from which metadata extraction is started with the LISTER app. LISTER then extracts Experiment documentation metadata directly linked to the Publication entry and appends additional metadata from each Study, System, and/or Project entry linked to the Experiment. Entries linking (such as linking between Experiment and Resource classes) can be done directly as a feature in eLabFTW, and these linkages can be derived from its API.

within eLabFTW's custom *Resource*. A user can import a relevant *Protocol/MM* entry into the experimental documentation and modify the parameters in the annotations according to the actual usage without having to rewrite and annotate the experiment from scratch.

Figure 2 details how the set of requirements introduced above leads to the developed LISTER workflow. The complex requirement of semiautomatic extraction of metadata yields a decomposition into multiple implementation steps, whereas the requirements of accessible and long-term storage result in one joint step.

Extending the ISA Model. To accommodate more complex research settings in large working groups and research organizations, we extended the ISA model, providing more fine-grained class types in eLabFTW that include *System*, *Study*, *Project*, *Protocol/MM*, *Publication*, and *Experiment*. Each of these class types has its attributes and relationships. Particularly *Project*, *Publication*, and *Protocol/MM* serve as container classes to group experiments and summarize information for later comprehensive metadata extraction. We depict this information in Figure 3. The *Experiment* class directly corresponds to the *Experiment* entry type in eLabFTW. The remaining classes were created by using the *Resource* button in eLabFTW, a feature that facilitates the creation of custom class types. More explanations of this implementation can be found in the *Details of the Implementation Steps* section.

Facilitating Metadata Creation and Extraction. Users utilize annotation markups to compose the *Protocol/MM* sections, that way creating metadata while writing experiment documentation. These annotations serve as the foundation for metadata extraction with the LISTER app, which will be elaborated on in the *Implementation of the LISTER app* section. LISTER's annotation mechanism offers a method to designate parts of the experiment documentation that will be identified as key-value pairs, accompanied by an optional measure and unit fields for the extracted pairs. Illustrations of this annotation process and extraction results are listed in Figure 4. The extraction process outputs metadata in.json and .xlsx format, as well as annotation-free .docx format located in the specified path upon running the LISTER app. The .json format is intended for the data repository platform for search indexing, and the .xlsx and .docx formats provide human readability in

https://doi.org/10.1021/acs.jcim.3c00744
J. Chem. Inf. Model. 2023, 63, 6224−6238

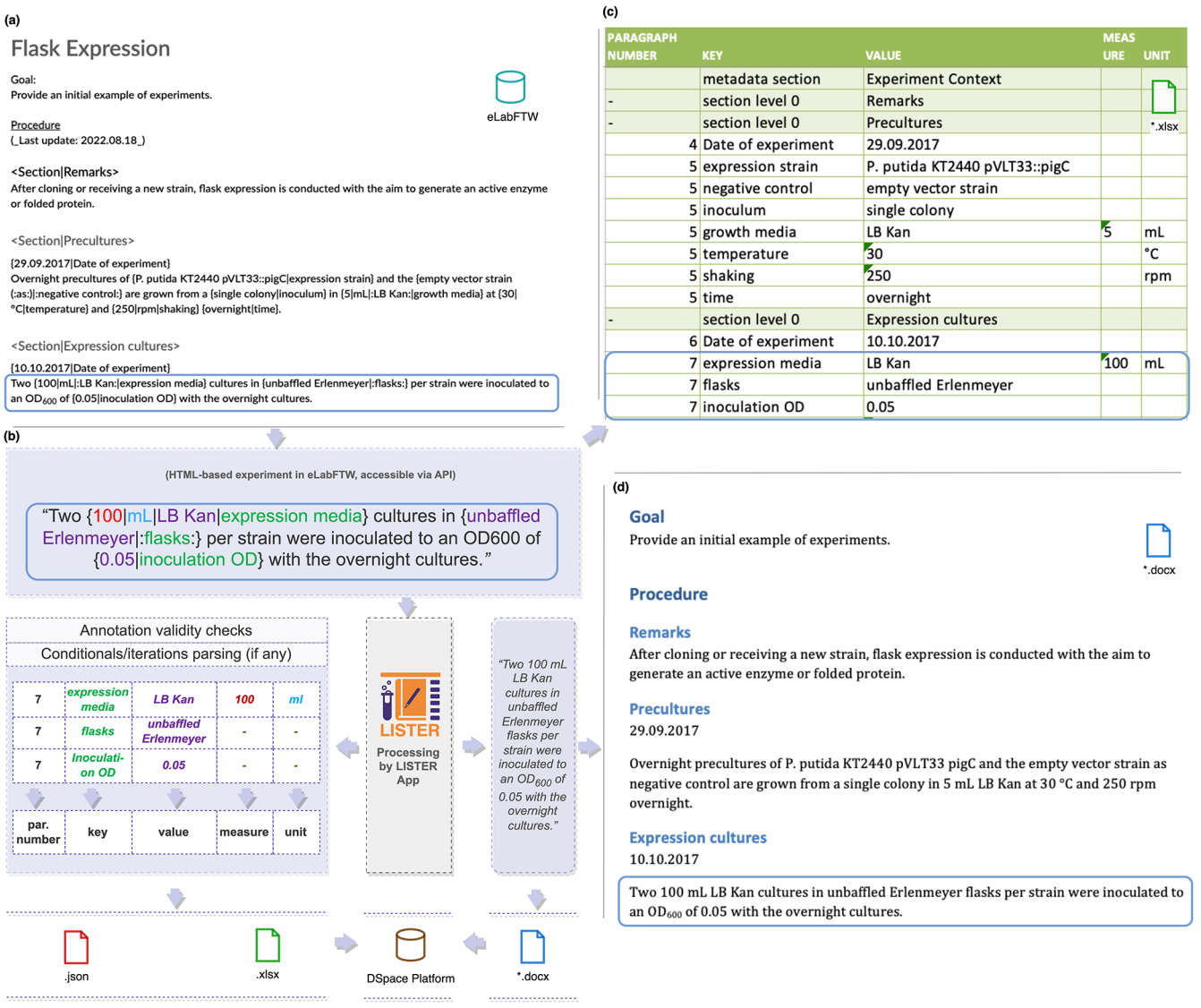


Figure 4. Overview of input/output to/from the LISTER pipeline. (a) Annotated and adapted experimental documentation is provided as an HTML page via the eLabFTW API as input to the LISTER app. (b) Overview of the results of metadata extraction with the LISTER app from annotated experiment documentation. The metadata is available in human-readable .xlsx format (c) and machine-readable.json format, and a .docx file (d) is created for documentation that is free of annotation marks intended for human readability.

tabular and descriptive formats, respectively. An archive can then be created in (intermediary) storage that includes both the original research data and the extracted metadata. We employ a customized version of DSpace as the data repository to host this archive. The metadata is searchable, and we can manage user access at a granular level.

After experiment documentation has been adapted from an annotated *Protocol/MM* entry, users provide the eLabFTW ID of the *Experiment* entry to the LISTER app's GUI (Figure 5), and the metadata will be extracted (Figure 4). Users can parse an individual experiment to extract the metadata and the annotation-free experiment documentation as well as a *container*, which is a set of experiments grouped into certain category types. In our implementation, these category types can be a *System*, *Study*, *Project*, and *Publication*. The parsing result will be saved to the path following the user-provided directory in the GUI, with experiment metadata parsed from a container grouped by the experiment title. A more detailed explanation regarding containers will be elaborated in the *Details of the Implementation Steps* Section.

The LISTER app and the documentation are available at https://github.com/CPCLab/lister. The GitHub repository is also linked with other repositories containing both the extended eLabFTW structure and the MM that can be imported to other eLabFTW instances to illustrate our implementation. Alternatively, the user can also browse the eLabFTW structure and the MM via our eLabFTW demo instance at https://elabftw.pharm.hhu.de/.

2.2. Details of the Implementation Steps. In Figure 2, the implementation steps (A)−(F) of the RDM workflow based on the above requirements are illustrated, which will be described now.

