

RESEARCH ARTICLE OPEN ACCESS

PASTA-ELN: Simplifying Research Data Management for Experimental Materials Science

 S. Brinckmann¹  | G. Winkens¹ | R. Schwaiger^{1,2}
¹Forschungszentrum Jülich GmbH, Institute of Energy Materials and Devices, Structure and Function of Materials (IMD-1), Juelich, Germany | ²Chair of Energy Engineering Materials, Faculty 5, RWTH Aachen University, Aachen, Germany

Correspondence: S. Brinckmann (s.brinckmann@fz-juelich.de)

Received: 11 December 2025 | **Revised:** 23 February 2026 | **Accepted:** 25 February 2026

Keywords: electronic lab notebook | FAIR data | materials science | open source | research data management

ABSTRACT

Research data management remains a persistent challenge across scientific disciplines, as many electronic lab notebooks (ELNs) impose steep learning curves, rigid workflows, or network dependencies that hinder everyday use. We present *PASTA-ELN*, an open-source, cross-platform ELN designed from the scientist's perspective to simplify documentation while supporting the transition toward FAIR and machine-actionable data practices. *PASTA-ELN* employs a local-first-then-sync architecture, ensuring full offline functionality, user control over data, and seamless, optional synchronization with server-based ELNs such as *elabFTW*. Its hierarchical folder structure mirrors natural laboratory organization, while modular Python add-ons enable customizable data extraction, visualization, semantic annotation, and project reporting without altering the core system. All information is stored in open, standards-based formats facilitating long-term preservation and interoperability via RO-Crate and the `.eln` file profile. By combining intuitive desktop usability with robust research data management, *PASTA-ELN* provides an accessible, extensible framework that facilitates the transition from basic file organization to digital workflows with user-defined levels of FAIR compliance.

1 | Introduction

The management, documentation, and reproducibility of scientific data remain central challenges across modern research disciplines, including materials science and engineering (MSE). As experimental complexity increases and laboratories routinely generate heterogeneous datasets—from microscopy images and diffraction patterns to mechanical-testing curves and proprietary instrument formats—researchers are increasingly expected to maintain workflows that are transparent, well structured, and aligned with the FAIR principles (Findable, Accessible, Interoperable, and Reusable) [1]. Yet, despite the growing availability of digital research-data-management (RDM) tools, many laboratories still rely on ad hoc folder structures, instrument-specific conventions, or personal note-taking practices that impede long-term preservation, reproducibility, and cross-instrument data integration.

A wide range of open-source ELNs—such as *elabFTW*, *Chemotion*, *Kadi4Mat*, and *SampleDB*, OpenSemanticlab, and OpenBIS—

have advanced the structuring, preservation, and sharing of research data. However, ELNs [2, 3] often fail to attract users. Many systems remain difficult to adopt in everyday laboratory practice because they require substantial configuration, depend on stable network access, or impose rigid, server-centric workflows that do not align naturally with how experimentalists organize and handle data.

elabFTW [4], for example, provides a robust self-hosted platform for managing experiments and other resources, as well as file attachments, templates, and collaborative features including versioning, timestamps, and access control. *SampleDB* [5] provides a web-based environment for capturing and managing metadata for samples, measurements, and simulations, offering customizable schemas and integrations such as Jupyter notebooks. *OpenSemanticLab* (OSL) [6] emphasizes ontology-driven interoperability by combining structured and unstructured data through JSON-Schema and JSON-LD, enabling

This is an open access article under the terms of the [Creative Commons Attribution](https://creativecommons.org/licenses/by/4.0/) License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

© 2026 The Author(s). *Advanced Engineering Materials* published by Wiley-VCH GmbH.

machine-actionable laboratory workflows. *OpenBIS* [7] offers a scalable information-management system originally developed for the life sciences, integrating heterogeneous datasets and supporting automated pipelines. *Kadi4Mat* [8] extends ELN functionality with a full research data-management infrastructure tailored to materials science, including workflow tracking, visualization, and publication tools. *Chemotion* [9], designed for organic chemistry, provides domain-specific support for molecular structures and reactions and integrates tightly with a semantically enriched data repository.

While highly capable within their intended domains, these systems exhibit several practical limitations when deployed in active MSE laboratories: network dependencies that disrupt instrument-room usage; steep learning curves for novice users; workflows that demand rigid, predefined structures; and limited support for directly manipulating local files—the dominant paradigm in experimental research.

To address these challenges, we introduce *PASTA-ELN* (Portable, Agile, Simple, Transparent, and Adaptable Electronic Lab Notebook)—a desktop-first, local-first-then-sync ELN explicitly designed around the realities of experimental materials-science work. *PASTA-ELN* emphasizes *simplicity, flexibility, and scientific autonomy* rather than workflow description. Its core strengths can be summarized as follows:

- Local-first operation ensures reliable data capture and annotation directly at instruments.
- Transparent, hierarchical on-disk structure stays synced with ELN.
- Unified ingestion pipeline standardizes diverse measurement formats and metadata.
- Supports gradual adoption of FAIR principles, incl. semantic definitions and `.eln` export.
- Extensible Python add-ons for extraction, visualization, annotation, and reporting.

