DISCLAIMER

ORNL/TM-2025/4219

FY 2025 Multidimensional Data Correlation Platform: Unified Software Architecture for Advanced Materials and Manufacturing Technologies Program Data Management and Processing

Stephanie Cooper
Gina Accawi
Abby Barnes
Hayden Esser
Wes Forman
et al.

September 2025

M3CR-22OR0403053

OAK RIDGE
National Laboratory

ORNL/TM-2025/4219

Advanced Materials and Manufacturing Technologies Program

FY 2025 MULTIDIMENSIONAL DATA CORRELATION PLATFORM: UNIFIED SOFTWARE ARCHITECTURE FOR ADVANCED MATERIALS AND MANUFACTURING TECHNOLOGIES PROGRAM DATA MANAGEMENT AND PROCESSING

Stephanie Cooper
Gina Accawi
Abby Barnes
Hayden Esser
Wes Forman
Randall Stapleton
Annika Wille
Vincent Paquit

September 2025

M3CR-22OR0403053

Prepared by
OAK RIDGE NATIONAL LABORATORY
Oak Ridge, TN 37831
managed by
UT-BATTELLE LLC
for the
US DEPARTMENT OF ENERGY
under contract DE-AC05-00OR22725

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

AMMT	Advanced Materials and Manufacturing Technologies
API	application programming interface
CI/CD	continuous integration/continuous delivery
CUI	Controlled Unclassified Information
DOE	US Department of Energy
EDM	electrical discharge machining
FIB	focused ion beam
LDAP	lightweight directory access protocol
LPBF	laser powder bed fusion, may also be abbreviated L-PBF
MDDC	Multi-Dimensional Data Correlation
MDF	Manufacturing Demonstration Facility
ORNL	Oak Ridge National Laboratory
SEM	scanning electron microscopy
XCT	x-ray computed tomography

ABSTRACT

The Advanced Materials and Manufacturing Technologies (AMMT) program continues to advance a data-driven approach to demonstrate the utility of additive manufacturing for fabricating components for nuclear applications. A key scientific goal is to leverage data to better understand manufacturing outcomes and thereby improve the performance, reliability, and lifespan of nuclear components. Ultimately, this effort supports the development of standards for certification and qualification of additively manufactured components, enabling broader industry adoption.

In support of this objective, the AMMT program is building and deploying a data management platform to record, index, analyze, and make available the manufacturing data generated across the AMMT program. In FY 2023, the team conceptualized the architecture of the platform and, in FY 2024, deployed the first functional version at the Oak Ridge National Laboratory (ORNL) Manufacturing Demonstration Facility (MDF). In FY 2025, the platform was officially opened to all AMMT members. To enable this expansion, core modifications and enhancements were developed, including improvements to the user interface and workflows for data entry and retrieval. Most notably, robust security and access control mechanisms were implemented to protect data and manage information sharing. This effort featured a logging system, protected views, and controlled access mechanisms. This report documents these enhancements and the transition of the platform into program-wide use.

1. INTRODUCTION

The Multi-Dimensional Data Correlation (MDDC) platform is a data-driven initiative sponsored by the Advanced Materials and Manufacturing Technologies (AMMT) program to support the development of a novel approach for certification and qualification of additively manufactured components for nuclear applications. Its success depends on the creation and availability of comprehensive datasets that capture the full manufacturing life cycle and that can be leveraged across multiple areas of development. Achieving this effort requires coordinated contributions from AMMT program members to ensure the completeness and relevance of the data needed to advance this field.

Over the past 2 years, this work package has focused on building a custom data management platform to address this challenge. In FY 2023, the effort centered on conceptualizing a framework that preserves the relationships between manufacturing steps, feedstock materials, postprocessing, and testing and characterization and also enables the construction of digital twins of fabricated components. In FY 2024 the focus shifted to implementation and testing, resulting in a functional platform deployed at the Oak Ridge National Laboratory (ORNL) Manufacturing Demonstration Facility (MDF) and demonstrated on laser powder bed additive manufacturing systems.

In FY 2025, the platform, named Damara Tern, was opened to all AMMT program members. To enable this expansion, significant enhancements were made to the user interface, workflows for data entry and retrieval, and overall usability. Additionally, robust security and access control functionalities were implemented, including logging systems, protected views, and controlled access mechanisms, to safeguard data and manage information sharing.

This report is organized as follows: Section 2 provides an overview of the design and evolution of Damara Tern, integrating activities from FY 2023 and FY 2024. Section 3 describes the user interface. Section 4 outlines broader platform improvements across four subsections, beginning with user interface refinements—focusing on improved workflows for data entry, retrieval, content creation tools, and overall usability—and followed by access control mechanisms and deployment functionalities. Section 4 concludes with a subsection summarizing additional functionalities implemented through collaboration with another US Department of Energy program.

2. OVERVIEW OF THE DAMARA TERN DIGITAL PLATFORM DATABASE AND INTERFACE FRAMEWORK

This section provides a brief contextual summary of the platform. It has been adapted from the FY 2024 report [1] and updated where necessary to reflect recent changes.

Damara Tern was initially developed to access the MDF Digital Platform data. Its primary function is to structure and organize the metadata collected along the digital thread and to serve as a library, allowing users to save, explore, and retrieve the data associated with each operation and component (i.e., trackable) involved in the manufacturing process. The framework succeeds the MDF Digital Tool and related relational database initially deployed under the Transformational Challenge Reactor program for accessing and exploring the data stored within the MDF Digital Platform. Although its predecessor successfully serves this function and enables component tracking during the manufacturing and testing processes, it relies on a printer-centric model that induces specificity and poses significant limitations in terms of extensibility and automation. To overcome these limitations, the Damara Tern framework relies on a generic and flexible operation-centric structure that accommodates a diverse range of machines, manufacturing and testing operations, and trackable components. Additionally, the framework aims to enhance search and view functionalities that reflect the digital thread for each manufacturing process.

2.1 IMPLEMENTATION CONSIDERATIONS

The main objective of the MDDC framework is to provide a unified data management platform that is accessible to all members of the AMMT program. This framework should enable (1) nationwide tracking of physical components and digital assets across multiple sites and (2) the creation of a database adhering to the findable, accessible, interoperable, and reusable best practices to support the research and development activities of the program. In contrast to the implementation for the Transformational Challenge Reactor program [2], the platform described here requires using a data management strategy that preserves the integrity of the digital thread of each component across multiple physical locations. This important requirement resulted in the following considerations.

- **Database location:** Damara Tern was built on the principle that data should not reside in a single centralized location, nor should complete replicas be maintained at each national laboratory. Instead, each AMMT program participant is expected to host the data they generate using individual databases interconnected through the cross-referencing mechanism described in Section 2.2. Beyond technical distribution, this philosophy emphasizes that accessing and using information should be a collaborative effort in which participants share, cross-reference, and build upon one another's data to collectively accelerate progress. This approach was pioneering at the time of its design, and it currently aligns closely with the model being advanced by the current administration for the *American Science Cloud* [3].

- **Database access:** Initially, users will access the database using credentials provided by the host national lab. As new data exchange and linkage functionalities of the platform are developed, web tools will be provided that allow direct access to data regardless of geographic location.

- **Information retrieval:** Rarely, users will need access to the entire database to work on scientific problems. In this case, they will retrieve subsets of the data based on scoped queries, for which application programming interface (API) functionalities will be developed to make intelligent and focused queries to the database.

- **Knowledge extraction and data visualization:** The selected Django framework—a high-level Python web framework designed for building secure, scalable, and maintainable applications—provides access to numerous Python libraries for advanced data visualization and processing. Domain experts will be expected to create their own data processing pipelines to access the database and format the data for their specific applications.

2.2 OPERATION–TRACKABLE DATA MODEL

The data model employed to develop the framework revolves around the concepts of *operations* and *trackables*, which are defined as follows:

- **Operation**—Any action performed via the use of a machine (e.g., print, cure, heat treatment, blue light scan, microscopy analysis, tensile test) or human interaction (e.g., procurement, annotation, registration, manipulation). Operations can be performed before, during, or after the manufacturing process itself to extend the data context.

- **Trackable**—Any physical or digital component that can be subjected to an operation (e.g., parts, builds, feedstock materials).

This model relies on the manufacturing process flow and provides the necessary framework for storing and organizing metadata collected along the digital thread. The database is implemented along this model, in which each physical or digital component (i.e., trackable) undergoes a series of tests or actions (i.e., operations). Each operation supports the collection of metadata or substantial in situ data and can lead to the creation, alteration, combination, or transformation of the trackables.

Collecting the data at each operation level and preserving the traceability of relationships between the trackables and applied operations make possible recreating the digital thread for each trackable. This pathway reflects the entire manufacturing process and is accompanied by the comprehensive context of collected data and metadata gathered throughout the entire process.

Figure 1 gives an example of the operation–trackable-based model of the digital thread.

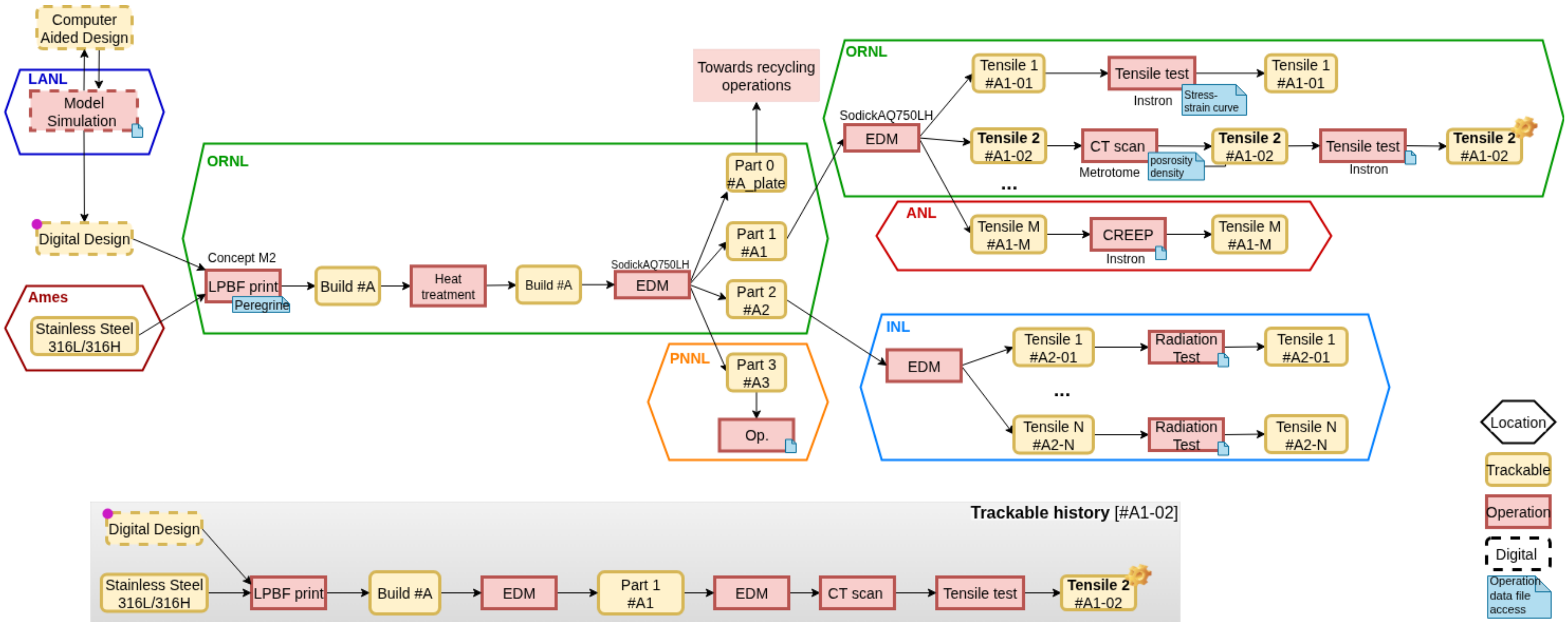


Figure 1. Representation of a multisite digital thread in the context of the operation–trackable data model.