2.2A. Incorporating the ISA Data Model into eLabFTW. As the ISA model is an abstraction of three main levels of research activities (investigation, study, and assay) and the relationship among those levels, a more concrete adaptation of the ISA model is required for application within eLabFTW. Our

https://doi.org/10.1021/acs.jcim.3c00744
J. Chem. Inf. Model. 2023, 63, 6224−6238

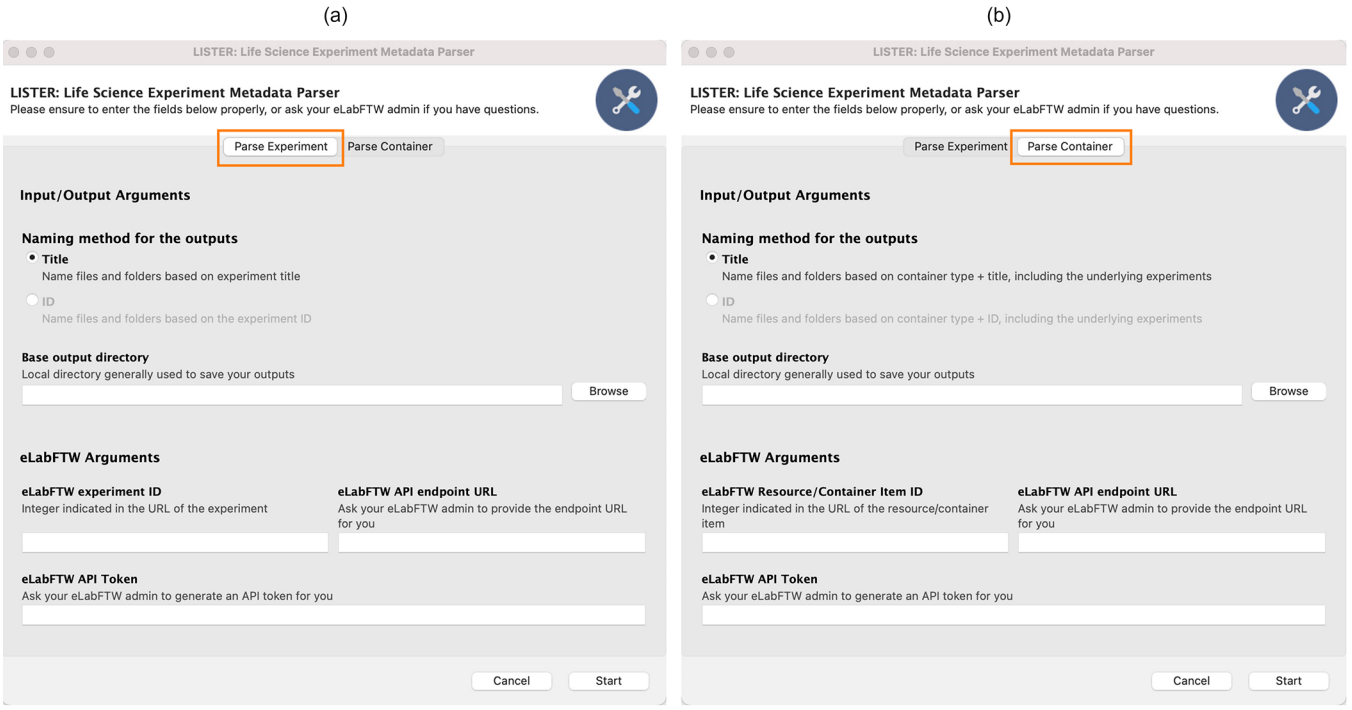


Figure 5. LISTER app's Graphical User Interface (GUI). The app has two interface tabs: (a) Experiment—in which users can extract metadata from an *Experiment* entry. (b) Container—in which users can extract metadata from a "*container* class", such as a *System*, *Project*, *Study*, or *Publication* entry. In the latter case, the metadata from a *System*, *Project*, or *Study* of the *Experiments* that are linked directly to the *Publication* will also be extracted.

adaptation provides a template for life science-centered research, although adaptation details might vary for different research groups, depending on the perceived meaning and granularity of the activity levels within the group. In addition, metadata regarding a *Project* (including project title and responsible person for the project) and Supporting Information (such as a *Publication*) is provided through an extension of this model within our eLabFTW implementation.

A.i. Mapping of the ISA Data Model to eLabFTW. The native eLabFTW entry types contain the following names: *Experiment* and *Resource*. Experiment is a fixed entry type, intended to be used to describe experiment documentation entries. ISA data model's *Assay*, by definition, can be mapped directly to eLabFTW's *Experiment*, which contains the experiment documentation. To map ISA's *Investigation* and *Study*, for which no direct equivalents are available in eLabFTW, we use eLabFTW's *Resource* type, a customizable abstract class from which a specific class implementation can be created via inheritance. To implement these classes in eLabFTW, an administrator defines what each class should contain. The defined class then has a template that requires the user to enter prespecified information (see next paragraph). We map ISA's *Investigation* to the specific class *System*, which represents the central molecular target or hypothesis investigated, among others. ISA's *Study* is mapped to the homonymous specific class in eLabFTW, which provides context regarding the study subject and characteristics and groups *Experiment* entries. The relationship among these three basic classes is shown in Figure 3a.

A *System* entry contains further information, such as the system name and the responsible person (Figure 3b). A *Study* entry contains information on the study's aim and the responsible person, along with the study's start and end date (Figure 3b). A *System* entry is a container for multiple *Study* and *Experiment* entries, and a *Study* entry is a container for multiple *Experiment* entries; the "has a" associations are realized via links in eLabFTW (Figure 3b). That way, questions can be asked such as "Which studies were performed for a system" or "Which experiments belong to a study or a system". Additionally, metadata to be created for an *Experiment* entry can be augmented by metadata from the *Study* and *System* containers, i.e., extracting metadata for an *Experiment* will also extract the metadata of a *System* and *Study* linked to that *Experiment*. The metadata of related classes will be appended to the metadata of the extracted *Experiment*.

A.ii. Extending the ISA Model within eLabFTW. We created three other classes (*Project, Publication*, as well as *Protocol/Material and Methods* (*MM*)) in the eLabFTW hierarchy from the abstract class *Resource* (Figure 3b). A *Project* complements the *System* class, and a *System* entry can have one or multiple *Project* entries associated. As the name implies, this class contains attributes related to a project such as cooperation partners or the project manager. A class can be solely independent of its own without having other classes categorized underneath, such as in the case of *Protocol/MM*. On the other hand, a class can be used as a "container" which groups several experiment entries or other class entries, such as *Project*, *System*, and *Study*. *Protocol/MM* serves as a library of the annotated template Protocol/MM, from which *Experiment* entries will be derived. We also define a *Publication* class, which is a container class to group experiments that have been reported in a publication (see point B: *Generating reusable experiment documentation* for details). That way, metadata and research data associated with that publication can be directly extracted (among others) for submission to the publisher or institutional archive (among others), which is also facilitated

https://doi.org/10.1021/acs.jcim.3c00744
J. Chem. Inf. Model. 2023, 63, 6224−6238

by the LISTER app. Each of the container classes (such as *System*, *Project*, *Study*, and *Publication*) has a two-column table containing metadata (see Figure 3b and SI Chapter 1/Table S1), which will be extracted as key-value pairs. The resulting metadata is appended to the *Experiment* metadata to enrich it with a more administrative context. While it is not mandatory, it is recommended that eLabFTW users group *Experiment* entries into (a) specific class(es) by linking an experiment to (a) class(es) in the bottom part of the experiment interface. This allows 1) the extraction of metadata from the linked class (like metadata about *System*, *Project*, and *Study*) along with the experiment itself, and appending the context about the class into the experiment metadata, as well as 2) bulk-extraction of experiments that belong to the same container class.

We intentionally kept redundant associations between classes. For example, while there is an indirect association between *Experiment* and *Project* via *Study* and *System* (see Figure 3b), we still keep the direct association between *Experiment* and *Project*, as this helps users of eLabFTW to identify the *Project* to which an *Experiment* belongs without having to go through the *Study* and *System* entries from the *Experiment*. Additionally, it disambiguates the *Project* corresponding to the *Experiment*, since the cardinality between *Project* and *System* is m:n.

In addition to linking experiments/class entries with other class entries/experiments in eLabFTW, we are using tags to organize entries to make them better findable by using the filtering mechanism provided in eLabFTW. Exemplary tags are shown in Figure 6, organized into three main categories: *lab-type tags* (dry lab versus wet lab), *system tags*, and *method tags*. They are numbered to indicate a hierarchy to ease using of the tags, although we do not require any tag standardization for our workflow to work. Creating these tags requires domain expertise and an overview of lab organization. Therefore, we gathered the expertise of senior researchers in our lab to create the conventions. For now, these tags are not based on an ontology or standardized terms, but this is planned for the future. Other categories than the three introduced here may be relevant for other laboratories, and each working group may need to define how their scientific workflow can be organized via customized hierarchical tags. Upon adding the tags to class entries, the eLabFTW GUI ensures the consistency of tag usage by providing an autocomplete functionality.