Collectively, these design principles lower the barrier to structured, digital research documentation, while supporting the demanding data heterogeneity typical of modern materials-science experimentation. By bridging intuitive, desktop-based workflows with robust RDM and FAIR-aligned practices, *PASTA-ELN* offers a practical, extensible framework that empowers MSE researchers

to transition naturally from basic file organization to fully machine-actionable scientific records. The system enables incremental adoption of FAIR principles, with the level of FAIR compliance determined by adherence to community-defined metadata standards (e.g., ontologies, vocabularies, schemas), the quality of user input, and the implementation of relevant add-ons.

The remainder of this article introduces the design principles and architecture of *PASTA-ELN*, highlighting how its local-first approach, transparent folder structure, and modular add-on ecosystem support everyday materials-science workflows. We then describe key implementation aspects, including the database structure, synchronization strategy, and interoperability with community standards such as RO-Crate and the `.eln` format. This is followed by an overview of the add-on framework and its role in data extraction, visualization, and project reporting. A representative materials-science use case illustrates how heterogeneous experimental data are ingested, annotated, and analyzed within *PASTA-ELN*. Finally, we discuss practical considerations, limitations, and design trade-offs.

PASTA-ELN addresses the persistent challenge of heterogeneous workflows and the need for immediate, local control of data and metadata in experimental laboratories, particularly in materials science. Unlike server-based ELNs, *PASTA-ELN* supports seamless offline operation and direct interaction with local file systems, lowering adoption barriers and enabling ad hoc, instrument-room workflows. It is particularly suited for individual researchers or small groups seeking a lightweight, extensible solution without the need for institutional IT support or complex server infrastructure. By integrating intuitive desktop usability with robust research data management and incremental FAIR compliance, *PASTA-ELN* provides a practical alternative for laboratories underserved by conventional web-based ELNs.

2 | System Design

2.1 | Intended User Workflow

We start by designing the user's workflow, as it directly informs the system architecture and the required add-ons. As illustrated in Figure 1, the user first creates a project and defines an overarching objective. The project is then organized into subtasks (subfolders), each of which can be annotated with comments

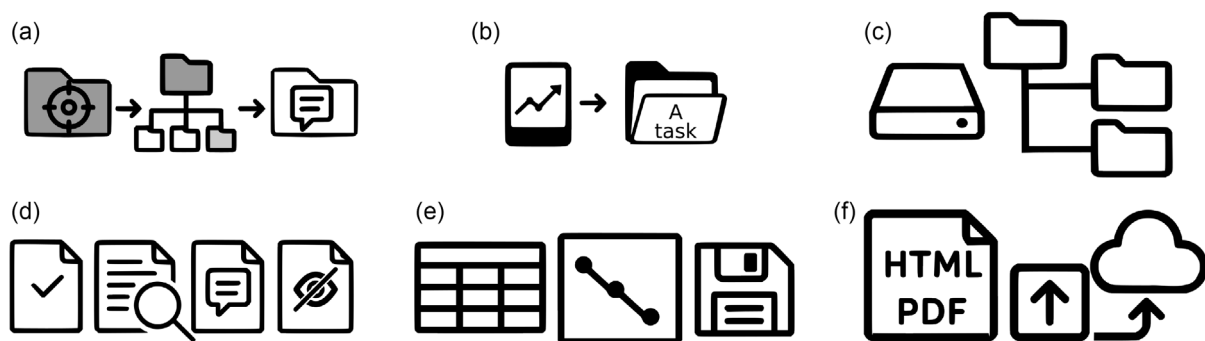


FIGURE 1 | By design, the user proceeds through the following steps: (a) creating a project with an objective and annotated tasks; (b) populating the project via drag-and-drop; or (c) allowing *PASTA-ELN* to automatically synchronize the in-app structure with the on-disk hierarchy; (d) inspecting, annotating, and optionally hiding data; (e) using analytical tools to identify correlations and save derived metadata; and (f) exporting the project to html-report or an `.eln` file for repository deposition.

or notes. Next, the user populates the project either by dragging and dropping measurement files into the project hierarchy or by selecting “Scan” to import newly added files from the on-disk data directory. All imported files appear in the hierarchical tree, allowing users to quickly locate experiments and their associated assets.

Data can be inspected using familiar tools, and users may annotate files, add comments, or hide data that is not directly relevant to the project’s progress. The built-in table view supports aggregating measurements, exploring correlations, and saving derived metadata or analysis results. Finally, users can export a project report (.html/.pdf) and, when needed, upload curated datasets to external repositories for sharing and long-term preservation.

2.2 | Architecture Design

Experimental materials-science workflows pose persistent, practical challenges for research data management: instruments often operate in network-limited environments, measurement files follow diverse formats and directory conventions, and metadata become fragmented across instrument PCs, notebooks, servers, and personal drives. *PASTA-ELN* was therefore designed around a set of pragmatic principles aimed at fitting naturally into everyday laboratory practice while enabling a gradual transition toward more structured, FAIR-aligned research data.