In this example, the trackable specimen "Tensile 2" history of operations and parent trackables are highlighted. The trackable's entire creation context can be retrieved by retracing this history and gathering the data and metadata collected along each operation. EDM stands for electrical discharge machining, and LPBF stands for laser powder bed fusion.

2.3 DAMARA TERN FRAMEWORK: SOFTWARE AND TECHNOLOGY

The Damara Tern framework combines the underlying MDF metadata database and a web interface for exploring, entering, and accessing data. It is implemented using the Python Django web framework. Django websites rely on a database, use model files that describe the data, and provide automatically generated database-access APIs. To preserve the portability of the metadata collected for the Digital Platform, this project used distinct databases (or schemas) to store the Django-specific data and the data collected for the Digital Platform. This section presents an overview of the database structure used for the collected Digital Platform metadata and the Django project architecture.

2.3.1 Database Implementation

The original database for the Digital Platform was printer-centric, with each build recorded in a machine-specific PostgreSQL schema. Although this design allowed detailed tracking, it lacked extensibility and required manual setup for each new machine.

The current database design builds on the operation–trackable data model (Section 2.2) and follows a semistructured approach, using generic fields and JSON columns to accommodate equipment-, material-, and operation-specific data. PostgreSQL, originally selected during the proof-of-concept phase to instantiate the database, was retained for its performance and strong JSON functionality.

Figure 2 illustrates a high-level abstraction of the database architecture. The database tables can be grouped into five logical blocks:

- **Operation**—Records operations, their types, and the contexts in which they occur.
- **Trackable**—Catalogues physical or digital components subject to operations, along with their classifications.
- **Machine**—Represents printers and other equipment used in operations, including calibration and configuration details.
- **Related Data—**References supporting process data (e.g., images, videos, documents) stored locally or remotely.
- **People and Affiliation**—Contains user, institution, and project information required for access control and attribution.

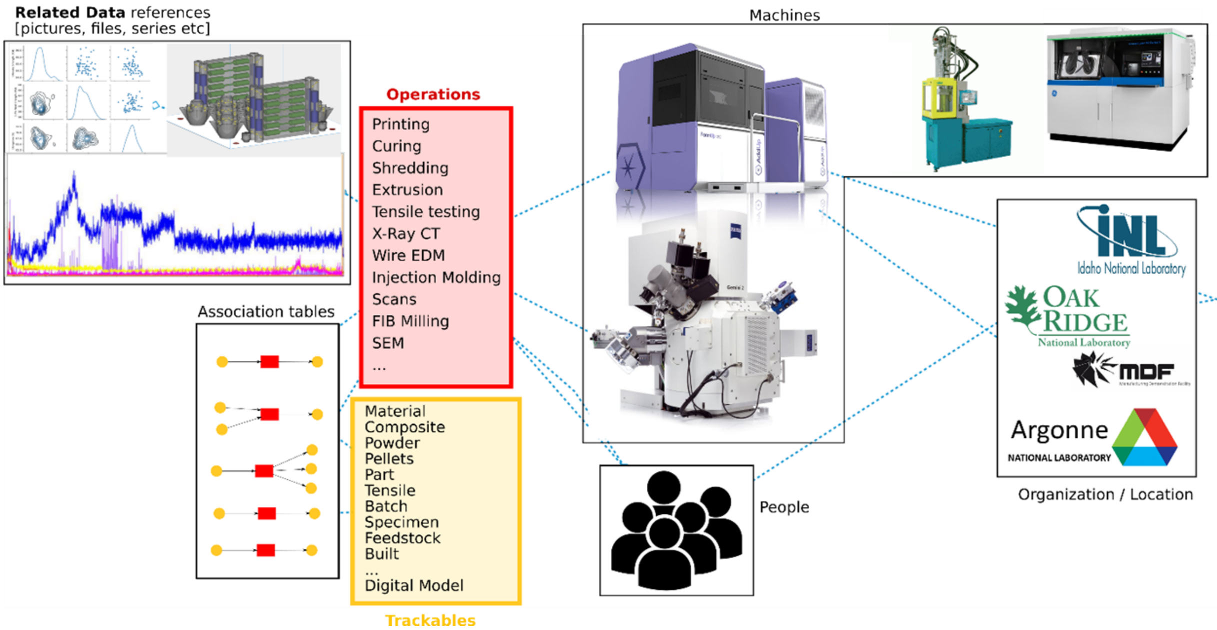


Figure 2. Simplified overview of the database relational structure.

The database is organized into five logical blocks: Operation, Trackable, Machine, Related Data, and People and Affiliations. Machines and systems present at currently indexed AMMT program member sites are associated with specific operations. Each operation contains information about the process, including metadata, interactions with individuals, and the data generated. Operations act as both consumers and producers of trackables, and their relationships are preserved through association tables recorded in the database. EDM stands for electrical discharge machining, FIB stands for focused ion beam, and SEM stands for scanning electron microscopy.

2.3.2 Django Architecture

Django was chosen to implement the Damara Tern user interface because of its versatility, off the shelf features, Python foundation, and proven success in large-scale projects (e.g., Instagram, Mozilla, National Geographic). The platform is built on a model–view–controller pattern and is organized into modular apps; it is thoughtfully designed to partition the project by core functionalities. The Damara Tern architecture, illustrated in Figure 3, relies on five core apps:

- **mdf_db**—Interfaces with the metadata database (Section 2.3.1) and provides the primary data structure model, including table definitions, validation rules, and admin interface support.
- **operation**—Manages creation, display, and search of operation records and their associated data.
- **trackable**—Manages creation, display, and search of trackable entries.
- **home**—Handles the main user interface, feedback form, and landing pages.
- **api**—Exposes the platform's data and functionality via RESTful end points for programmatic access.

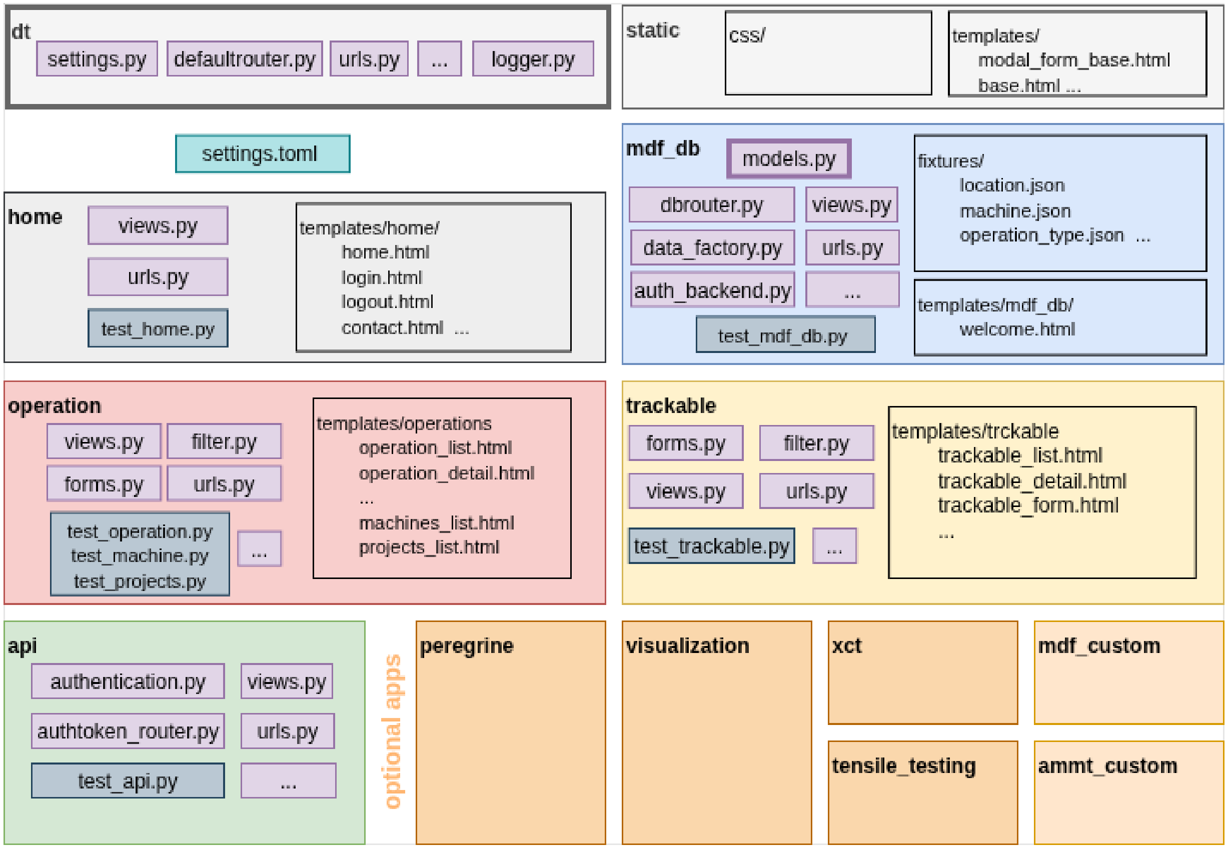


Figure 3. Overview of the updated Damara Tern Django app-based architecture. Optional modules are shown in orange.

Additional optional modules have been developed to enrich the platform with features and customizations, including the following. These optional apps provide various features and can be enabled or disabled as needed to fit the requirements of each deployed platform.

Feature apps:

- **peregrine**—Provides API and functionality for integration with the Peregrine software tool.
- **visualization**—Adds visualization capabilities, including embedded CSV tab displays.
- **tensile_testing**—Contains tools for plotting, visualizing, and comparing format-specific tensile test results.
- **xct**—Supports the MDF x-ray computed tomography (XCT) pipeline, offering interface trigger/monitor components and backend automation for the creation and ingestion of XCT operations and data.
- **llm**—Offers an experimental module introducing language model functionality within the platform.

Customization apps:

- **mdf_custom**—Site-specific customizations for internal ORNL deployments, including additional privileges, multilocation data storage configuration, prerequisite data checks, and prefilled values.
- **ammt_custom**—Tailored features for AMMT program requirements, including custom messages, default permission settings, and XCAMS user management.

3. OVERVIEW OF THE DAMARA TERN FRAMEWORK INTERFACE

Damara Tern is the name of the new framework for accessing the MDF Digital Platform data hosted at the MDF (main page displayed in Figure 4). Following the concept described in Section 2.2, the development team created a web-based platform for submitting and retrieving data generated during the AMMT program. This section describes major types of pages in the platform.