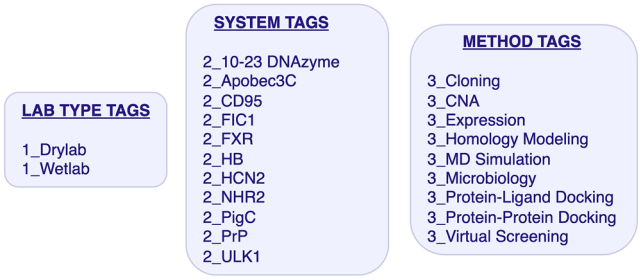


Figure 6. An exemplary implementation of a tag structure. Each experiment or relevant class entry (such as Protocol/MM) should contain tags, which are used to categorize the entries and make them better findable by using the filtering mechanism provided in eLabFTW.

2.2B. Generating Reusable Experiment Documentation. We distinguish two categories of experimental documentation: protocol versus MM descriptions. Protocols, also termed standard operating procedures (SOP), are step-by-step guidelines for how to conduct a specific experiment and are typically very detailed. MM provides a textual description typically found in journal articles; MM can often be generated by condensing the protocol. Research groups typically have catalogs of protocols or MMs to document how specific experiments are performed and to enable the reuse of these protocols. We use such protocols and MM catalogs (Figure 7), eventually redesigning them to become broadly reusable templates. In our case, the MM descriptions are used to derive experimental documentation in eLabFTW, from which MM sections for theses or paper manuscripts can be automatically generated via LISTER extraction into the .docx format.

The *Protocol/MM* entries (see Figure 8 as an example) should be structured in a way that mirrors the modular nature of experimental procedures in a laboratory, while avoiding excessive overlap or duplication. These entries are written by experienced lab members as a catalog of protocols/MMs for the whole lab members. While this process may take some time and require expertise, the templates can be reused efficiently when a related experiment is done, contributing to the *reusability* aspect of the FAIR RDM workflow. This saves scientists from writing *Protocol/MM* from scratch to document their experiments, although new experimental details might need to be adapted. Experiment documentation in eLabFTW can be derived from a template stored in the *Protocol/MM* class. To do this, the hashtag character followed by the title in the referred *Resource* entry is used, and eLabFTW automatically searches the entry title and autosuggests relevant entries. When changing or revising *Protocol/MM*, the revision history feature of eLabFTW can be leveraged, such that one can use permanent links to refer to particular versions of a *Protocol/MM*. This enables users to access the version of *Protocol/MM* at the time of referencing, even when updates were made in the meantime.

2.2C. Annotating Reusable Experiment Documentation for Automated Metadata Extraction. To allow automated extraction of metadata (such as key-value pairs, which can be augmented with measure and unit information; see Figure 4b), the protocol and MM templates are annotated with markups of the LISTER annotation language. An illustration of the implementation of eLabFTW is provided in Figure 8. The key-value pairs can be extracted with the LISTER extraction app (see Figure 4b for an overview of the annotation mechanism, and Figure 1 for the overall interplay between these components). Metadata generation is necessary before providing research data for archival in institutional, publisher, or community-wide repositories to fulfill the FAIR principles. The combination of annotated reusable experiment documentation and automated metadata extraction alleviates the time-consuming, cumbersome, and error-prone hurdle of annotating primary research data with metadata by hand each time such research data is generated.

C.i. Annotating Reusable Experiment Documentation. Domain experts in the respective research fields or working groups will add markups to the protocol or MM templates adhering to the format of the LISTER annotation language. A brief example of an annotated MM is given in Figure 4a–d and Figure 8; more detailed descriptions are given in the *Methods* subsection as well as in SI Chapter 2, Table S2, and SI Chapter 3, with screenshots of the structure and annotation provided in Figure S1 and Figure S2, respectively. Figure 7 provides an

https://doi.org/10.1021/acs.jcim.3c00744
J. Chem. Inf. Model. 2023, 63, 6224−6238

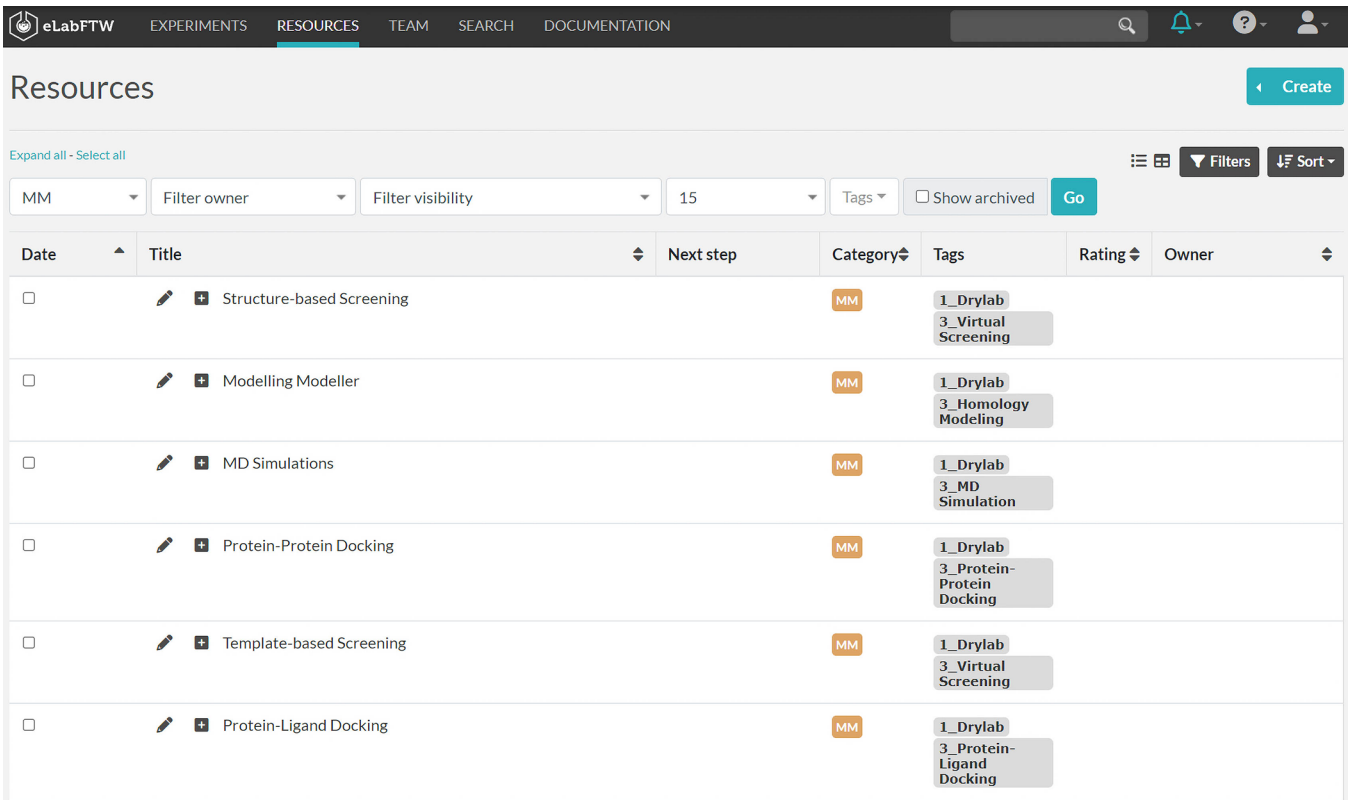


Figure 7. An example of an MM catalog is in our implementation. Each of the MM titles describes certain procedures uniformly.