2.2.1 | Local-First Reliability With Optional Synchronization

In many MSE laboratories, network connectivity is inconsistent—especially at instrument workstations. *PASTA-ELN* therefore adopts a *local-first-then-sync* architecture, in which all data and metadata are stored and remain fully accessible on the user’s computer. This ensures uninterrupted functionality during experiments, avoids reliance on institutional infrastructure, and supports simple, transparent backup strategies. Synchronization with server-based ELNs such as *elabFTW* is available but entirely optional, enabling collaboration without enforcing a web-first workflow. Currently, synchronization entails verifying the latest version and propagating updates. In the event of conflicts, affected items are omitted and users are notified. During propagation, both data and metadata—including comments, tags, and associated data files—are synchronized. Furthermore, data for certain clients must be retained locally on a single desktop to ensure full compliance with contractual data-protection requirements.

2.2.2 | Transparent Data Organization Aligned With Laboratory Practice

PASTA-ELN employs a hierarchical on-disk folder structure that reflects the way experimentalists naturally organize projects, samples, and measurements. Every item shown in the ELN corresponds to a real file or directory on the user’s hard drive, minimizing the conceptual gap between physical work and digital documentation. Users can continue working with familiar analysis tools (e.g., Origin, ImageJ, and ParaView), and any edits are detected through a simple rescan action that keeps the ELN synchronized with the underlying file system.

2.2.3 | Uniform Treatment of Heterogeneous Measurement Data

MSE data originates from a wide and diverse range of instruments, many of which produce proprietary or multifile outputs. *PASTA-ELN* treats every measurement as a first-class entity and ingests files through modular extractor add-ons. These extractors generate thumbnails, parse embedded metadata, and harmonize information into a consistent schema—regardless of file format or device vendor. For instruments that produce multifile datasets, *PASTA-ELN* suggests bundling all components into a single archive to preserve integrity and support reproducible processing. This approach ensures that microscopy images, mechanical-testing curves, diffraction patterns, and spectroscopy files are handled in a unified and predictable manner.

2.2.4 | Incremental Adoption Toward Structured and Semantic Metadata

Recognizing that researchers differ widely in their RDM maturity, *PASTA-ELN* is designed to support gradual adoption. Users may begin with minimal annotations—simply dragging files into a project—and enrich metadata over time as needs evolve. Semantic descriptions can be added retrospectively through definition add-ons, which link properties to human-readable explanations and persistent identifiers (PURLs). This incremental approach allows scientists to work efficiently while progressively advancing toward more interoperable and machine-actionable metadata.

2.2.5 | Modularity and Extensibility Through Add-Ons

To keep the core software lightweight and focused, *PASTA-ELN* delegates specialized or domain-specific functionality to Python-based add-ons. Extractors handle file parsing, table add-ons support visualization and analysis, comment add-ons automate text generation, and project add-ons enable custom reporting. This modular architecture allows new instrument types, emerging workflows, and evolving domain conventions to be integrated without modifying the core system. It also encourages community-driven extensions tailored to specific MSE subfields.

2.2.6 | Data Integrity, Reproducibility, and Long-Term Accessibility

PASTA-ELN places strong emphasis on data provenance and long-term accessibility. File checksums detect modifications, measurement-level metadata is stored in an open relational format (SQLite), and all project contents can be exported using the RO-Crate-based `.eln` file format. This ensures self-contained, standards-based packages suitable for long-term preservation, repository deposition, and cross-platform reuse.

The system comprises four primary components—frontend, backend, database, and storage—each with a clearly defined role (Figure 2). The backend serves as the central orchestration layer. Running on its own thread, it coordinates all changes to the local database and the *project-groups*’s on-disk state. Its responsibilities include scanning project folders, invoking extractor add-ons to generate thumbnails and measurement metadata, creating and managing all supported item types, and performing consistency checks to detect and repair discrepancies between the database and the file system. The frontend, executed on a separate thread to maintain a responsive and stable graphical user interface,

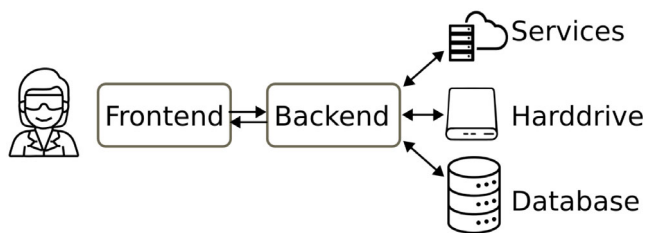


FIGURE 2 | *PASTA-ELN* architecture: The frontend interacts with users and communicates with the backend thread. The backend manages synchronization between the local database and the file system, ensuring consistency and integrity of stored records. It can also interface with external services—such as remote repositories and server-hosted ELNs—to exchange data and metadata.

guides user interaction. It presents dialogs and visualizations, issues commands to the backend, and updates the interface as backend tasks progress. The database is a local file stored within the *project-group*'s designated storage area. The entire storage hierarchy resides in user space, enabling straightforward permissions management, backup routines, and manual file inspection when needed. Communication between the frontend and backend is asynchronous and thread-safe, ensuring that long-running backend tasks do not block the UI and the allowing the system to handle errors and interruptions gracefully.