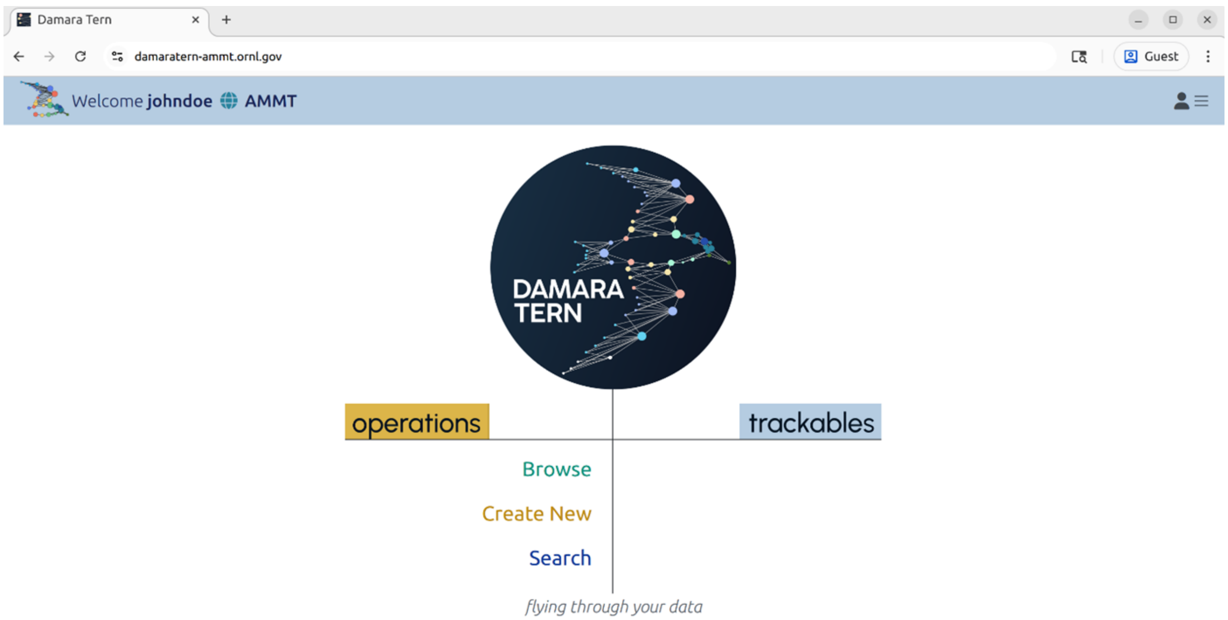


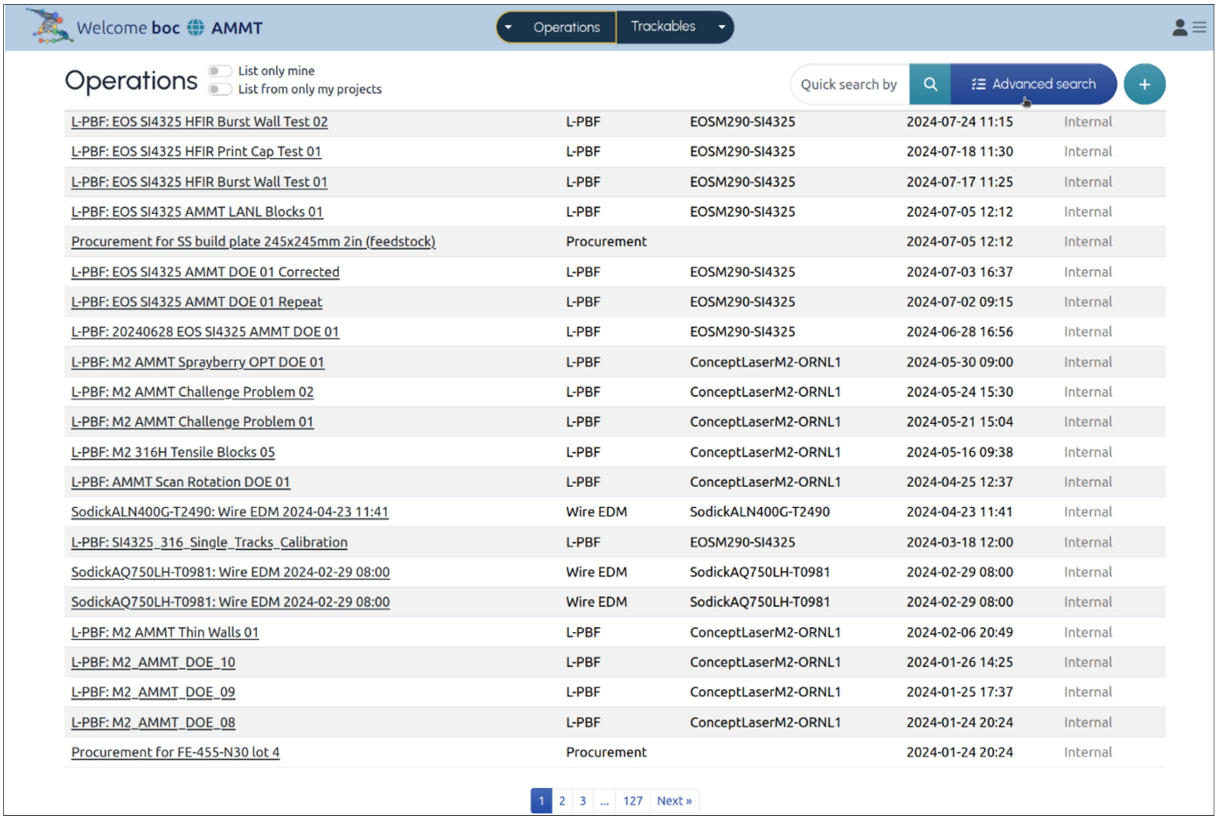
Figure 4. Main page of the Damara Tern digital platform.

The general layout of the platform includes three main views for both operations and trackables: list views, detail views, and creation forms presented in the following subsections.

3.1 LIST VIEWS

List views, also called list pages, provide an overview of all the operations or trackables records that a user has permission to access. These pages function as tables, displaying a subset of metadata for each object. They include search functionality, allowing users to filter and query objects by relevant attributes, such as the creation or procurement time of a trackable or the operators involved in an operation. Searches can be performed using either a quick search or an advanced form accessible via the search field or button at the top-right corner of the list pages or through the Operations/Trackables navigation bar at the top of the page. The available search fields differ between operations and trackables, reflecting the distinct metadata collected for each object type.

An example of the Operations list view is shown Figure 5, and Figure 6 illustrates the search form used to narrow down the results.



L-PBF: EOS SI4325 HFIR Burst Wall Test 02	L-PBF	EOSM290-SI4325	2024-07-24 11:15	Internal
L-PBF: EOS SI4325 HFIR Print Cap Test 01	L-PBF	EOSM290-SI4325	2024-07-18 11:30	Internal
L-PBF: EOS SI4325 HFIR Burst Wall Test 01	L-PBF	EOSM290-SI4325	2024-07-17 11:25	Internal
L-PBF: EOS SI4325 AMMT LANL Blocks 01	L-PBF	EOSM290-SI4325	2024-07-05 12:12	Internal
Procurement for SS build plate 245x245mm 2in (feedstock)	Procurement		2024-07-05 12:12	Internal
L-PBF: EOS SI4325 AMMT DOE 01 Corrected	L-PBF	EOSM290-SI4325	2024-07-03 16:37	Internal
L-PBF: EOS SI4325 AMMT DOE 01 Repeat	L-PBF	EOSM290-SI4325	2024-07-02 09:15	Internal
L-PBF: 20240628 EOS SI4325 AMMT DOE 01	L-PBF	EOSM290-SI4325	2024-06-28 16:56	Internal
L-PBF: M2 AMMT Sprayberry OPT DOE 01	L-PBF	ConceptLaserM2-ORNL1	2024-05-30 09:00	Internal
L-PBF: M2 AMMT Challenge Problem 02	L-PBF	ConceptLaserM2-ORNL1	2024-05-24 15:30	Internal
L-PBF: M2 AMMT Challenge Problem 01	L-PBF	ConceptLaserM2-ORNL1	2024-05-21 15:04	Internal
L-PBF: M2 316H Tensile Blocks 05	L-PBF	ConceptLaserM2-ORNL1	2024-05-16 09:38	Internal
L-PBF: AMMT Scan Rotation DOE 01	L-PBF	ConceptLaserM2-ORNL1	2024-04-25 12:37	Internal
SodickALN400G-T2490: Wire EDM 2024-04-23 11:41	Wire EDM	SodickALN400G-T2490	2024-04-23 11:41	Internal
L-PBF: SI4325_316_Single_Tracks_Calibration	L-PBF	EOSM290-SI4325	2024-03-18 12:00	Internal
SodickAQ750LH-T0981: Wire EDM 2024-02-29 08:00	Wire EDM	SodickAQ750LH-T0981	2024-02-29 08:00	Internal
SodickAQ750LH-T0981: Wire EDM 2024-02-29 08:00	Wire EDM	SodickAQ750LH-T0981	2024-02-29 08:00	Internal
L-PBF: M2 AMMT Thin Walls 01	L-PBF	ConceptLaserM2-ORNL1	2024-02-06 20:49	Internal
L-PBF: M2_AMMT_DOE_10	L-PBF	ConceptLaserM2-ORNL1	2024-01-26 14:25	Internal
L-PBF: M2_AMMT_DOE_09	L-PBF	ConceptLaserM2-ORNL1	2024-01-25 17:37	Internal
L-PBF: M2_AMMT_DOE_08	L-PBF	ConceptLaserM2-ORNL1	2024-01-24 20:24	Internal
Procurement for FE-455-N30 lot 4	Procurement		2024-01-24 20:24	Internal

Figure 5. Operations list view.

The page displays a searchable table of existing operation records. Quick and advanced search options are available in the upper-right corner. Contributor-level users can create new operations using the + button.

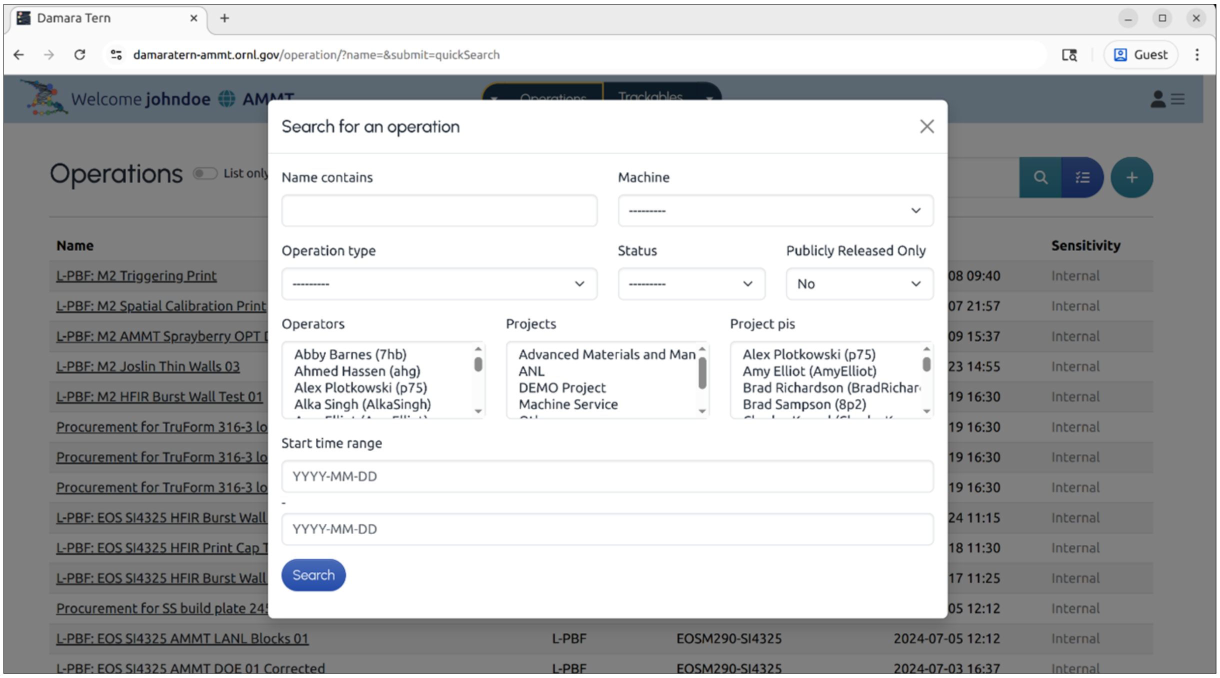


Figure 6. Advanced operation search form opened as a modal from the Trackables list view.

3.2 DETAIL VIEWS

Detail views provide in-depth information about a specific operation or trackable. They are accessed from the list view by selecting the desired item.

3.2.1 Operation Detail View

Operation detail pages (Figure 7) display all metadata associated with a given operation, including required fields, machine- or operation-specific metadata, and the list of linked input and output trackables (used or produced by the operation). Related documents, images, or data files stored on external storage can also be viewed or downloaded directly from the page.

Key sections of an operation detail page include the following:

- **Operation information**—General and type- or machine-specific metadata fields, including notes describing the operation.
- **Input and output trackables**—Lists of the trackables used or produced by the operation, with links to their detail pages.
- **Related data**—List of and access to associated files, featuring highlighted content, browsing, preview, and download capabilities.
- **Actions and editing controls**—Available to authorized users, enabling metadata editing, updating the list of associated trackables, and adding or uploading new content.