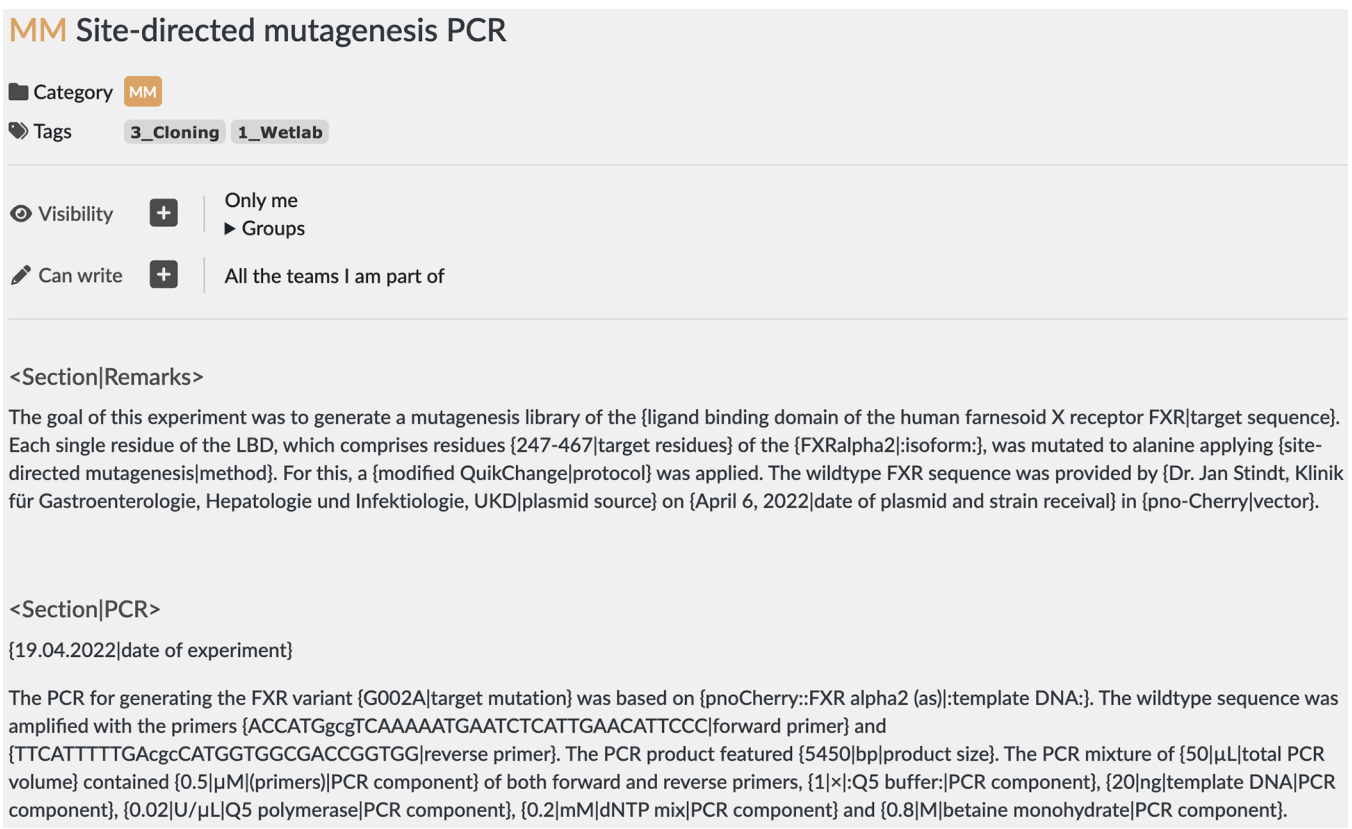


Figure 8. An example of the MM content. The MM describes the materials and methods for an experiment using LISTER annotation markup. The MM can be imported to relevant experiment documentation, and the values of the key-value pairs are then adapted to reflect the actual values used for the experiment. Based on this annotation, the LISTER app extracts the contextual metadata used in the experiment.

https://doi.org/10.1021/acs.jcim.3c00744
J. Chem. Inf. Model. 2023, 63, 6224−6238

example of how a particular MM within eLabFTW is structured. The design of the LISTER annotation language was guided by simplicity principles, allowing researchers to use the implementation without having a steep learning curve. The LISTER annotation preserves the readability of the text and, that way, mimics other markup languages such as Markdown or the long-known HTML style (but with fewer annotation symbols striving for more readability). It allows the inclusion of comments, iterations, and conditionals. Some additional principles for writing an annotated protocol or MM are described in SI Chapter 4.

C.ii. Adapting Parameters and Extracting Metadata from Annotated Experiment Documentation. After scientists import the annotated protocol or MM templates relevant to their experiment from the *Protocol/MM* class into their *Experiment* entry (Figure 4a), they likely need to adapt the predefined parameters according to the experiment details. Additionally, irrelevant parts of the templates should be removed, and new parts can be added, thereby keeping compliant with the format of the LISTER annotation language. The LISTER app checks the syntax of the annotation markups with respect to unmatched brackets or the number and types of elements in a key-value pair, among others, as a validation mechanism upon parsing the experiment entries. This will yield either a warning message (when the issue does not affect the validity of the output) or an error message (when the issue affects the validity of the output), pointing to the problematic line(s) in the evaluated experiment entry (SI Chapter 2). After the check, the LISTER app extracts metadata from the annotated experimental documentation (Figure 4b) and the log file along with more human-readable experimental documentation (Figure 4d). To do this, users are required to fill out the parameters Experiment ID, output path, and eLabFTW API token, among others, in the LISTER app's GUI (Figure 5). The LISTER app needs to be executed by the user only when metadata needs to be extracted.

C.iii. Parsing a Group of Experiments under the Same Container Class. *Experiment* entries can be parsed for metadata individually via LISTER's GUI (Figure 5a). Additionally, another important requirement is to parse the content of a container class, such as a *Publication* entry (Figure 5b). As mentioned in the *Details of the Implementation Steps* section, a *Publication* is a container class and can be linked to several *Experiment* entries, which were conducted for the publication, and implicitly to *System* entries, *Project* entries, and *Study* entries (Figure 3a). In the *Resource* tab in LISTER's GUI (Figure 5b), the eLabFTW ID of a *Publication* entry can be provided, and LISTER will extract the metadata output from all experiment documentation under that publication, as well as from the associated *System*, *Project*, and *Study* entries.

2.2D. Using Well-Known Input and Output Formats. As input, LISTER parses the content of annotated, adapted experimental documentation in eLabFTW provided as HTML pages (Figure 4a) via the eLabFTW API. LISTER transforms the content of annotated documentation into several outputs (Figure 4b–d): (I) Experiment documentation as "clean" text without annotations to be used as MM sections in theses and manuscripts is provided in .docx format. The .docx format was chosen due to its familiarity among most users, its widespread support across various office suites, and because it can be converted to other formats (such as .pdf); (II) Contextual experiment metadata is provided in .xlsx and .json formats, the first as human-readable version and the second as the machine-readable version to be used as input for DSpace's data cataloging platform. The selection of the .xlsx format was based on its extensive usage for storing tabular spreadsheets and the availability of programming libraries that support reading and creating this format. While the .csv format is an alternative, users often need to manually select the appropriate delimiter and encoding when working with .csv-based files.

2.2E. Configuring Input and Outputs in Graphical User Interface (GUI). Although it is possible to provide metadata parsing parameters through the command prompt or terminal, this approach may be less user-friendly for general users. To address this, the LISTER app offers a graphical user interface (GUI) that simplifies the process (Figure 5). The GUI also allows users to load parameters conveniently through a .json configuration file, containing information such as the eLabFTW API token as well as the end point URL.

2.2F. Providing Generic Storage for Research Data and Its Contextual Metadata. The research data, along with the extracted metadata, are consolidated and organized into a searchable data archive. In our case, this archive is stored in a customized data repository built on the DSpace platform.

2.3. Methods. *2.3.1. LISTER Annotation Language.* We briefly summarize LISTER annotation elements here (see also Figure 4a and Figure 8 for examples) and provide full documentation of the annotation mechanism on LISTER's GitHub repository (https://github.com/CPCLab/lister) and in the SI (SI Chapter 2, with an illustration of the MM provided in the SI Chapter 3 and Figure 8). The annotation process takes place within eLabFTW, following a specific convention outlined in this section. Although it is not feasible to create an eLabFTW plugin for in-editor validation, we perform a validation process when the user runs the LISTER app to parse the metadata. Depending on the severity of the issues found, the app will either continue with a warning message or terminate. The corresponding warning or error messages are displayed and saved in a log file, allowing users to identify the underlying problem in the experiment documentation. The following points elaborate how the LISTER annotation mechanism is designed.

2.3.1A. Basic Elements. A key-value (KV) pair is represented as {value|key} in an experiment documentation entry. A KV pair can be extended with a measure (denoting the measured quantity) and the unit, if necessary. Due to this extension, there are two more variations on how a KV pair may be written: {measure|unit|key} (in which the measure and unit will be mapped into value, with the unit at the end) and {measure|unit|value|key} (in which the measure and unit will be taken as given) (see Figure 4a for an illustration).

2.3.1B. Key Visibility in .docx Output. In most cases, it is superfluous to have the *key* part of a KV pair available in the .docx output, as illustrated in Figure 4b for the key "expression media". Hence, keys in the experiment documentation are hidden by default in the.docx output. However, if the key needs to be explicitly shown in the.docx output, users can indicate this by wrapping the key with colons ": . . .":, i.e., {value|:key:} (Figure 4a).