In summary, *PASTA-ELN* addresses three central challenges encountered by MSE experimentalists: (1) it operates reliably offline through a local-first architecture; (2) it provides a unified and intuitive structure for managing large volumes of heterogeneous data; and (3) it preserves metadata across instruments, modalities, and workflows through automated extraction and semantic annotation. Together, these capabilities transform fragmented laboratory records into coherent, FAIR-aligned datasets while respecting the established working practices of experimental scientists.

2.3 | Add-on Design

A central feature of *PASTA-ELN* is its support for add-ons that extend and customize the application. Because add-ons are written in Python—the de facto language of scientific computing with a low barrier to entry—researchers can easily create and tailor these small scripts to their workflows. By contrast, server-based ELNs require more advanced skills, including managing API access tokens and crafting API requests. Add-ons are intended to be developed by advanced users within the community and distributed either through conventional channels, such as email, or via the Matwerk Node Store, which will host Python functions for common workflows.

PASTA-ELN currently supports five types of add-ons: data extraction, table display, project view/edit, item generation, and definitions. This modular architecture facilitates add-on sharing and community-driven innovation without modifying the core *PASTA-ELN* application. Figure 3 shows a typical user workflow and the key points where add-ons integrate.

Extractors are responsible for ingesting measurement files into *PASTA-ELN*. Given a file location, an extractor parses the file, generates a representative thumbnail (default width: 600 px) for rapid visual triage, and captures both vendor-embedded metadata and project-specific metadata required for the study. Extractors also apply user-defined visualization preferences (e.g., plot styles, tick placement, and font sizes), ensuring reproducible, team-wide visualization standards. Multiple visualization ‘recipes’ allow researchers to render alternative views of the same measurement data, such as (1) time series annotated with identified peaks and troughs, (2) accumulated histograms, and (3) 2D/3D projections. Extractors can additionally orchestrate data transformation workflows. For example, one extractor may convert proprietary binary data into an open format (HDF5), which a second extractor can then read for downstream processing and visualization.

The **table add-ons** enable visualization of tabular datasets within *PASTA-ELN*. Core operations such as filtering and sorting are handled by the platform, allowing these add-ons to focus on plotting. By default, they support visualizing table metadata (e.g., line plots and histograms) and rendering individual measurement traces. The framework is extensible to 2D and 3D visualizations and can be enhanced with analytical capabilities such as peak detection and outlier identification.

The **comments add-on** enables programmatic generation and modification of comment content through integration with external tools and services. It is designed to support future extensions leveraging large language models (LLMs). In addition, the add-on can update other properties of associated items, including tags and metadata.

The **definitions add-on** automatically generates semantically grounded definitions for terms using advanced techniques such as unsupervised term clustering followed by cluster-level semantic synthesis. By default, it retrieves corresponding definitions from Wikidata.

The **project add-on** allows for custom project reporting. The default add-on generates an html project report, which can be further exported as pdf-file. Future extensions may support additional outputs, such as publication drafts generated via LLMs. Project add-ons are highly flexible and can support other functionalities as well. A standard example is the bulk import add-on,

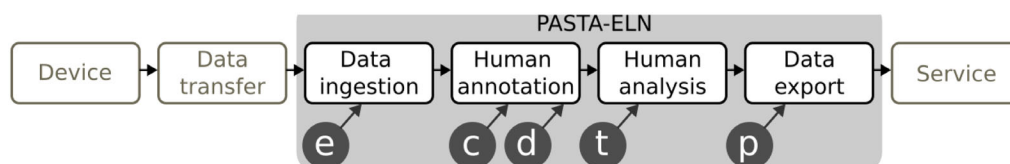


FIGURE 3 | Typical workflow of scientific data. Data generated by an instrument is transferred via network or storage device to *PASTA-ELN*, where it is ingested from the file system, followed by human annotation and analysis. Finally, the data can be exported to external services, such as repositories or other ELNs, for collaboration. Add-ons are highlighted in dark gray and represent the @extractors, @comment add-ons, @definition add-ons, @table add-ons, and @project add-ons.

which enables importing large numbers of samples from a csv file containing their metadata. Unlike other add-ons, this type has no safeguards and can modify all items within a project. The recommended safeguard against faulty project add-ons is adherence to standard IT practices, specifically regular backups of all user data, including both scientific data and the metadata database.

3 | Implementation Details

PASTA-ELN is a cross-platform application that runs natively on Windows, Linux, and macOS. Figure 4 shows the tree view within *PASTA-ELN* alongside the system prompt in Windows 10. The codebase is implemented entirely in Python using PySide6 (Qt for Python), providing a single-language stack for both application logic and UI while enabling a unified graphical style through Qt’s styling system and add-ons. As an open-source project under the MIT license, *PASTA-ELN* is highly adaptable, encouraging customization, extension, and integration into diverse research workflows.