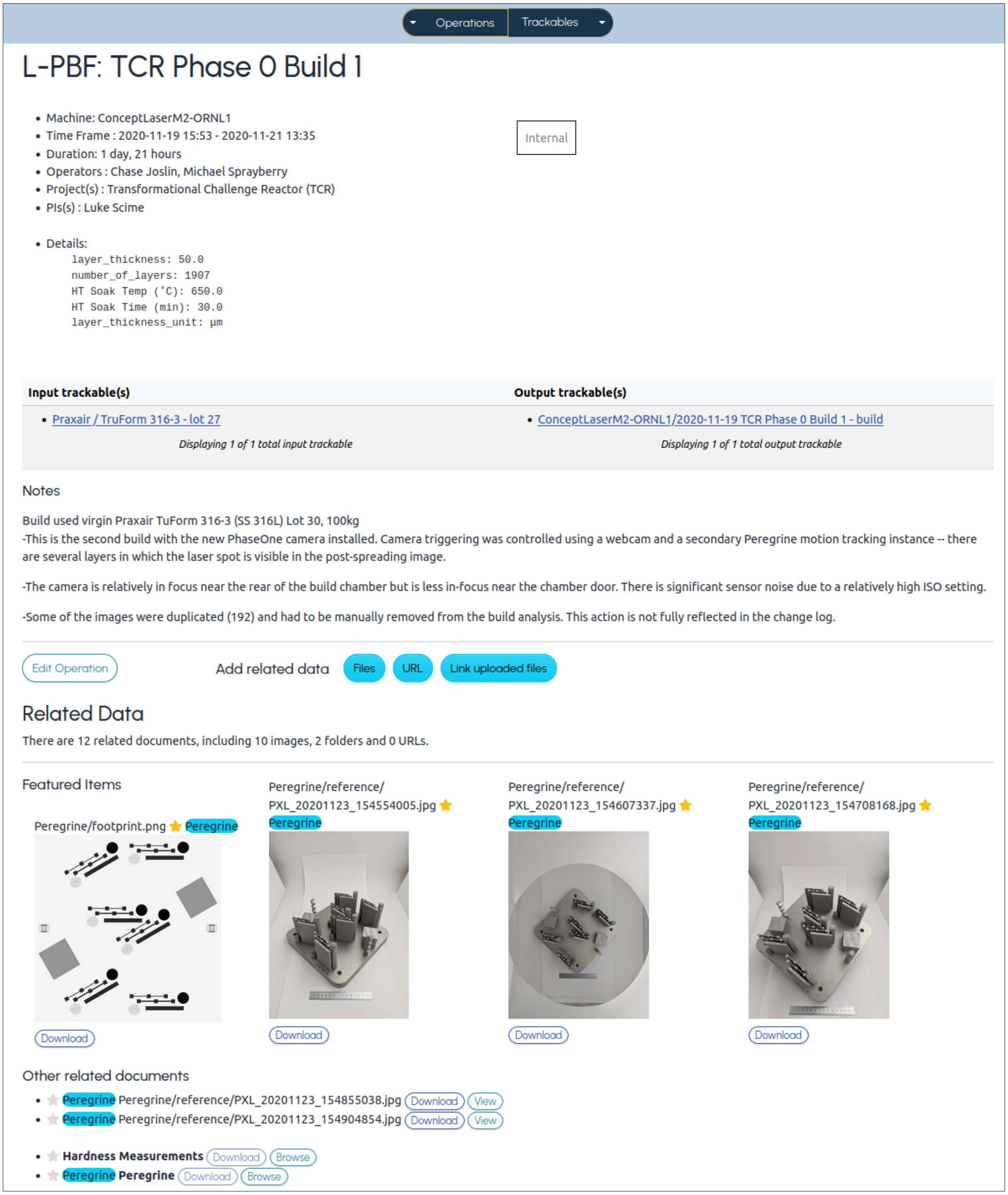


Figure 7. Example of operation detail view on Damara Tern platform.

3.2.2 Trackable Detail View

Trackable detail pages display metadata associated with a given trackable, such as ID, QR code, sensitivity, and originating operation (procurement or creation). They also present an interactive list of subsequent operations the trackable has undergone, each with expandable detail panels, followed by any parent trackables that contributed to its creation (see Figure 8).

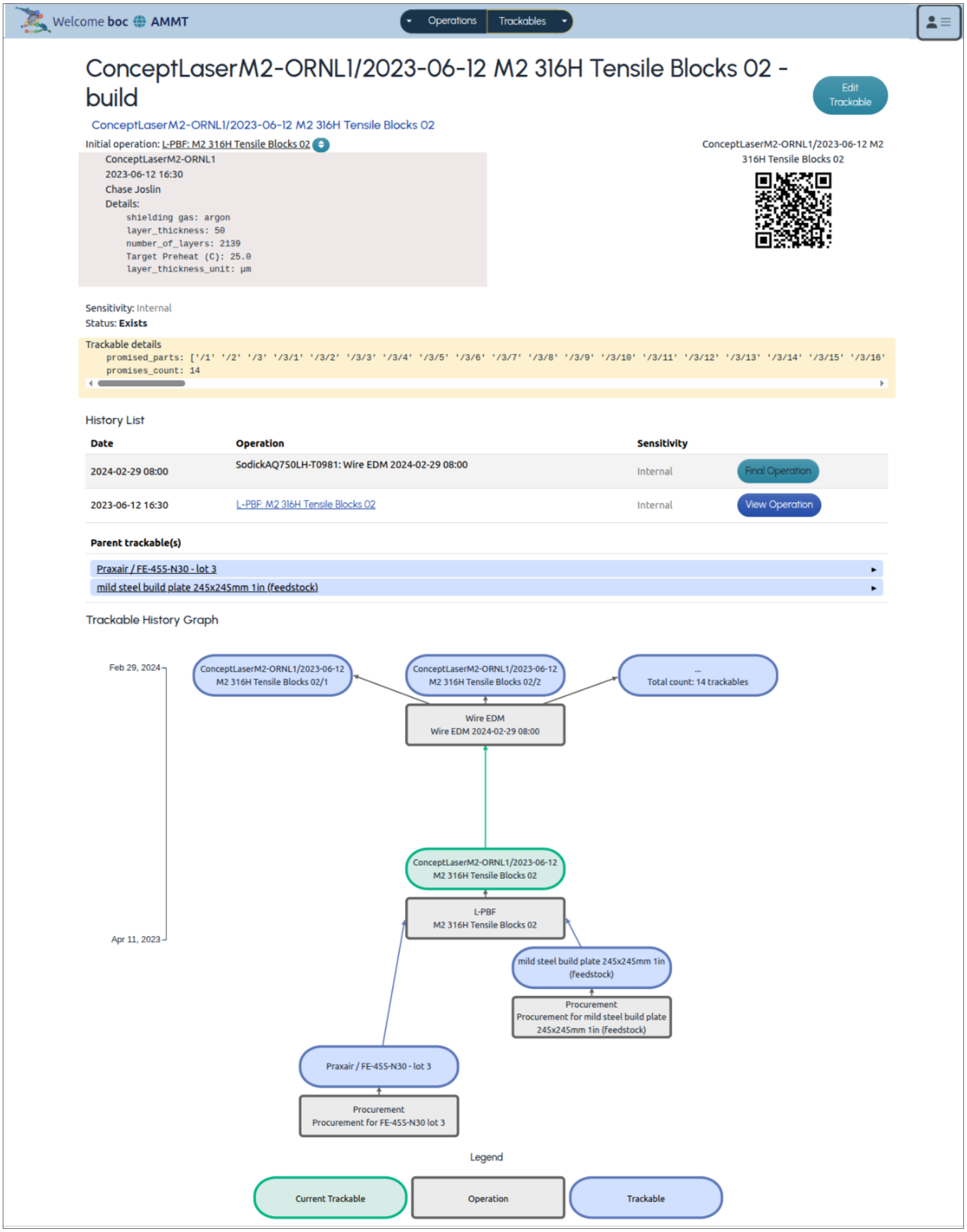


Figure 8. Example of trackable detail view on Damara Tern platform.

A central feature of the trackable detail view is the trackable history graph, which places the trackable within the broader digital thread. This interactive graph illustrates parent and child trackables along with their associated operations, which enables users to explore lineage and dependencies. From the graph, users can navigate through the digital thread linkages to inspect related operations, review associated data, and trace the evolution of a trackable across multiple processes.

3.3 CREATION FORMS

New operations and trackables are added through dedicated creation forms (shown in Figure 9 for operations and Figure 10 for trackables), each enforcing a set of generic metadata for identification and traceability.

To accommodate additional equipment-, operation-, or material-specific metadata, the platform leverages the semistructured database model introduced in Section 2.3 and stores extended fields in JSON format. Schema-driven extensions dynamically generate additional form entries based on user selections, allowing users to enter these extended metadata without directly interacting with the underlying JSON structure. Administrators define these schemas in the administration interface and associate them with operation types and/or machines. As a result, when a user selects an operation type or machine in the operation creation form—or a trackable type or composition in the trackable creation form—the form is automatically updated with the relevant additional fields, which ensures a flexible, context-aware, and user-friendly data entry workflow (Figure 9).

The creation forms also manage relationships between objects. Input and output trackables involved in an operation can be selected from existing records, retrieved via a QR code scan, or created on the fly. Trackables can be created in two ways:

- **Procurement trackables**—Used to register an incoming trackable from an external source. Submission generates both the trackable and an initial procurement operation (Figure 10a).
- **Resulting trackables**—Created as an output of the operation itself. Multiple resulting trackables can be added simultaneously and are automatically linked to the operation's output field (Figure 10b).

This workflow enables users to efficiently manage the metadata and relationships of operations and trackables, which streamlines data entry and maintains full traceability within the platform. Edit forms for operations and trackables follow a similar structure and model as the creation forms, allowing users with the appropriate permissions to modify existing records while preserving metadata, schema integrity, and object relationships.

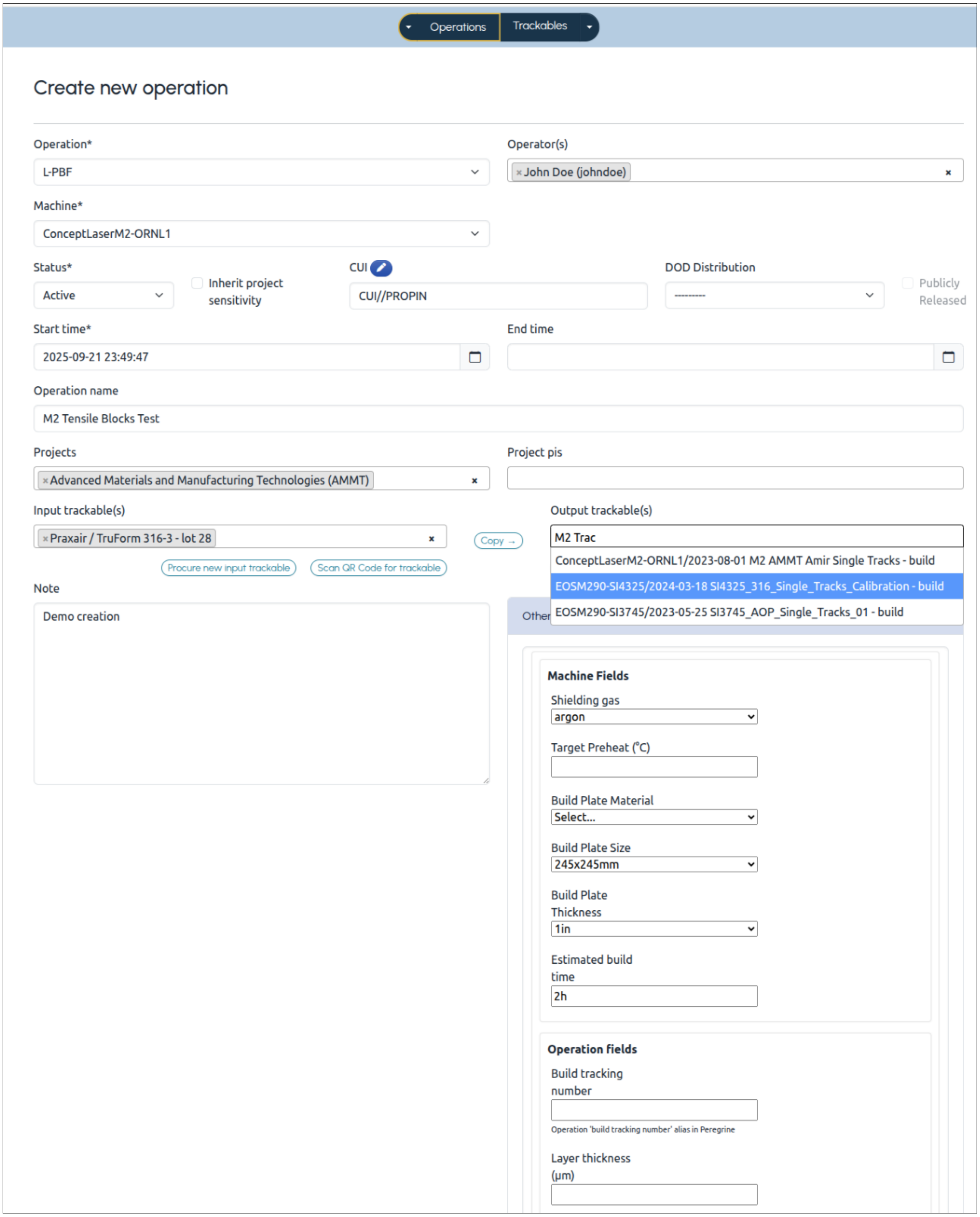


Figure 9. Operation creation form. The selected operation type and machine dynamically populate the Other Fields metadata section with schema-driven fields. Input and output trackables can be selected from existing records, retrieved via QR code scan, or created on the fly using Procure New Trackable or Create Resulting Trackable.