2.3.1C. Order. Each extracted KV pair is assigned an "order designator" to disambiguate the order mapping of the keys. The order designator is derived from the paragraph number where the KV pair appears (Figure 4b).

2.3.1D. Comments. There are three different types of comments supported in LISTER.

https://doi.org/10.1021/acs.jcim.3c00744
J. Chem. Inf. Model. 2023, 63, 6224−6238

D.i Comments are parsed as-is, retaining brackets and content in the.docx output. This is used to retain comments without modifications in the eLabFTW experiment documentation entry and in the .docx output. Such comments are marked as "(this example)", which will be written as (this example) in the.docx output, whereas nothing is written in the metadata output (see SI Chapter 2).

D.ii Invisible comments with removed annotations. This is used when additional notes need to be specified in Protocol/MM that should be hidden from the.docx output. Typical examples are to detail 1) the meaning of a specific parameter, 2) the Protocol/MM entry author(s), or 3) the Protocol/MM entry version. Such comments are marked as "(_invisible comment_)".

D.iii Comments for which the content is retained but not the brackets. This is used for comments inside KV pairs. For example, "the {empty vector strain (:as:)|:negative control:}" will be written as "the empty vector strain as negative control" in the.docx output, with "negative control" as the key and "empty vector strain" as the value.

2.3.1E. Conditionals and Iterations. LISTER supports documenting conditionals and iterations in protocol and MM templates. Conditionals may be used when a step has multiple possibilities, with each possibility leading to a specific result, further steps to take, or termination. Iterations may be used when one or more steps need to be done repetitively until one or more specific condition(s) is (are) satisfied. See Table 1 for the three supported iteration types. Nonetheless, these elements should be used with caution in the final experiment documentation. When adapting the templates, researchers are encouraged to resolve conditionals by documenting the actual results or steps taken, thereby removing the alternatives from the experiment documentation. Likewise, for iterations, researchers are encouraged to document the respective repetitive steps explicitly.

Table 1. Three Supported Types of Iteration Are Shown in[a], along with Examples and Extracted Keys

Types	Example	Keys extracted	Value extracted
While	<while|pH|lte|7>	step type	iteration
		flow type	while
		flow parameter	pH
		flow logical parameter	lte
		flow compared value	7
		flow type	iterate
		flow operation	+
		flow magnitude	1
For	<for|pH|[1−7]|+|1>	step type	iteration
		flow type	for
		flow parameter	pH
		flow range	[1−7]
		start iteration value	1
		end iteration value	7
		flow operation	+
		flow magnitude	1
For each	<for each |generated pose>	step type	iteration
		flow type	for each
		flow parameter	generated pose

[a]Each iteration type has its own set of extracted keys, which, in some cases, are implicitly defined to provide more clarity in the resulting metadata. Please refer to the LISTER GitHub documentation and SI Chapter 2/Table S2 for an extended annotation table including the LISTER general annotation mechanism.

2.3.1F. Reference Management. A reference in the LISTER annotation is supported by using a bracketed DOI (SI Chapter 2). The annotation is parsed and listed as a numerical reference annotated in squared brackets in the text, with the referenced publications provided as a list of DOI at the end of the .docx document. DOIs are recognized based on their patterns using regular expressions.

2.3.1G. Sections. The <section|section name> annotation is designated to provide separation between sections.

An entry of the *Experiment* type is parsed according to the above LISTER annotation rules. By contrast, *System*, *Project*, *Publication*, and *Study* entries are parsed for their attributes given in Figure 3, i.e., the table in such an entry is transformed into KV pairs.

2.3.2. Implementation of the LISTER App. LISTER is open-source under the GPLv3 license[36] and implemented using Python 3.9. Metadata elements (key, value, measure, unit, comment, conditional/iteration, reference, and section/subsection) are identified through regular expressions. We used the following Python libraries to develop LISTER: BeautifulSoup[37] for parsing HTML content, elabapy[38] and elabapi-python[39] to communicate with the eLabFTW API end point, json, xlsxwriter,[40] and python-docx[41] to write files in .json, .xlsx, and .docx formats, respectively. Gooey,[42] a Python library to create a GUI on top of a command line, is used to provide a graphical layer for LISTER. Other utility libraries used are re (regular expression), enum (enumeration), os, PyInstaller application packager,[43] ssl, platform, pathlib, pandas, and lxml.[44] For utilizing LISTER without having to configure Python and install Python libraries, LISTER has been packaged for different operating systems: Windows 10 and 11, Ubuntu-based Linux distributions (tested on Linux Mint 21), and macOS (tested on macOS v10.12, 12.4, and 13.0) for both intel and M1/M2 chips. The code documentation is provided using the reStructuredText (reST) format.[45]

Parameters detailing the respective input source for syntax checking and metadata extraction are provided via the GUI (Figure 5a,b). For experiment documentation coming from eLabFTW, the experiment ID (which is indicated on the URL of the experiment), API end point URL, and eLabFTW API Token are necessary. Both the API end point URL and eLabFTW API Token can be obtained from the administrator of the eLabFTW instance. The outputs are provided in the user-specified output directory. An accompanying "config.json" file allows preloading parameters to the GUI, including the output file/directory names, and eLabFTW-specific parameters (such as experiment ID, API end point URL, and eLabFTW API Token) to ease the startup with the LISTER app or allow batch processing.

3. EVALUATION

3.1. Implementation. As proof of concept and for our research documentation, we generated 11 annotated MM templates for the domain of computational biophysical chemistry and 4 templates for the domain of protein

https://doi.org/10.1021/acs.jcim.3c00744
J. Chem. Inf. Model. 2023, 63, 6224−6238

biochemistry and molecular biology. Generation of these templates took about 80 man-hours, including cross-correction rounds, accompanied by six meetings to define the granularity and scope of the MM. These MM templates are being used in our research group to follow the designed RDM workflow. The MM templates are available at https://github.com/CPCLab/materials-and-methods to initiate the development of research group-specific templates. An illustrative example is provided in https://github.com/CPCLab/lister-container, containing a *Publication* entry within eLabFTW and its corresponding *System*, *Project*, *Study*, and *Experiment*, which are provided as an eLabFTW export-containing ELN file. The extraction of the metadata took about 8 s for an experiment linked with corresponding entries (such as *System*, *Project*, and *Study*), and 20 s for a *Publication* containing experiments with over 30 attachments in total, each linked with corresponding entries.

During the initial phase, we collected feedback from our working group members on various aspects, including the workflow, annotation mechanism, and potential output requirements. This iterative feedback process took place simultaneously with the app development and eLabFTW adaptation. Once the workflow had been integrated into the app and the feedback had been incorporated, we also arranged meetings with other working groups in diverse life science fields such as microbiology, bioinformatics, and biochemistry. These meetings allowed us to gather further feedback, such as insight on the importance of annotation-free and human-readable .docx documentation and that in most cases the key part in the KV pair is not relevant for human-readable experiment documentation; hence, we decided to make it invisible in the .docx file by default. The users are trained via live demos on several occasions, and the users can create and derive experiments by themselves. In most cases, users are using the default key-value pairs without advanced functionality such as using DOI references. However, we provide detailed documentation regarding the annotation usage on our GitHub page. We also set up a demo eLabFTW server for users to try the LISTER workflow at https://elabftw.pharm.hhu.de, and the config .json file is shared at https://github.com/CPCLab/lister, including the manual to reuse/modify the configuration file.

3.2. Related Works. Research data have to be stored in either general-domain or domain-specific data repositories for access. Several general-domain data repositories have been established for public use, such as Zenodo,[46] Dryad,[47] FigShare,[48] and Open Science Framework (OSF).[49] A comparison of some of these repositories as to the storage quota on its free tier, possible storage extension and additional costs, DOI provision, funding source, the base of operations, and whether it is required to make contextual metadata available is given in SI Chapter 5 and Table S3. To make the published data more compliant with FAIR guidelines, key-value-based contextual metadata can be added. However, the contextual metadata of uploaded research data items is often not available, and when it is available, the content of contextual metadata, uploaded as an additional file(s), is not necessarily searchable in these portals. The LISTER workflow provides contextual metadata along with the research data, and the customized Dspace instance allows searching it.