PASTA-ELN’s storage layer evolved from a document-oriented backend (Apache CouchDB) to a file-based relational backend (SQLite). While CouchDB’s document model offered fully flexible data structures, its Windows installation could not be reliably automated, and frequent changes to its installers created a significant maintenance burden. Consequently, we adopted SQLite, a widely used, highly optimized, zero-configuration database embedded in a single file, which simplifies deployment, backup, and portability. The item types are stored in the *docType* table, with their per-type schemas stored in *docTypeSchema*. All items are recorded in the *main* table with persistent unique identifiers, names, comments, creation and modification timestamps, and associated metadata. Potentially multiple on-disk locations are tracked in the *branches* table. All item properties are stored in the *properties* table, while their semantic descriptions are maintained in the *definitions* table. Separate tables exist for *tags*, *QR codes*, and *database-changes*.

All data for a given *project-group* is stored locally within a single, isolated structure—a real folder on the computer’s hard disk, as illustrated in Figure 4—allowing multiple *project-groups* to coexist without cross-contamination. Keeping the database local to each *project-group* prevents accidental sharing and reduces the risk of data leaks when working with competing clients. This architecture prioritizes control, straightforward backups, and clear *project-group* boundaries, while still allowing coordinated syncing when appropriate or when a network connection is available.

The default data model is domain-agnostic and suitable for a wide range of research fields, not limited to MSE. It defines multiple item types and organizes content hierarchically: projects serve as the primary organizational units within a *project-group*, with each project recording an objective, an activity status, and metadata that are common to all item types (e.g., name, comments, and tags). Projects contain folders (also an item type) that establish the hierarchical structure, with leaf nodes representing samples, devices, workflows, and measurements. Samples may carry additional attributes such as QR-codes, initial geometry, or color, while the measurement item type uses a substantially different schema compared to other item types. User-facing labels for these item types can be easily customized without changing the underlying data structure; for example, a sample can be renamed as a specimen, and a folder can be renamed as a task.

Each measurement in *PASTA-ELN* is typically associated with a single data file from an instrument. When an instrument vendor distributes a dataset across multiple files, the measurement instead references a compressed archive (e.g., ZIP) that preserves the original directory structure and all auxiliary files. This archive serves as the canonical source from which extractor add-ons parse and populate standardized metadata and properties, as described in Section 2.3. This one-artifact-per-measurement convention streamlines ingestion, validation, and reproducible reuse. Furthermore, identical files with the same SHA checksum are logically deduplicated and share metadata, so features annotated or identified in one copy are automatically visible across all identical copies within the *project-group*.

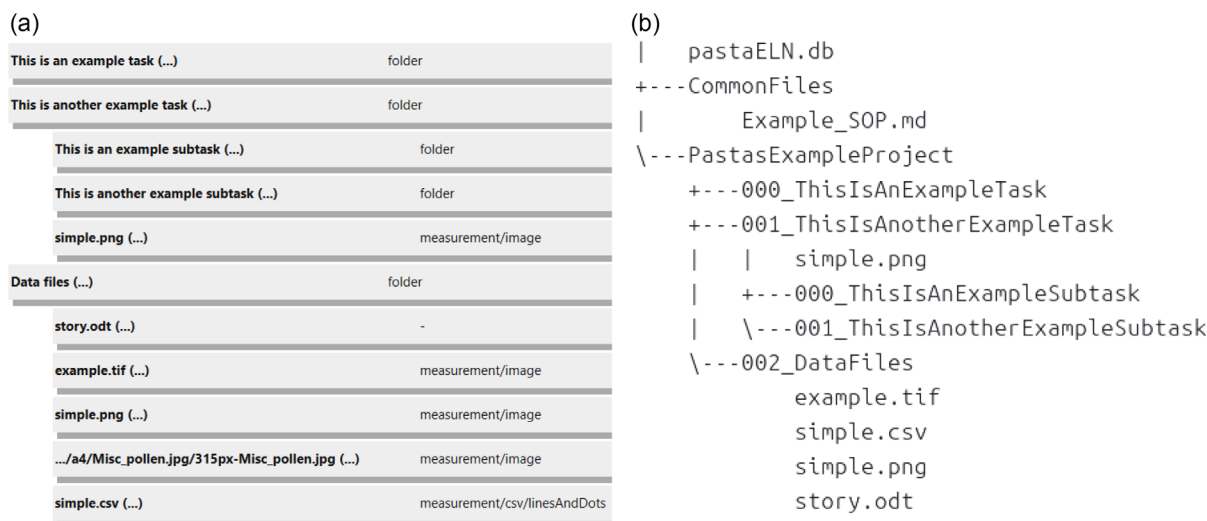


FIGURE 4 | Two views of the same example project on Windows 10. (a) The project view within *PASTA-ELN*, showing folders and items. (b) The corresponding folder structure on disk, generated with the “tree/f/a” command (hidden files omitted). The database—containing the metadata of the entire *project-group*—is listed at the top.

By default, a workflow comprises three item types:

1. **Workflow plan:** A priori description of the intended measurements and experimental sequence. The plan may differ from actual execution due to changes in instrument availability, experimental constraints, or updated preliminary expectations.
2. **Workflow log:** A sequential record of the events and actions carried out during execution, including start and end times for each process step and links to associated measurement files where applicable. A cryptographic file checksum (e.g., SHA-256) is recorded for each file to ensure data integrity and detect subsequent modifications.
3. **Procedure/Standard Operating Procedure (SOP):** A detailed, reproducible description of instrument configuration and operational instructions, presented in an enumerated format. The SOP specifies all relevant settings, consumables, safety precautions, and acceptance criteria required to reproduce the measurement.