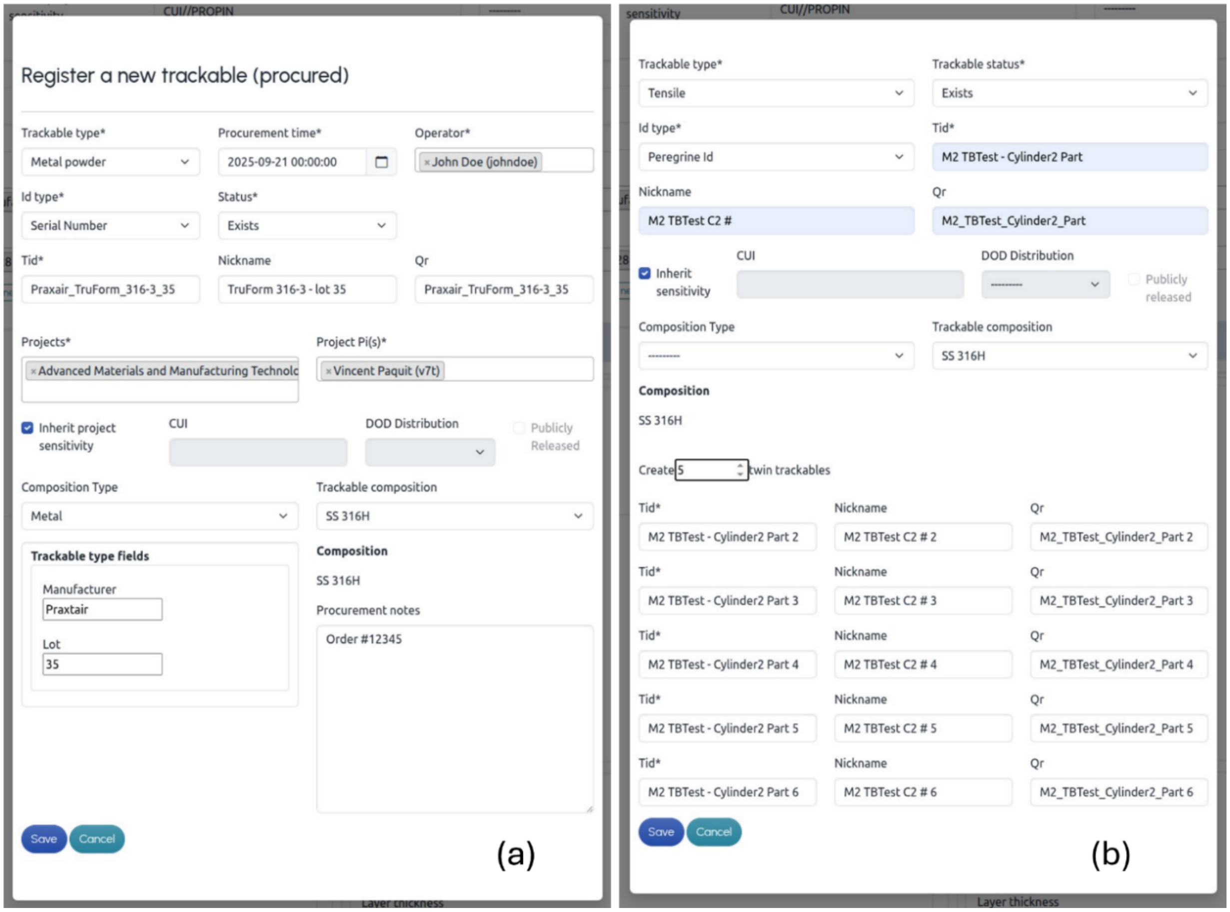


Figure 10. Trackable creation forms example. (a) Procurement form—Submission generates both the trackable and its initial procurement operation. (b) Resulting trackable form—In this example, six trackables (one main + five twins) are created and automatically added to the output trackable field of the operation form.

4. PLATFORM FUNCTIONALITY AND ENHANCEMENTS INTRODUCED IN FY 2025

During FY 2025, the Damara Tern team focused on enhancing usability, workflow efficiency, and platform stability. The team used feedback from users to guide improvements to the interface design, website navigation, and content creation features. Reliability was strengthened through backend refactoring, expanded automated testing, and continuous integration/continuous delivery (CI/CD) integration, which together improved maintainability and robustness. A key effort this year was implementing reliable access control and security measures reinforced with fine-grained permissions, group- and role-based restrictions, protected views, and controlled media delivery, which work together to provide a secure and traceable environment. For the first time, visualization tools integrated with analytical software were introduced to allow users to explore data directly within the platform. The platform's content was also expanded through targeted ingestion of new and legacy datasets, and new operation types and form entries—developed with input from domain experts—broadened the range of operations that could be supported. Finally, all these new functionalities were deployed in a dedicated AMMT program instance, which is now accessible to all AMMT program-affiliated national labs to better support future collaborations. Details of the instance are provided in a separate report [4].

4.1 INTERFACE AND USER WORKFLOW ENHANCEMENT

During FY 2025, significant efforts were made to enhance the Damara Tern user interface and improve overall workflow efficiency. The front-end design was refined to provide more intuitive navigation, including an improved user menu and clearer pathways between key platform pages. Interactive widgets, quick filters, and toggle filtering options were added to facilitate faster data exploration, and forms were updated with prefilled default values to reduce repetitive input and streamline user interactions.

New functionalities were also introduced to extend the platform's capabilities. Machine management pages (shown in Figure 11) follow the same model as the operation and trackable list, detail, and creation views. Through this interface, users can search, view, create, and edit machine records. These features allow non administrators to take part in machine management, with access limited to a designated group of users granted machine manager privileges.

Significant enhancements were made to form validation and data integrity checks to guide users and prevent errors. Error messages were improved, and fail-safes were introduced to catch common mistakes, including duplicate entries, orphan trackables (created as part of an operation that was never fully completed), and timeline inconsistencies, such as assigning a trackable before its creation date.

One of the major workflow improvements targeted the file upload functionality, which now relies on an enhanced backend optimized for large file transfers using the TUS[1] (tus.io) library for the Django Rest Framework. The interface supports multifile uploads, drag-and-drop operations, progress bars, automatic folder creation, tagging, subfolders, and duplicate warnings and provides a more reliable and user-friendly experience when managing operation-related data (Figure 12).

[1] https://github.com/dirkmoors/drf-tus

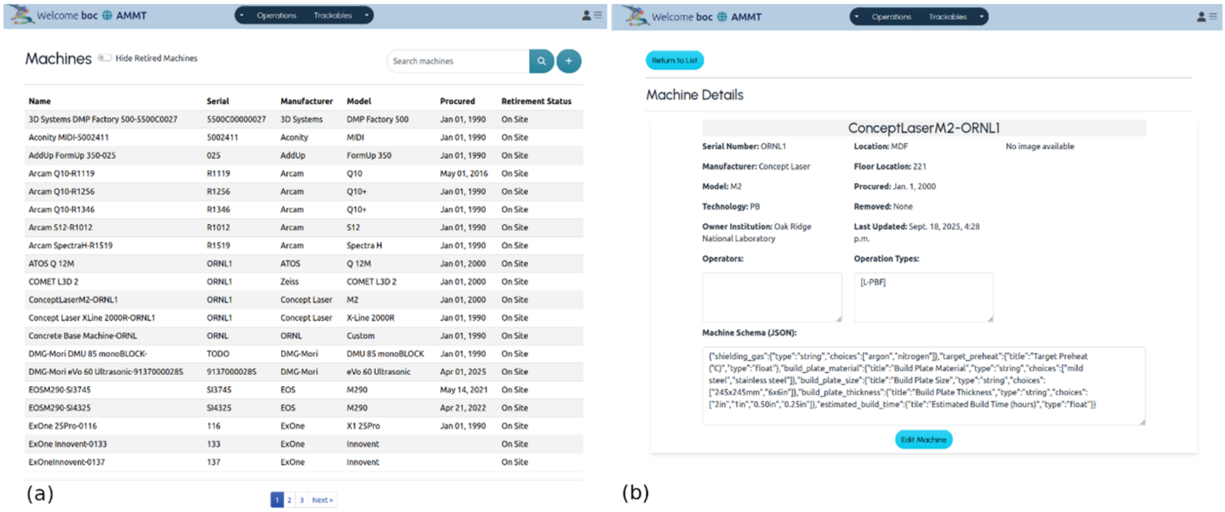


Figure 11. Machine management interface.

(a) Machine list view and (b) machine detail page.