Complementary to the general-domain data repositories, there are also specific-domain data portals such as the *Protein Data Bank* (*PDB*), *Universal Protein Database* (*UniProt*), and *Ensembl*. *PDB*[50] is a well-established repository of resolved structures of biomolecules and their associated research data as well as derived data. *UniProt*[51] is a protein sequence database organized as UniProt Knowledge Base, Archive, and Reference Clusters. *Ensembl*[52] is a database for genomic information. In these cases, the repositories' content can be browsed by specific annotations,[53−55] which utilize vocabularies or ontologies organized by the Gene Ontology Consortium[56] for semantic categorization.

Providing eLabFTW *Experiment* entries with ID to those specific-domain data portals (such as PDB, UniProt, and Ensembl) would align the experiment with specific, standardized, and highly curated entries of the specific research entity, allowing conceptual interlinking between eLabFTW entries and the entries in the specific-domain data portals. In this context, we propose LISTER to allow the extraction of experiment documentation into structured key-value pair metadata. While at this stage LISTER does not facilitate semantic annotation of the extracted key-value pair metadata, LISTER's workflow lays the groundwork for semantic alignment by making contextual metadata available in the first place.

With regards to related works in domain-specific RDM workflow concepts, the NFDI consortium DataPLANT[57] developed technology stacks to manage research data in plant science, namely *Annotated Research Context* (ARC) and *Swate*. ARC provides a packaging mechanism for research data that includes measurement data, metadata, data annotations, tools, and scripts surrounding the research cycle in plant science. In addition to ARC, Swate[58] is a plugin implemented for Microsoft Excel that allows annotating research using contextual metadata with alignment to specific and standardized ontologies.[57] Compared to either approach, LISTER does the packaging/archival of experiment documentation through the ELN and thus is more tailored and integrated toward ELN users. Furthermore, LISTER disentangles the creation of metadata and ontology alignment into two steps, with the first supported by protocol/MM templates and the second, as a future work, intended to be automated semantic metadata annotation via the use of external semantic terminology services, such as the service provided by the NFDI4Chem.[59,60] We envision that this way LISTER reduces the barrier of creating metadata for experiment documentation, albeit without immediate semantic annotation as a trade-off.

The work by Kunis et al.[61] makes use of data generated by lab equipment, specifically from the microscopy field. This is done by extracting metadata from research equipment output—such as the log files of a microscope and recording details—in connection with microscopy data, which later may be augmented with additional metadata.[61] At this stage, LISTER does not directly facilitate extracting metadata from specific input/output files generated by lab equipment but changes to its architecture to support independent plug-ins for such extraction are considered for future works.

LISTER bears some similarity to the work of Schröder et al.,[21] which also uses eLabFTW to store experiment documentation. The authors provide a proof of concept in the Calcium imaging domain by implementing an automated semantic metadata extraction from a manually engineered protocol with a canonical experiment entry structure. The protocol requires domain expertise to write, also to ensure that the protocol and resource elements are mapped to the correct vocabulary term or an ontological class instance. While this approach directly provides semantic annotation for a protocol,

https://doi.org/10.1021/acs.jcim.3c00744
J. Chem. Inf. Model. 2023, 63, 6224−6238

it requires that domain ontologies for annotating the protocol already exist and that the protocol writer has the background to associate the correct ontological annotations to the protocol. These two requirements are not necessarily met in other scenarios; therefore, LISTER's workflow simplifies the annotation mechanism process with fewer technical barriers by focusing on regular metadata instead of semantic metadata. The semantification of this regular metadata can be done later once the regular metadata has been collected and cataloged.

ChemDataExtractor[62] is a toolkit that extracts metadata from the chemical domain. It relies on natural language processing pipelines (including tokenization, part-of-speech tagging, and named entity recognition) to process HTML/XML/PDF formats for outputting chemical records and publication metadata. PDFDataExtractor[63] is a plugin for ChemDataExtractor that utilizes patterns found in several chemistry journals to detect general-domain metadata (such as title, abstract, DOI, journal, keywords, author, sections, captions, and references) based on layout position analysis. Compared to these tools, LISTER requires more intervention by requiring annotated experiment templates, but it also yields more fine-grained metadata regarding the context of the experiment; hence, it suits more the reproducibility goal. LISTER is also more streamlined toward the experiment pipeline and data archival through the container concepts.

The Helmholtz Metadata Collaboration (HMC) focuses on facilitating research data documentation and handling across various fields for the Helmholtz Association. The HMC produces several tools and services ranging from, among others, FAIR data publication guidelines, metadata specification/validation/structuring/sharing tools, and services to find metadata standards and provides persistent URLs.[64] LISTER, as a metadata extraction workflow, could potentially be aligned with tools such as the HMC metadata specification and validation tool. This alignment could facilitate validation of the extracted metadata.

3.3. Compatibility with RDM Principles. How does LISTER comply with the FAIR[6] guidelines? In terms of *findability*, LISTER's RDM workflow is able to create rich contextual metadata by using semiautomatic metadata extraction; the metadata can later be indexed to enable a search over the data cataloging platform DSpace. In terms of *accessibility*, the research data are published on the web using DSpace, and the HTTP/S protocol is used to access it. Regarding *interoperability*, metadata are serialized using the .json format for data exchange. Metadata are also provided in .xlsx format for human readability. We do not yet use ontologies to represent the metadata in the Resource Description Framework (RDF)[65] format, i.a., due to the limited availability of ontologies in our pilot domain computational biophysical chemistry. RDF allows information to be represented as a triple unit consisting of the subject, predicate, and object. Each triple can be connected to other triples, yielding interconnected triples as linked data in the form of a knowledge graph, which can then be linked to another knowledge graph, making connected data well-integrated. We intend to incorporate alignment with ontologies or use the output from LISTER's metadata extraction for collecting ontological terms in the future. Finally, concerning *reusability*, accurate and relevant attributes can be extracted using LISTER as it directly parses experiment documentation derived from annotated Protocol/MM and adapted by the researcher, which should minimize the number of inaccuracies or omissions.

4. CONCLUSION AND FUTURE WORK

We introduced LISTER as a methodological and algorithmic solution to extract metadata from research data using minimum efforts and making the research data accessible and downloadable. LISTER is tailored to the field of life sciences, makes use of existing RDM standards and platforms, and consists of four components: LISTER annotation language, customized eLabFTW entries, a "container" concept in eLabFTW, and an app to enable easy-to-use, semiautomated metadata extraction from eLabFTW entries. For our research documentation and as a showcase, we apply LISTER to computational biophysical chemistry, protein biochemistry, and molecular biology. Our concept of reusable Protocol/MM templates to derive documentation from experiments and extract metadata from annotated experiment entries should also be extendable to other life science areas.

Future work will aim at standardizing terms that occur in the extracted experimental metadata; such terms can then be aligned with existing ontologies or could also be used to extend related ontologies. We envision that deep-learning-based approaches for the automatic labeling of keys or units in experimental documentation might become available when enough training data for a specific domain exists. LISTER may contribute to the generation of such training data. In addition, we aim to support a plug-in-based approach to enable third-party developers to add their implementation to parse metadata for their specific use-case, such as those provided by specific software or lab equipment. We will also investigate approaches to align extracted metadata from LISTER with existing semantic terminology services.

■ ASSOCIATED CONTENT

Data Availability Statement

The LISTER source code is available at https://github.com/CPCLab/lister. eLabFTW version 4.5.14 used here is available at https://www.elabftw.net/. The MM templates generated in this work are available at https://github.com/CPCLab/materials-and-methods and the class definitions for our eLabFTW adoption are available at https://github.com/CPCLab/lister, which can be imported to other eLabFTW instances as well.

Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acs.jcim.3c00744.

> Technical details of LISTER, writing principles for Protocol/MM, an example of how to structure and annotate MM in eLabFTW, and a comparison of features of public research data repositories (PDF)

■ AUTHOR INFORMATION

Corresponding Author

Holger Gohlke − *Institute for Pharmaceutical and Medicinal Chemistry, Heinrich Heine University Düsseldorf, 40225 Düsseldorf, Germany; Institute of Bio- and Geosciences (IBG-4: Bioinformatics), Forschungszentrum Jülich GmbH, 52425 Jülich, Germany;* orcid.org/0000-0001-8613-1447; Phone: (+49) 211 81 13662; Email: gohlke@uni-duesseldorf.de, h.gohlke@fz-juelich.de

https://doi.org/10.1021/acs.jcim.3c00744
J. Chem. Inf. Model. 2023, 63, 6224−6238

Authors

Fathoni A. Musyaffa − *Institute for Pharmaceutical and Medicinal Chemistry, Heinrich Heine University Düsseldorf, 40225 Düsseldorf, Germany*

Kirsten Rapp − *Institute for Pharmaceutical and Medicinal Chemistry, Heinrich Heine University Düsseldorf, 40225 Düsseldorf, Germany*

Complete contact information is available at: https://pubs.acs.org/10.1021/acs.jcim.3c00744

Author Contributions

F.M. conceptualization, investigation, programming, analysis, visualization; K.R. analysis, project management; H.G. conceptualization, supervision, analysis. The manuscript was written with the contributions of all authors. All authors have given approval for the final version of the manuscript.