In *PASTA-ELN*, all scientific metadata is stored in the *properties* table as a tuple of persistent identifier, key, value, and unit; for example, item *12345* may have an *image-width* of 2 with unit *mm*. Semantic definitions for all properties and classes are maintained in a central table keyed by the property, containing a human-readable description (optionally localized) and a PURL to the formal semantic definition. This separation of per-item facts from shared semantics ensures consistency, multilingual usability, and durable connections to external vocabularies and ontologies.

PASTA-ELN also supports **synchronization** via a central *elabFTW* server [4]: scientists can upload all data and metadata using the built-in sync, and other users can retrieve that content to annotate and reuse it. Users may also edit the main body in *elabFTW* and add tags directly. However, the two systems differ in design: *elabFTW* is a multiuser, server-based ELN without direct access to local file systems, whereas *PASTA-ELN* is single-user desktop software that maintains explicit links to files on the user's disk. *elabFTW* allows multiple files to be attached to an experiment, while *PASTA-ELN* extracts metadata from single files. Consequently, feature parity is not possible: only a subset

of properties uploaded from *PASTA-ELN* can be displayed in *elabFTW*, and only certain types of changes made in *elabFTW* are synchronized back to *PASTA-ELN*, ensuring consistency across both systems.

PASTA-ELN can **export and import** datasets using the RO-Crate-based *.eln* profile [10]. RO-Crate combines a file hierarchy with a JSON-LD manifest, allowing research data and metadata to be fully and explicitly defined in a semantic framework [11, 12]. Following schema.org principles [13], RO-Crate is both well-defined and flexible: entities and properties can be provided when available but are not strictly required. The *.eln* profile adds targeted constraints on top of RO-Crate to reduce variability and facilitate reliable import. Importantly, this profile represents a cross-vendor compromise among ELNs with different designs (e.g., flat versus hierarchical structures), so complete feature parity is not achievable; some elements, such as color profiles, cannot be transferred between systems, although core metadata and content remain fully interoperable.

The *.eln* file bundles all data and metadata associated with a research project. By building on the standard RO-Crate structure (a file hierarchy paired with a JSON-LD manifest) it is readily **shareable via repositories** such as Zenodo [14] and Dataverse [15] for reuse by other scientists. *PASTA-ELN* supports direct uploads to Zenodo and Dataverse and allows users to select which item types to include, enabling selective sharing while preserving the contextual metadata necessary for reproducibility and interoperability.

4 | Application and Discussion

In a previous study [16], we applied *PASTA-ELN* to nanoindentation experiments, demonstrating automated metadata extraction, visualization, and collaborative sharing via the *.eln* export. That work served as an initial example of how *PASTA-ELN* supports high-resolution imaging and micromechanical testing workflows. In this study, we turn to a different application domain to further illustrate the flexibility of the system. Specifically, we consider synthetic force–displacement data representative of tensile tests on additively manufactured metals. Starting from a single baseline experiment, two sets of 10 curves

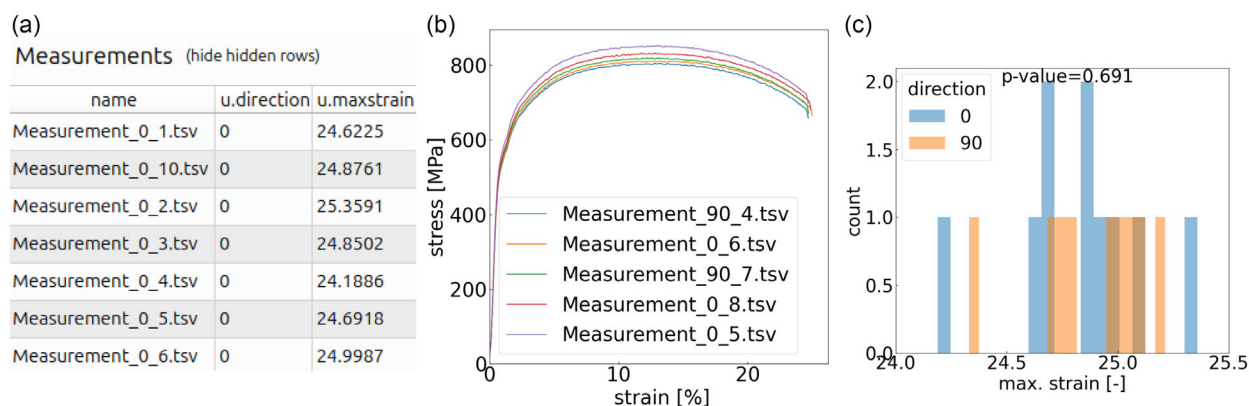


FIGURE 5 | Use case: characterization of additively manufactured metals. (a) Interactive tabular view in the core of *PASTA-ELN*, enabling filtering, sorting, and selection of entries. (b) Per-specimen tensile-test records visualized using the default table add-on. (c) Histogram of user-selected metadata from (a), color-coded by printing direction, with the corresponding *p*-value indicating whether the two distributions differ significantly. Panels (b) and (c) are generated automatically from the tabular data, requiring no coding.

were produced by independently applying random scaling factors to force and displacement. These data were then ingested into *PASTA-ELN*, and the tabular representation of the associated metadata was examined (Figure 5a). Subsequently, the two default table add-ons were used to (b) visualize all force–displacement curves and (c) plot the corresponding hardening slopes, with the resulting figures are shown in Figure 5.