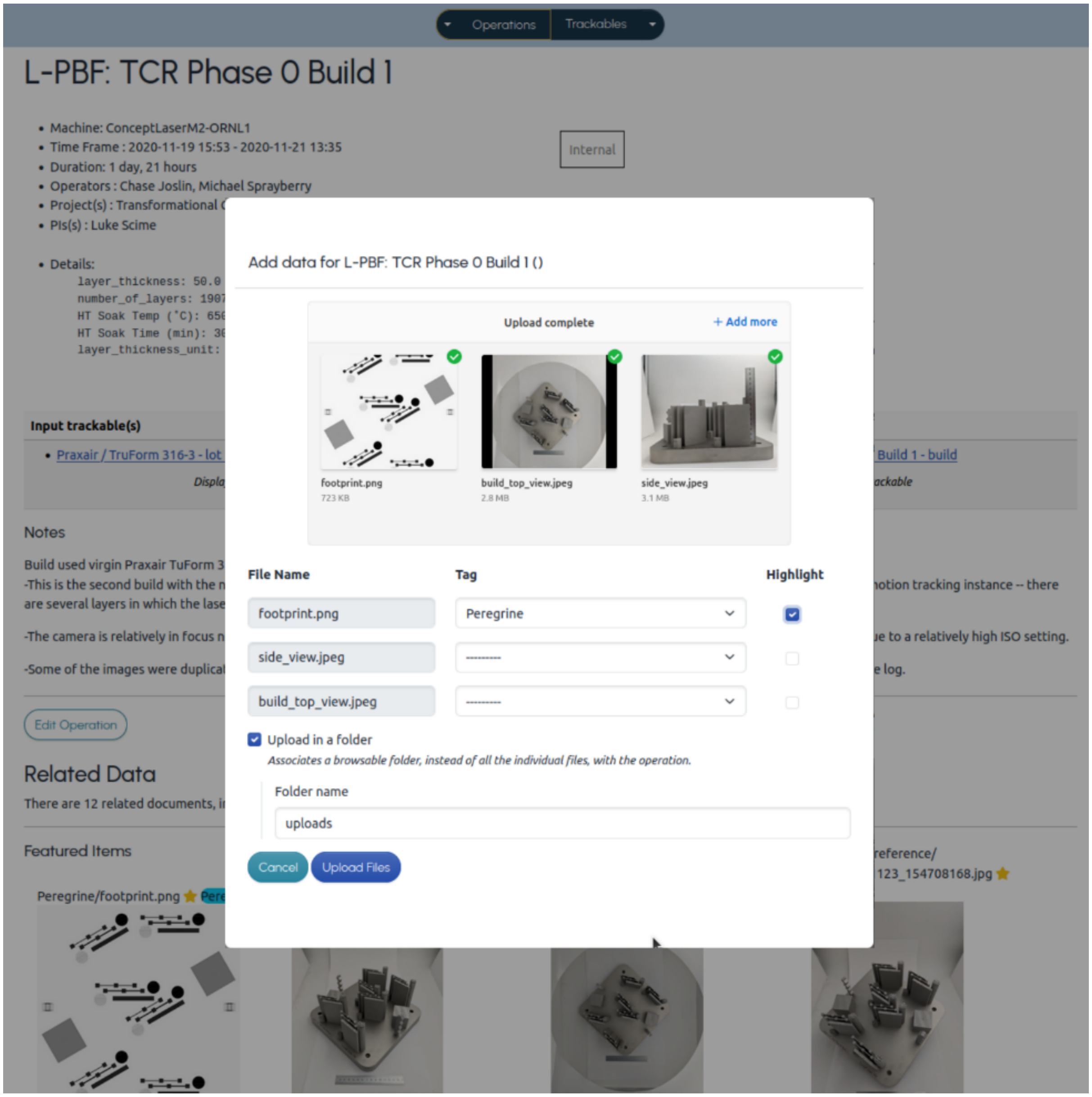


Figure 12. File upload form interface supporting large file transfers with status and progress indicators. Files can be tagged and highlighted during upload and placed in either existing or newly created folders.

In FY 2025, a Controlled Unclassified Information (CUI) label builder form was introduced to the platform, illustrated in Figure 13, which guides users through constructing valid labels using predefined components and allows manual edits for less common entries. Real-time compliance validation ensures consistency with labeling standards, which reduces errors and improves data quality.

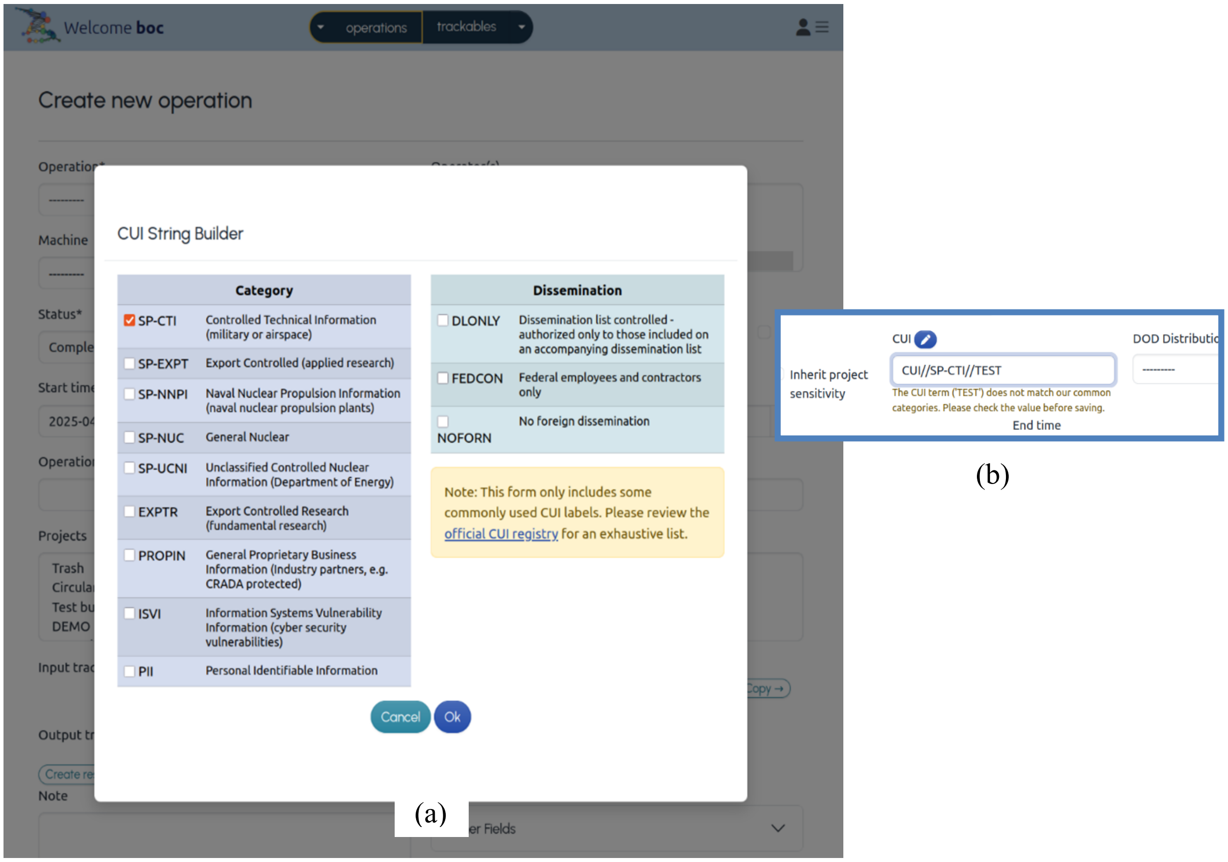


Figure 13. The CUI label builder form, showing (a) guided term selection.

(b) The CUI label field, which is part of the initial form, is editable and provides compliance warning messages when modified outside the predefined selection list.

One of the most requested features was to introduce file browsing navigation through the operation data folders. Data folders attached to an operation can now be explored directly from the operation detail page (see Figure 14). Within this interface, users can navigate the folder structure and open files directly in an exploration modal window with built-in viewers available for supported file types. This functionality streamlines user workflows and provides greater flexibility in related data management. Files no longer need to be explicitly linked or declared as related data to be accessed, which helps keep the list of declared related files concise and meaningful and still ensures comprehensive access to all operation data.

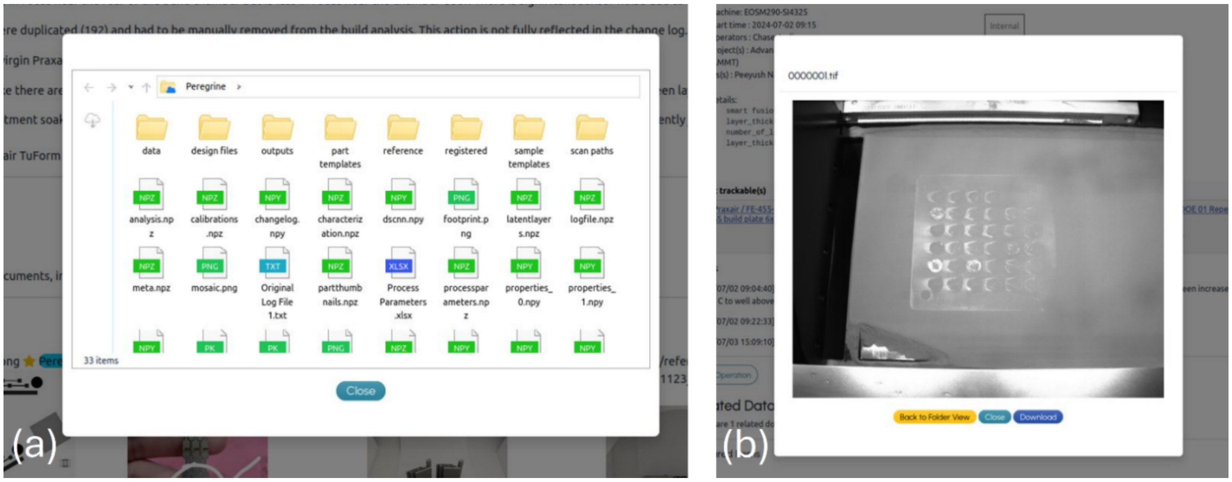


Figure 14. Illustration of the new file browsing and viewing functionality.

(a) Navigation of operation data folders from the operation detail page and (b) file opened directly within the exploration modal window with built-in viewer support for compatible file types.

4.2 CONTENT CREATION

Throughout the year, the Damara Tern development team worked diligently to expand the platform's user base and the data available to users. This effort included adding new machines, such as Renishaw, Arcam, and ExOne Innovent systems; new operation types (e.g., publication, creep testing, machining); and schema-driven entry forms to extend the data collection model.

In parallel, preliminary work was conducted to onboard other operations, adjacent or complementary to additive manufacturing. The team engaged with domain specialists to plan and review the metadata fields required for a variety of related processes, including AMCM machines, XCT, CNC machining, tensile and creep testing, and directed energy deposition. This collaborative effort resulted in drafting the metadata schemas that underpin the schema-driven creation forms described in Section 3.3, as well as a set of example entries. These schemas enable dynamic generation of creation forms for operations and trackables and ensures users can enter standard and extended metadata in a consistent and context-aware manner.

In addition to creating new content, efforts were made to migrate existing data from the legacy Digital Factory into the updated platform structure. This involved mapping legacy records from the prior MDF system to the newly defined operation–trackable model, ensuring metadata consistency and completeness. Special attention was given to maintaining relationships between operations, trackables, and machines, as well as validating historical datasets against the new metadata standards. The migration process, based on dedicated scripts, was validated using all preexisting machine schemas and nonsensitive sample data. The full migration, including sensitive data, is planned for early FY 2026 and will leverage the newly implemented access controls described in Section 4.3.

4.3 ACCESS CONTROL, DEPLOYMENT, AND ROBUSTNESS

4.3.1 Group-Based Access Control and Security Permissions

One of the major features added this year is a role-based access control system to support security permissions and user access restrictions across platform components. These enhancements ensure that access to sensitive data and functionality are governed by user role and institutional affiliation and are validated against CUI and US Department of Defense distribution requirements. The newly implemented group-based permission system for operations and trackables restricts visibility and contribution capabilities (e.g., data upload, object creation) based on a user's role. Roles are defined according to two factors: *affiliation* (either *internal* or *external*), which determines the scope of accessible content, and *contribution level*, which is tiered as *viewer*, *contributor*, or *operator*, thereby defining each user's functional permissions (Figure 15).

For example, internal users may access all nonsensitive operations, whereas external users are limited to publicly released content unless explicitly granted additional privileges. An *operator* is granted full access to create and edit operations and trackables, a *contributor* may upload data to existing operations but cannot modify their metadata, and a *viewer* is restricted to read-only access without the ability to alter any operation or associated data.