Funding

This study was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) project no. 267205415/CRC 1208, subproject INF to H.G.

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

We are grateful to the researchers who have contributed to writing the Materials and Methods templates to be further processed by LISTER, Stephan Schott-Verdugo, Michele Bonus, Jesko Kaiser, Christoph Gertzen, Stefanie Brands, Jonas Dittrich, Christopher Pfleger, Christina Gohlke, and Filip König. This work is guided by our experiences in the Collaborative Research Center (CRC) 1208. We thank Lutz Schmitt, Stefanie Weidtkamp-Peters, and other CRC1208 members for continued discussions. We are grateful for discussions with and support from the Research Data Management Team (Dirk Fleischer, Rafael Dellen, Christian Hohenfeld) and the Zentrum für Informations- und Medientechnologie (Nina Knipprath, Bert Zulauf, Sebastian Manten, and Thomas Dziurzyk) at Heinrich Heine University Düsseldorf.

■ ABBREVIATIONS

ACS, American Chemical Society; API, Application Programming Interface; ARC, Annotated Research Context; DFG, Deutsche Forschungsgemeinschaft (German Research Foundation); DOI, Digital Object Identifier; eLabFTW, electronic Lab For The World; ELN, Electronic Laboratory Notebook; FAIR, Findable, Accessible, Interoperable, and Reusable; GUI, Graphical User Interface; HMC, Helmholtz Metadata Collaboration; HTML, HyperText Markup Language; HTTP/S, Hypertext Transfer Protocol/Secure; ID, Identifier; ISA, Investigation-Study-Assay; JSON, JavaScript Object Notation; KV, Key-Value; LISTER, (LI)fe (S)cience Me(t)adata Pars(er); MM, Materials and Methods; NFDI, Nationale Forschungsdateninfrastruktur (National Research Data Infrastructure); NFDI4Chem, NFDI for Chemistry; NSF, National Science Foundation; OSF, Open Science Framework; PDB, Protein Data Bank; RDF, Resource Description Framework; RDM, Research Data Management; SOP, Standard Operating Procedures; UniProt, Universal Protein Database; URL, Uniform Resource Locator

■ REFERENCES

(1) Baker, M. 1,500 Scientists Lift the Lid on Reproducibility. *Nature* **2016**, *533*, 452−454.

(2) Deutsche Forschungsgemeinschaft: DFG. *Leitlinien Zum Umgang Mit Forschungsdaten*https://www.dfg.de/download/pdf/foerderung/grundlagen_dfg_foerderung/forschungsdaten/leitlinien_forschungsdaten.pdf (accessed Oct. 24, 2022).

(3) Nationale Forschungsdaten Infrastruktur. https://www.nfdi.de (accessed Sept. 11, 2022).

(4) National Science Foundation. *Dissemination and Sharing of Research Results - NSF Data Management Plan Requirements*. https://www.nsf.gov/bfa/dias/policy/dmp.jsp (accessed Jan. 24, 2022).

(5) American Chemical Society. *ACS Research Data Policy*. https://publish.acs.org/publish/data_policy (accessed Sept. 11, 2022).

(6) Wilkinson, M. D.; Dumontier, M.; Aalbersberg, Ij. J.; Appleton, G.; Axton, M.; Baak, A.; Blomberg, N.; Boiten, J.-W.; da Silva Santos, L. B.; Bourne, P. E.; Bouwman, J.; Brookes, A. J.; Clark, T.; Crosas, M.; Dillo, I.; Dumon, O.; Edmunds, S.; Evelo, C. T.; Finkers, R.; Gonzalez-Beltran, A.; Gray, A. J. G.; Groth, P.; Goble, C.; Grethe, J. S.; Heringa, J.; 't Hoen, P. A. C.; Hooft, R.; Kuhn, T.; Kok, R.; Kok, J.; Lusher, S. J.; Martone, M. E.; Mons, A.; Packer, A. L.; Persson, B.; Rocca-Serra, P.; Roos, M.; van Schaik, R.; Sansone, S.-A.; Schultes, E.; Sengstag, T.; Slater, T.; Strawn, G.; Swertz, M. A.; Thompson, M.; van der Lei, J.; van Mulligen, E.; Velterop, J.; Waagmeester, A.; Wittenburg, P.; Wolstencroft, K.; Zhao, J.; Mons, B. The FAIR Guiding Principles for Scientific Data Management and Stewardship. *Sci. Data* **2016**, *3*, 160018.

(7) Heil, B. J.; Hoffman, M. M.; Markowetz, F.; Lee, S.-I.; Greene, C. S.; Hicks, S. C. Reproducibility Standards for Machine Learning in the Life Sciences. *Nat. Methods* **2021**, *18*, 1132−1135.

(8) Plesser, H. E. Reproducibility vs. Replicability: A Brief History of a Confused Terminology. *Front. Neuroinformatics* **2018**, *11*, 76.

(9) Association for Computing Machinery. Artifact Review and Badging. https://www.acm.org/publications/policies/artifact-review-badging# (accessed July 14, 2023).

(10) Patil, P.; Peng, R. D.; Leek, J. T. A Statistical Definition for Reproducibility and Replicability. *bioRxiv*, July 29, 2016. DOI: 10.1101/066803

(11) Cacioppo, J. T.; Kaplan, R. M.; Krosnick, J. A.; Olds, J. L.; Dean, H. *Social, Behavioral, and Economic Sciences Perspectives on Robust and Reliable Science*; Report of the Subcommittee on Replicability in Science Advisory Committee to the National Science Foundation Directorate for Social, Behavioral, and Economic Sciences; National Science Foundation, 2015.

(12) Goodman, S. N.; Fanelli, D.; Ioannidis, J. P. A. What Does Research Reproducibility Mean? *Sci. Transl. Med.* **2016**, *8*, 341.

(13) Sansone, S.-A.; Rocca-Serra, P.; Gonzalez-Beltran, A.; Johnson, D.; ISA Community. *ISA Model and Serialization Specifications 1.0*. **2016**, DOI: 10.5281/zenodo.163640.

(14) Briney, K.; Coates, H.; Goben, A. Foundational Practices of Research Data Management. *Res. Ideas Outcomes* **2020**, *6*, No. e56508.

(15) Harvard Medical School Longwood Research Data Management. *Electronic Lab Notebooks*. https://datamanagement.hms.harvard.edu/collect-analyze/electronic-lab-notebooks (accessed July 14, 2023).

(16) Riley, E. M.; Hattaway, H. Z.; Felse, P. A. Implementation and Use of Cloud-Based Electronic Lab Notebook in a Bioprocess Engineering Teaching Laboratory. *J. Biol. Eng.* **2017**, *11*, 40.

(17) Kanza, S.; Willoughby, C.; Gibbins, N.; Whitby, R.; Frey, J. G.; Erjavec, J.; Zupančič, K.; Hren, M.; Kovač, K. Electronic Lab Notebooks: Can They Replace Paper? *J. Cheminformatics* **2017**, *9*, 31.

(18) Carpi, N. *eLabFTW Homepage*. https://www.elabftw.net (accessed Oct. 24, 2022).

(19) Carpi, N.; Minges, A.; Piel, M. ELabFTW: An Open-Source Laboratory Notebook for Research Lab. *J. Open Source Softw* **2017**, *2*, 146.

(20) Carpi, N. *eLabFTW GitHub Page*. https://github.com/elabftw/elabftw (accessed Oct. 24, 2022).