PASTA-ELN usage illustrates a user’s progression into the digital realm: users typically begin with basic data ingestion, organization, and exploration, and gradually adopt more advanced techniques such as semantic annotation and the use of LLMs to auto-generate and enrich metadata, resulting in better searchable and interoperable records. Tables 1 and 2 show that *PASTA-ELN* is the only local-first, single-user system, offering very easy setup and extensibility through lightweight Python add-on. By contrast, other ELNs are web-first, multiuser platforms that require more complex server deployments. Regarding metadata models, *PASTA-ELN* uses a flexible, folder-aligned structure, whereas systems such as openBIS, Kadi4Mat, and SampleDB rely on stricter schema-driven or domain-specific metadata, and eLabFTW provides customizable but template-based records.

Automating extractors and providing fast preview plotting greatly reduces time and effort by eliminating repetitive manual processing. This approach enables immediate visual feedback on data quality and trends, accelerates iterative analysis, and allows users to focus on interpretation and higher-value tasks.

A clear, well-structured project view is essential: users must be able to hide entries entirely to reduce clutter and collapse items to headline-only height to maintain focus amid the volume of data typical in modern MSE. In version 3.2, the view is constrained by screen size—often displaying only about six items at

once—which limits situational awareness. Continued UI improvements, such as denser layouts, smarter grouping, and adaptive summaries, are needed to enhance the overall overview.

The concept of five raw item types, deliberately free of subtypes for different measurement modalities, has proven sufficient and effective. Measurement definitions remain general, while importing files with different extensions allows scientists to filter and organize datasets efficiently, supporting practical differentiation during analysis.

Although *PASTA-ELN* supports comprehensive semantic annotations across the framework, presenting these annotations in the GUI without introducing clutter remains challenging. In version 3.2, semantic metadata is therefore visible only in the item view and during editing.

Recurring user requests and feature ideas create a tension between expanding functionality and maintaining a clear, uncluttered GUI. *PASTA-ELN* adopts a minimalist approach by hiding most information in local menus. However, this can shift clutter from the main view into an increasingly crowded set of menu items, requiring careful design and prioritization to balance discoverability with simplicity.

PASTA-ELN also faces a trade-off between simplicity and flexibility. In its default configuration, all folders are treated as the same item type—mirroring conventional operating system folders—to keep the interface approachable for newcomers. Some users, however, prefer typed folders (e.g., for samples or devices) to better model domain concepts. While the system can accommodate this for advanced users, exposing such complexity risks overwhelming beginners with intricate configurations.

The two largest open design questions concern file semantics and data retention. First, *PASTA-ELN* currently uses a single-copy

TABLE 1 | Comparison of selected ELN/RDM systems relevant to MSE workflows: core operational features.

System	Local-first vs. Web-first	Single-user vs. Multiuser	Extensibility	Difficulty of initial setup
PASTA-ELN	Local-first with sync	Single-user (per workstation)	Add-ons for file parsing, metadata extraction, automation	Very low; runs locally without server configuration
eLabFTW	Web-first	Multiuser	Plugins, REST API, custom templates	Moderate; requires server or Docker deployment
openBIS	Web-first	Multiuser	Extensive API, custom entities, scripting hooks	High; multiservice server stack
Kadi4Mat	Web-first	Multiuser	Plugin system, workflow modules, APIs	High; full RDM platform with complex setup
SampleDB	Web-first	Multiuser	Import/export extensions and custom schemas	High; Python-based service setup

TABLE 2 | Comparison of selected ELN/RDM systems relevant to MSE workflows: metadata models.

System	Metadata model
PASTA-ELN	Flexible JSON-based metadata; hierarchical folder structure
eLabFTW	Free-form or structured experiment records with custom templates or machine-readable extra-fields
openBIS	Schema-driven, ontology-capable metadata system with typed entities
Kadi4Mat	Domain-specific MSE metadata with workflow and sample entities
SampleDB	Schema-driven metadata with configurable parameters

model in which file copies exist only once and share the same metadata and comments. This facilitates reuse of preannotated content, but some users expect each copy to behave independently with its own annotations. Second, although research data management guidelines generally discourage deletion, many beginners experiment with data structures and want to remove unsuccessful trials. *PASTA-ELN* therefore permits metadata removal while labeling hard disk content as trash, trading strict archival preservation for user flexibility during early-stage workflows.

Future work will focus on streamlining the core rather than adding built-in features. The add-on framework can provide specialized capabilities while keeping the core stable. This modular design enables advanced integrations—such as LLM-based tools—to connect to *PASTA-ELN* through defined interfaces without impacting core functionality.