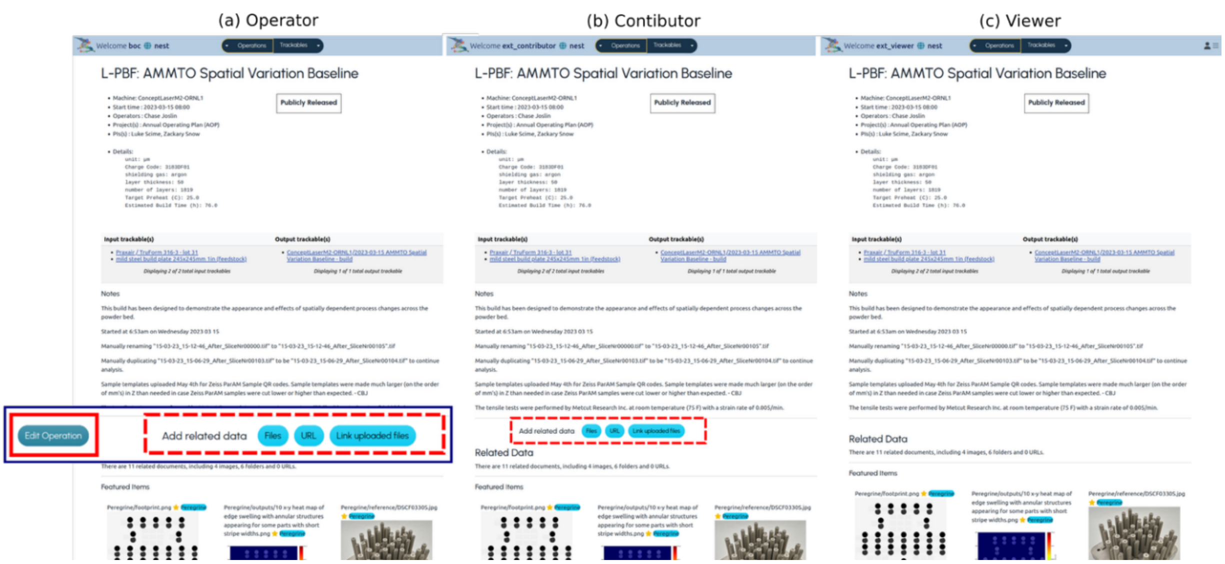


Figure 15. Detail page for a publicly released operation, shown here accessed under three different permission roles.

(a) The operator role grants update and contribution functionalities accessible via the enlarged, red-outlined buttons: (box with solid red line) editing the operation and (box with dashed red line) contributing data. (b) The contributor role displays only the data contribution button, which grants upload and contribution privileges. (c) The viewer role is limited to viewing and downloading; no action buttons are shown for editing or contributing because these permissions are not granted.

In addition to group-level control, the system includes a layer of permissions based on individual user status (whether an individual is a US citizen, US person, or a foreigner). This enables more granular security filtering for datasets subject to specific CUI or US Department of Defense distribution rules. Although user status cannot be fully automated because of the sensitivity of personally identifiable information, the platform incorporates basic lightweight directory access protocol (LDAP)–based automation to prefill user attributes during provisioning when such attributes are available within LDAP directory query limits.

On the implementation side, this fine-grained access control relies on a Django *mixin* (a reusable abstract class that provides functionality via inheritance) called SecurityAccessMixin. This class evaluates the combination of user attributes, including role, institutional affiliation, and citizenship status, against the sensitivity of each object and its ownership (e.g., object operator or creator). Based on these criteria, the mixin determines whether access should be granted for viewing or modifying each object. This implementation strategy allows the functionality to be applied across the various views and objects of the platform. It lays down the path for extending the authentication mechanism in the future to include project- and sponsor-based validation, which could provide additional control over dataset access if implemented.

The new permission infrastructure introduced a notable performance overhead, particularly when rendering large object lists. To mitigate these constraints and reduce the cost of permission filtering, an initial set of caching strategies was implemented to improve responsiveness in permission-restricted list views.

4.3.2 Platform Security and Access Management

To enhance platform security and ensure a reliable user environment, several improvements were implemented in FY 2025 that targeted logging and access control. A centralized logging system was developed to track errors and monitor user activity across the platform. This system captures application-level and user-driven events, providing administrators with detailed context to diagnose issues and maintain system integrity.

In parallel, the platform leverages dedicated access classes (mixins) to protect end points intended for specific functionalities, such as HTMX-driven dynamic swaps or internal interactions, from direct access. These inherited classes act as gatekeepers that ensure only requests originating from the proper workflows or interface components are allowed. Any attempt to access these views directly through the URL or outside the intended context is blocked, which reduces the risk of accidental misuse or unauthorized operations. Administrative users retain explicit bypass privileges, which allow controlled, direct access when needed for debugging, support, or testing purposes. By centralizing these protections in reusable components, the platform enforces consistent access rules across multiple views and maintains flexibility for legitimate administrative operations.

Media files and static content are rigorously protected on all deployed instances using Nginx. Access to these files is only permitted through the platform's views with proper authorization to prevent direct URL access. Functional needs for direct media access, such as viewing and exploring related data within a page, are instead handled via protected media views to ensure files are served securely but remain fully integrated within their contextual workflows.

Token-based authentication was implemented to secure the REST API, which ensures that API end points cannot be accessed directly without valid user credentials and an active token. Enhancements streamline token creation and improve expiration handling, making programmatic access secure and manageable.

The authentication framework was also extended with XCAMS-based authentication, bolstering secure access for external partners. This capability allows users holding an XCAMS ORNL account to securely access the platform from external networks and supports authentication for dedicated collaborative instances, including those used with AMMT program partners. This extension preserves security and access control consistency and enables collaboration with external partners.

4.3.3 Testing Improvements

Significant effort during FY 2025 was devoted to refactoring, cleaning up, and expanding the platform's testing infrastructure. Test loops were implemented to allow scenarios to be applied under varying conditions and parameters, which improved readability, reusability, and maintainability of the test code.

With the introduction of the new access control system, the tests were updated to include scenarios covering different permission levels, user roles, and access statuses. This update ensures that security restrictions and role-based functionality are consistently validated across the platform. Specific testing scenarios were created for operations with varying CUI labels and for test users with unique combinations of authentication groups (e.g., MDF Operator or External Viewer) and access statuses (e.g., US citizen, US person, or none). The view and end point are tested under these conditions to verify correct responses, such as 403 Forbidden for unauthorized access or 200 OK for successful operations.

To improve test development efficiency, the team implemented shared utility classes for testing certain view types, including protected end points and API access points. These utilities systematize testing across multiple end points and enforce consistency in expected outcomes.

Additionally, a test data factory was introduced that is capable of generating content on the fly for testing purposes. The factory ensures that generated objects respect relationships defined in the data model, leverage existing test database content where available, and support automatic cleanup after tests are completed. This approach improves test reliability and reduces the overhead of manual test data management.

The platform now integrates approximately 220 automated tests covering backend functionality, user permissions, and the first front-end tests for the home page. Test coverage across key applications is monitored and evaluated using the [2] package, which ensures comprehensive insight into tested code paths. Current test coverage across key applications is summarized in Table 1.

Table 1. Summary of automated test coverage for Damara Tern applications, including backend functionality, user permissions, and front-end tests.

The table shows the number of tests implemented per application and the current percentage of code coverage, measured using the pytest-cov package.

App	Test count	Coverage (%)
home	9	55
xct	20	93
api	115	93
operation:	—	63
• test_operation	28	—
• test_machine	4	—
• test_project	14	—
trackable	9	50
mdf_db	3	NA
peregrine	3	48

Before implementing the new security restrictions, test coverage for the Operation app was approximately 50%, and the API end points were covered at 75%. After adding permission-based test scenarios and further test case expansion, coverage has now increased to 63% for the Operation app, 50% for the Trackable app, and 93% for the API app.

These improvements collectively enhance the platform’s reliability, ensure robust enforcement of security and access controls, and provide a scalable foundation for future test expansion as new functionality is added.

[2] https://pytest-cov.readthedocs.io/en/latest/reporting.html

4.3.4 CI/CD Deployment Pipeline

The Damara Tern team expanded its infrastructure services and deployment capabilities by containerizing the application and its support services, which established a consistent runtime across all environments. All instances are deployed as Docker containers organized into four primary components: PostgreSQL (database storage), Nginx + Gunicorn (web server), Redis (caching and queue management), and Django (application framework).

A fully automated CI/CD pipeline, illustrated in Figure 16, now governs promotion from commit to release. The pipeline automates all stages of deployment through three main phases: build (bundling JavaScript and style sheet assets, collecting and preparing static files), test (lightweight validation checks can be enabled to confirm a correct environment before deployment), and deploy (using Docker Compose to apply branch-specific configurations, mount resources, and provision shared storage). Together, these steps ensure consistent, repeatable deployments across environments.

For each change request, the pipeline provisions isolated, on-demand quality assurance dynamic environments and manages their life cycle, from creation to cleanup, with minimal manual intervention, supporting integration testing, debugging, and code review. This setup enables rapid provisioning, continuous updates, and reproducible environments, which accelerates the feedback cycle and improves reliability and confidence from one release to the next. To further support maintainability and scalability, platform settings were reformatted and centralized to simplify configuration, which reduced drift across deployments. Package dependencies were also refined to ensure uniformity, with all dependencies now served locally.

In parallel, three additional collaborative platform instances were introduced and seeded with representative sample data. These instances engage a broader and more diverse group of users, including external teams and institutional partners. Currently, the CI/CD pipeline supports the following static environments, which are persistent and continuously maintained:

- **Collaborative/external instance for AMMT program partners** – A stable environment for validating workflows, gathering user feedback, and testing integration with external collaborators. This platform is the focus of another AMMT program work package [5].
- **External API integration and interconnection instance** – Shared with non-ORNL contributors, this platform provides API end points for prototyping and testing interconnections with other data management platforms.
- **Internal development instance** – A controlled environment for ORNL contributors to prototype features, validate functionality, and test software integrations. This instance is used for secondary development work and testing, not for the core Damara Tern development.

These improvements streamline CI/CD-managed instances and facilitate the packaging and deployment of stand-alone Damara Tern instances at external sites that operate outside the primary CI/CD pipeline.

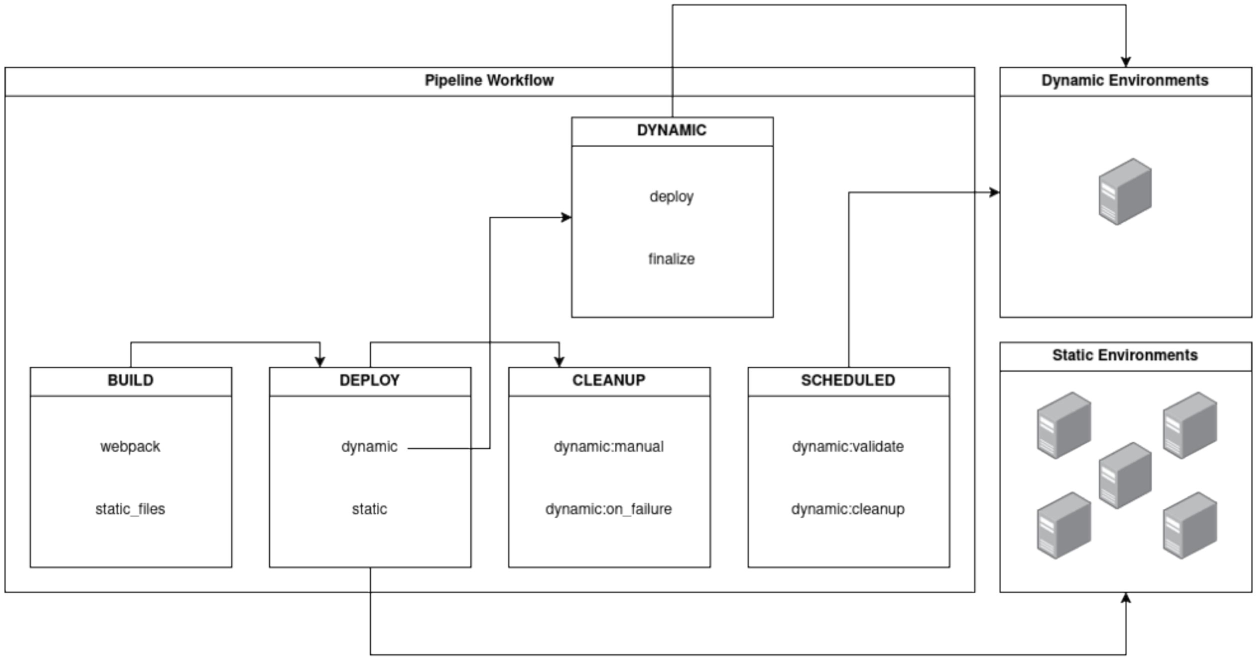


Figure 16. Overview of the Damara Tern CI/CD deployment pipeline. The pipeline automates the building, testing, and deployment of multiple platform instances. Feature branches trigger dynamic environment instances for development, testing, and validation, and static instances provide stable environments for collaboration, integration, and internal development.