(21) Schröder, M.; Staehlke, S.; Groth, P.; Nebe, J. B.; Spors, S.; Krüger, F. Structure-Based Knowledge Acquisition from Electronic Lab Notebooks for Research Data Provenance Documentation. *J. Biomed. Semant.* **2022**, *13*, 4.
(22) Smith, M.; Barton, M.; Bass, M.; Branschofsky, M.; McClellan, G.; Stuve, D.; Tansley, R.; Walker, J. H.DSpace: An Open Source Dynamic Digital Repositor*D-Lib Magazine* **2003**, *9*. DOI: 10.1045/january2003-smith
(23) Lyrasis. *DSpace Homepage*. https://dspace.lyrasis.org/ (accessed Oct. 24, 2022).
(24) Tremouilhac, P.; Nguyen, A.; Huang, Y.-C.; Kotov, S.; Lütjohann, D. S.; Hübsch, F.; Jung, N.; Bräse, S. Chemotion ELN: An Open Source Electronic Lab Notebook for Chemists in Academia. *J. Cheminformatics* **2017**, *9*, 54.
(25) Chemotion Developers. *Chemotion README* https://github.com/ComPlat/chemotion_ELN (accessed Nov. 18, 2022).
(26) The ELN Consortium. *ELN Consortium*. https://github.com/TheELNConsortium (accessed Nov. 18, 2022).
(27) Chemedata Initiative. *Chemedata Initiave: Goal and Scope*. https://chemedata.github.io/ (accessed Nov. 30, 2022).
(28) Jeannerat, D.; Trevorrow, P. Exploring CHEMeDATA. An Interview with Damien Jeannerat: What Is the CHEMeDATA Movement? *Anal. Sci. Adv.* **2020**, *1*, 254−257.
(29) Heimholtz-Zentrum hereon. *Aktuelle Projekte: I²B MgELB - Elektronisches Laborbuch*. https://www.hereon.de/institutes/metallic_biomaterials/powder_based_materials_development/projects/index.php.de (accessed Nov. 30, 2022).
(30) Bronger, T. *Introduction - JuliaBase, the samples database*. https://www.juliabase.org/ (accessed Nov. 30, 2022).
(31) Brandt, N.; Griem, L.; Herrmann, C.; Schoof, E.; Tosato, G.; Zhao, Y.; Zschumme, P.; Selzer, M. Kadi4Mat: A Research Data Infrastructure for Materials Science. *Data Sci. J.* **2021**, *20*.8
(32) Brinckmann, S. *PASTA-ELN | The favorite ELN for experimental scientists*. https://github.com/PASTA-ELN/pasta-eln (accessed Nov. 30, 2022).
(33) Rhiem, F. K.; Daniel Deckers, Malte; Mayer, Bjoern; Noffke, R.; Heuwes, Maximilian; Carpi, Nicolas; Holle, Nils. *sciapp/sampledb: Sample and Measurement Metadata Database*. https://github.com/sciapp/sampledb (accessed Nov. 30, 2022).
(34) Rhiem, F. SampleDB: A Sample and Measurement Metadata Database. *J. Open Source Softw.* **2021**, *6*, 2107.
(35) DSpace Developers. *DSpace Readme (GitHub)*. https://github.com/DSpace/DSpace (accessed Aug. 12, 2022).
(36) The Free Software Foundation. *GNU General Public License*. https://www.gnu.org/licenses/gpl-3.0.en.html (accessed Nov. 24, 2022).
(37) Richardson, L. *Beautiful Soup Documentation*. https://beautiful-soup-4.readthedocs.io/en/latest/ (accessed Oct. 24, 2022).
(38) Carpi, N. *elabapy 0.8.2*. https://pypi.org/project/elabapy/ (accessed Oct. 24, 2022).
(39) Carpi, N. *elabftw/elabapi-python: eLabFTW REST API v2 Python library*. https://github.com/elabftw/elabapi-python (accessed Jan. 1, 2023).
(40) McNamara, J. *XlsxWriter Homepage*. https://pypi.org/project/XlsxWriter/ (accessed Oct. 24, 2022).
(41) Canny, S. *python-docx 0.8.11 Documentation*. https://python-docx.readthedocs.io/en/latest/ (accessed Oct. 24, 2022).
(42) Kiehl, C. *Gooey GitHub Page*. https://github.com/chriskiehl/Gooey (accessed Nov. 24, 2022).
(43) Cortesi, D. B.; Giovanni; Caban, W.; McMillan, G *PyInstaller Homepage*. https://pyinstaller.org/en/stable/ (accessed Oct. 24, 2022).
(44) Richter, S. *lxml - XML and HTML with Python*. https://lxml.de/ (accessed Nov. 24, 2022).
(45) Sphinx Developers. *reStructuredText Primer*. https://www.sphinx-doc.org/en/master/usage/restructuredtext/basics.html (accessed Oct. 24, 2022).
(46) Zenodo.. https://zenodo.org/ (accessed Nov. 17, 2022).
(47) Dryad. *Dryad Home - publish and preserve your data*. https://datadryad.org/stash (accessed Nov. 17, 2022).
(48) Figshare. *Figshare*. https://figshare.com/ (accessed Nov. 17, 2022).
(49) Open Science Framework. *Open Science Framework*. https://osf.io/ (accessed Nov. 17, 2022).
(50) Berman, H.; Henrick, K.; Nakamura, H.; Markley, J. L. The Worldwide Protein Data Bank (WwPDB): Ensuring a Single, Uniform Archive of PDB Data. *Nucleic Acids Res.* **2007**, *35*, D301−3.
(51) The UniProt Consortium. UniProt: A Worldwide Hub of Protein Knowledge. *Nucleic Acids Res.* **2019**, *47*, D506−D515.
(52) Hubbard, T.; Barker, D.; Birney, E.; Cameron, G.; Chen, Y.; Clark, L.; Cox, T.; Cuff, J.; Curwen, V.; Down, T.; Durbin, R.; Eyras, E.; Gilbert, J.; Hammond, M.; Huminiecki, L.; Kasprzyk, A.; Lehvaslaiho, H.; Lijnzaad, P.; Melsopp, C.; Mongin, E.; Pettett, R.; Pocock, M.; Potter, S.; Rust, A.; Schmidt, E.; Searle, S.; Slater, G.; Smith, J.; Spooner, W.; Stabenau, A.; Stalker, J.; Stupka, E.; Ureta-Vidal, A.; Vastrik, I.; Clamp, M. The Ensembl Genome Database Project. *Nucleic Acids Res.* **2002**, *30*, 38−41.
(53) RCSB Protein Data Bank. *Structure Overview: Browse.*. https://www.rcsb.org/.
(54) EMBL-EBI. *Ensembl Help: Gene Ontology*. https://www.ensembl.org/Help/View?id=285 (accessed July 12, 2022).
(55) UniProt Consortium. *UniProt Help: Gene Ontology*. https://www.uniprot.org/help/gene_ontology (accessed July 12, 2022).
(56) Gene Ontology Consortium. *Gene Ontology Resource*. http://geneontology.org/ (accessed Nov. 17, 2022).
(57) Mühlhaus, T.; Brillhaus, D.; Tschöpe, M.; Maus, O.; Grüning, B.; Garth, C.; Rodrigues, C. M. DataPLANT−Tools and Services to Structure the Data Jungle for Fundamental Plant Researchers.
(58) NFDI4Plants Consortium. *Swate GitHub Repository*. https://github.com/nfdi4plants/Swate (accessed Nov. 18, 2022).
(59) NFDI4Chem Consortium. *NFDI4Chem. Terminology Service*. https://terminology.nfdi4chem.de/ts (accessed Nov. 18, 2022).
(60) NFDI4Chem Consortium. *NFDI4Chem. - Chemistry Consortium in the NFDI*. https://www.nfdi4chem.de/ (accessed Nov. 17, 2022).
(61) Kunis, S.; Hänsch, S.; Schmidt, C.; Wong, F.; Strambio-De-Castillia, C.; Weidtkamp-Peters, S. MDEmic: A Metadata Annotation Tool to Facilitate Management of FAIR Image Data in the Bioimaging Community. *Nat. Methods* **2021**, *18*, 1416−1417.
(62) Swain, M. C.; Cole, J. M. ChemDataExtractor: A Toolkit for Automated Extraction of Chemical Information from the Scientific Literature. *J. Chem. Inf. Model.* **2016**, *56*, 1894−1904.
(63) Zhu, M.; Cole, J. M. PDFDataExtractor: A Tool for Reading Scientific Text and Interpreting Metadata from the Typeset Literature in the Portable Document Format. *J. Chem. Inf. Model.* **2022**, *62*, 1633−1643.
(64) Helmholtz Metadata Collaboration. *Helmholtz Metadata Collaboration: Tools and Services*. https://helmholtz-metadaten.de/en/tools (accessed Sept. 12, 2022).
(65) Cyganiak, R.; Wood, D.; Lanthaler, M. *RDF 1.1 Concepts and Abstract Syntax*. https://www.w3.org/TR/rdf11-concepts/ (accessed Nov. 24, 2022).

https://doi.org/10.1021/acs.jcim.3c00744
J. Chem. Inf. Model. 2023, 63, 6224−6238