5 | Conclusions

PASTA-ELN provides a practical, desktop-first approach to modern research data management, combining local usability with optional synchronization to external services. Its minimalist core-centered on five generic item types—lowers adoption barriers while enabling flexible organization through file-import conventions and Python-based add-on extensions. Automated extractors, rapid preview plotting, and semantic annotation accelerate routine workflows, improve discoverability, and enhance reusability. Data can be shared via the standardized `.eln` file format and synchronized with central *elabFTW* instances.

Design trade-offs were carefully considered. The decluttered GUI facilitates onboarding but can hide functionality in nested menus; tiered beginner/advanced modes and cleaner affordances are recommended. The single-copy file model encourages reuse of preannotated content but may conflict with expectations for independent file instances; configurable copy/instance options could address this. Allowing data removal supports novice experimentation but diverges from strict archival RDM guidelines; configurable retention and audit policies can reconcile flexibility with preservation.

By storing data and metadata in standard formats, recording explicit units, and linking properties to semantic descriptions, *PASTA-ELN* preserves contextual meaning and facilitates reuse. These features advance FAIR principles, making data findable (rich metadata), accessible (repository deposition), interoperable (`.eln` exchange), and reusable (clear semantics and complete context). Collectively, *PASTA-ELN* demonstrates that a local-first, extensible, and semantically aware desktop ELN can balance usability, flexibility, and rigorous research data management.

Author Contributions

Steffen Brinckmann: conceptualization (lead), software (lead). **Georg Winkens:** formal analysis (lead), methodology (lead), writing – review and editing (supporting). **Ruth Schwaiger:** conceptualization (supporting), supervision (lead), writing – review and editing (lead).

Acknowledgements

The authors would like to thank Dr. Hanna Tsybenko and Jithu Murugan for their support in developing the first versions of *PASTA-ELN*.

This work is part of the consortium NFDI-MatWerk, funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under the National Research Data Infrastructure—NFDI 38/1—project number 460247524.

Open Access funding enabled and organized by Projekt DEAL.

Funding

This work was supported by the Deutsche Forschungsgemeinschaft (460247524).

Data Availability Statement

Data sharing is not applicable to this article as no new data were created or analyzed in this study.

References

1. M. D. Wilkinson, M. Dumontier, I. J. J. Aalbersberg, et al., “The Fair Guiding Principles for Scientific Data Management and Stewardship,” *Scientific Data* 3 (2016): 160018.
2. S. Kanza, C. Willoughby, N. Gibbins, et al., “Electronic Lab Notebooks: Can They Replace Paper?,” *Journal of Cheminformatics* 9 (2017): 31.
3. S. G. Higgins, A. A. Nogiwa-Valdez, and M. M. Stevens, “Considerations for Implementing Electronic Laboratory Notebooks in an Academic Research Environment,” *Nature Protocols* 17 (2022): 179–189.
4. elabFTW, “A Free and Open Source Electronic Lab Notebook,” 2025, accessed November 10, 2025, <https://www.elabftw.net/>.
5. F. Rhiem, “Sampledb: A Sample and Measurement Metadata Database,” *Journal of Open Source Software* 6, no. 58 (2021): 2107, <https://doi.org/10.21105/joss.02107>.
6. P. Simon Stier, L. Gold, A. Räder, M. A. Popp, A. Triol, and L. Scheuplein, *The Opensemanticlab Platform*, (2023).
7. A. Bauch, I. Adamczyk, P. Buczek, et al., “openBIS: A Flexible Framework for Managing and Analyzing Complex Data in Biology Research,” *BMC Bioinformatics* 12 (2011): 468, <https://doi.org/10.1186/1471-2105-12-468>.
8. N. Brandt, L. Griem, C. Herrmann, et al., “Kadi4mat: A Research Data Infrastructure for Materials Science,” *Data Science Journal* (2021): 20, <https://doi.org/10.5334/dsj-2021-008>.
9. P. Tremouilhac, A. Nguyen, Y.-C. Huang, et al., “Chemotion Eln: An Open Source Electronic Lab Notebook for Chemists in Academia,” *Journal of Cheminformatics* (2017): 9, <https://doi.org/10.1186/s13321-017-0240-0>.
10. ELN file format (.eln). Eln file format (.eln), 2025, accessed November 10, 2025, <https://github.com/TheELNConsortium/TheELNFileFormat>.
11. S. Soiland-Reyes, P. Sefton, M. Crosas, et al., “Packaging Research Artefacts with Ro-Crate,” *Data Science* 5 (2022): 97–138.
12. P. Sefton, E. Ó. Carragáin, S. Soiland-Reyes, et al., *Ro-Crate Metadata specification 1.1.3* (2023).
13. Schemaorg Schemaorg, 2025, accessed November 11, 2025, Available at: <https://schema.org/>.
14. European Organization For Nuclear Research and OpenAIRE, (2013), <https://www.zenodo.org/>.
15. The Dataverse Project. Dataverse: Harvard dataverse repository, (2024), Version 5.x, <https://dataverse.harvard.edu>.
16. H. Tsybenko, S. Menon, F. Chen, et al., “Digital Transformation in Materials Science: A User Journey of Nanoindentation, Image Analysis and Simulations,” *Data Science Journal* 24 (2025).