4.4 FUNCTIONALITIES SPONSORED BY OTHER PROGRAMS

In FY 2025, the US Department of Energy Advanced Materials and Manufacturing Technologies Office within the Office of Energy Efficiency and Renewable Energy supported the development of complementary functionalities for Damara Tern. These enhancements centered on visualization tools and software integration, which were delivered as modular Django apps. Designed for flexibility, they can be readily enabled in platform instances, broadening the range of available capabilities.

4.4.1 Visualization Functionalities

During FY 2025, Damara Tern's visualization capabilities were significantly expanded, which enabled users to review and interact with data directly in the browser without requiring downloads or external tools. These features were implemented as modular Django apps, allowing them to be enabled or disabled across platform instances as needed.

A TIFF viewer (Figure 17a), HDF5 viewer, and STL viewer (Figure 17b) were initially deployed within the Operation app as proofs of concept that provided quick inspection of images, interactive exploration of hierarchical datasets, and 3D model inspection, respectively.

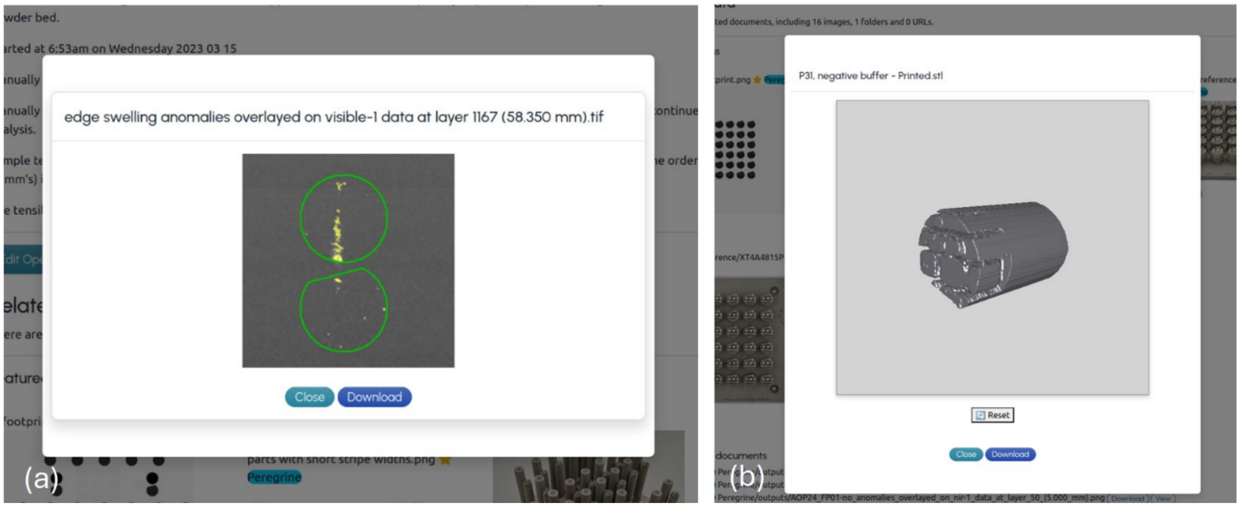


Figure 17. Illustration of newly implemented visualization capabilities: (a) TIFF viewer and (b) STL viewer. Both viewers can be opened directly from the operation detail page in a modal window, allowing interactive inspection of the associated data.

For tabular data, a CSV viewer was integrated into the dedicated Visualization app, which enabled interactive browsing of structured data within the platform. Building on this, a Tensile Testing app was developed to provide a dedicated viewer for mechanical test results stored in standardized CSV files. This viewer offers immediate visual feedback, supports zooming and navigation, automatically generates preview thumbnails, and fully integrates multiplot functionality, which allows direct comparison of sibling tensile tests derived from a common part (Figure 18).

Together, these visualization tools mark a major step toward more user-centered workflows, giving researchers a more intuitive and efficient way to explore, validate, and compare their data directly within Damara Tern.

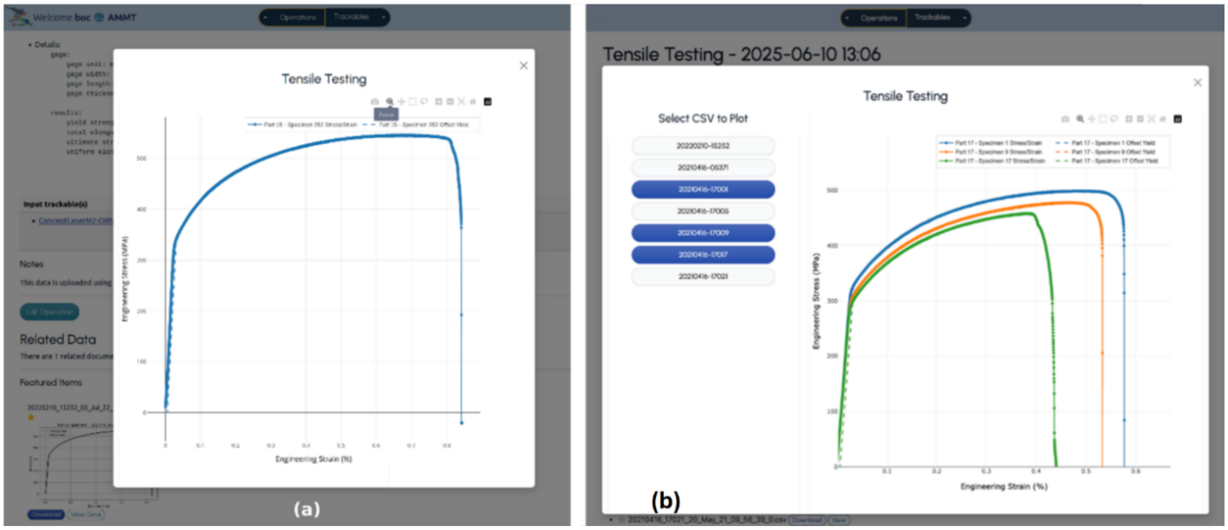


Figure 18. Tensile test plot viewer showing (a) a single test plot and (b) the multiplot comparison feature for sibling tensile tests.

This viewer allows users to consult and compare results derived from a common part.

4.4.2 API and Software Integration

During FY 2025, the Damara Tern generic API was extended to support programmatic queries to enable users and external tools to efficiently access individual trackables, lists of objects, and partial updates to records. Figure 19 provides a summary list of the current API end points. These core end points provide a flexible foundation for automated information retrieval and focused data access.

Building on this foundation, the team integrated the platform with locally developed and commonly used ORNL software, including Peregrine, Hyacinth, and Robot Operating System. These integrations guided the refinement and validation of the generic API end points, ensuring that the available programmatic interfaces met real-world operational requirements and could reliably support external tool workflows.

Server status API

GET /api/serverstatus/ Retrieve status of server and user authentication status

Authentication

POST /api/token-auth/ Submit encrypted credentials to obtain an authentication token

Machine API

GET /api/machines/ List all machines

GET /api/machines/{id}/ Get machine details by ID

GET /api/machines?optype={id} List machines filtered by operation type

Operation Type API

GET /api/operationtypes/ List all available operation types

Operation API

GET /api/operations/ Retrieve operations, filtered by date, machine, operator, or name

GET /api/operation/{id}/ Fetch details for a specific operation

GET /api/operations/metadata/ Get metadata for builds, filtered by date, name, and machine

POST /api/operation/update/{id}/ Update operation metadata attributes

Peregrine

POST /api/operations/announce_new_files/ Trigger and confirm the addition of new data files for an operation

POST /api/operations/list_new_ops/ Detect new builds for given machines available for Peregrine analysis

Trackable API

GET /api/trackables/ Retrieve trackable items, filtered by ID, nickname, TID, etc.

GET /api/trackable/{id}/ Fetch details for a specific trackable item

POST /api/trackable/update/{id}/ Update trackable attributes

Figure 19. Summary list of API end points, including generic end points and two specific Peregrine end points.

In addition to the core generic API implemented within the Django api core apps, a dedicated Peregrine app was developed to handle integration-specific functionality, including authentication, license verification, and operational metadata exchanges. Token-based authentication was implemented to provide secure access to API end points, and all new end points were accompanied by automated tests to ensure reliability and proper access control.

Together, these developments validate and strengthen the platform's generic API and provide a modular framework that supports targeted data retrieval, integration with analytical and operational tools, and creation of custom pipelines for knowledge extraction and visualization tailored to specific scientific workflows.

5. CONCLUSION

During FY 2025, the Damara Tern platform underwent significant enhancements, strengthening its functionality, security, and usability. Key achievements include improved user interfaces and workflows, implementation of fine-grained access control and role-based permissions, and expanded automated testing covering a broader range of scenarios, which ensured the integrity of core content creation, management, and access features. The fully automated CI/CD pipeline now enables rapid, reproducible deployments across multiple platform instances, supporting both internal development and collaborative external engagements. Dedicated external instances were deployed, including an AMMT program-specific collaboration instance,[3] to facilitate secure collaboration with external partners and institutional users.

In parallel, the platform's data model and content infrastructure were extended by adding new machines, operation types, and schema-specific objects to support advanced manufacturing workflows. Preliminary work on onboarding adjacent operations produced draft metadata schemas and test entries, which established a foundation for future integration and enhanced data richness.

Together, these enhancements have improved platform reliability, scalability, and interoperability, which has enabled seamless integration with external analytical tools and collaborative environments. Damara Tern is now better positioned to support diverse scientific workflows, facilitate secure collaboration with internal and external partners, and provide a robust framework for future feature expansions. By combining usability improvements, rigorous security controls, and infrastructure modernization, the platform demonstrates readiness to meet the evolving needs of advanced manufacturing research and data-driven experimentation.

Going forward, Damara Tern will focus on expanding its analytical capabilities by integrating advanced data processing and artificial intelligence–driven tools to support predictive modeling and decision-making across manufacturing workflows—core functionalities of the MDDC platform. Efforts will include developing standardized interfaces for multi laboratory data sharing, further refining metadata schemas to enhance interoperability, and implementing automated provenance tracking to ensure data traceability and reproducibility.

The platform will also prioritize scaling collaborative access, with the aim of enabling additional AMMT program participants and external partners to securely contribute, explore, and analyze datasets. Continuous enhancement of the user experience, combined with monitoring and optimization of performance and security, will ensure that Damara Tern remains a reliable, flexible, and high-impact tool for the program.

These forward-looking initiatives will position Damara Tern to accelerate scientific discovery, support robust certification and qualification processes for additively manufactured components, and serve as a foundational element for the broader vision of a distributed, collaborative, and data-driven American Science Cloud.

[3] https://damaratern-ammt.ornl.gov/

6. REFERENCES

[1] S. Cooper, G. Accawi, A. Barnes, W. Forman, G. Jones, W. Halsey, V. Orlyanchik, V. Paquit, J. Price, L. Scime, Z. Snow, A. Wille and A. Ziabari, "FY 2024 Multidimensional Data Correlation Platform: Unified Software Architecture For Advanced Materials And Manufacturing Technologies Data Management And Processing," USDOE, 2024.

[2] L. Scime, J. Haley, W. Halsey, A. Singh and M. Sprayberry, "Report on progress of correlationof in-situ and ex-situ data and the use of artificial intelligence to predict defects.," 2020.

[3] US Departement of Energy - Office of Science, *The American Science Cloud (AmSC),* USDOE, 2025.

[4] Stephanie Cooper , A. Gina, B. Abby, E. Hayden, F. Wes, S. Randall, W. Annika and P. Vincent, "Instantiation of the Damara Tern Platform for AMMT Collaborative Data Management," USDOE, 2025.

[5] S. Cooper, G. Accawi, A. Barnes, H. Esser, W. Forman, R. Stapleton, A. Wille, V. Paquit, J. Price, L. Scime, Z. Snow, A. Wille and A. Ziabari, "Instantiation of the Damara Tern Platform for Advanced Materials and Manufacturing Technologies Program Collaborative Data Management," 2025.