Research Article

Open Access

Maria Esteva,* Ramona L. Walls, Andrew B. Magill, Weijia Xu, Ruizhu Huang, James Carson, Jawon Song

Identifier Services: Modeling and Implementing Distributed Data Management in Cyberinfrastructure

https://doi.org/10.2478/dim-2019-0002
received September 6, 2018; accepted February 27, 2019.

Abstract: The Identifier Services (IDS) project conducted research into and built a prototype to manage distributed genomics datasets remotely and over time. Inspired by archival concepts, IDS allows researchers to track dataset evolution through multiple copies, modifications, and derivatives, independent of where data are located – both symbolically, in the research lifecycle, and physically, in a repository or storage facility. The prototype implementation is based on a three-step data modeling process involving: a) understanding and recording of different researcher workflows, b) mapping the workflows and data to a generic data model and identifying functions, and c) integrating the data model as architecture and interactive functions into cyberinfrastructure (CI). Identity functions are operationalized as continuous tracking of authenticity attributes including data location, differences between seemingly identical datasets, metadata, data integrity, and the roles of different types of local and global identifiers used during the research lifecycle. CI resources were used to conduct identity functions at scale, including scheduling content comparison tasks on high-performance computing resources. The prototype was developed and evaluated considering six data test cases, and feedback was received through a focus-group activity. While there are some technical roadblocks to overcome, our project demonstrates that identity functions are innovative solutions to manage large distributed genomic datasets.

Keywords: data identity, content comparison, distributed data management, data modeling, life sciences, cyberinfrastructure

1 Introduction

As open science and open repository movements gain ground, a disconnect remains among in-research, published, and reused data. This is especially true for projects in which large datasets are stored and analyzed across different computational resources by researchers at different institutions, conducting studies as parallel workflows, over the course of several years. The isolation and disparity of data and processing components of a same project make it difficult to track relations between data sources and derivatives and to maintain adequate metadata across systems. This gap poses extra burdens for researchers and, consequently, for the sustainability of an open data environment. Furthermore, repositories generally accept only final datasets and many accept only a segment of the complete set (e.g., the National Center for Biotechnology Information (NCBI) (2018), only accepts sequence data). As a result, the bulk of curation happens when data are about to be published, and researchers often have to resort to multiple repositories to publish all of a project's data. This leads to a disconnection among data publications and between active, published, and non-published related data. Whereas data in repositories remain preserved and stable, the data components continuously managed by the research team may change in both content and location. Without mechanisms to track relationships and changes among data components over time, repositories and researchers cannot create a useful and sustainable network of data relations and semantics.

To overcome these challenges, we conducted research into a set of complementary identity functions that can be used to track relations among in-research, published, and reused genomics data in a distributed environment.

***Corresponding author: Maria Esteva,** Texas Advanced Computing Center, The University of Texas at Austin, Austin, TX, United States, E-mail: maria@tacc.utexas.edu
Ramona L. Walls: CyVerse, University of Arizona, Arizona, United States
Andrew B. Magill, Weijia Xu, Ruizhu Huang, James Carson, Jawon Song: Texas Advanced Computing Center, The University of Texas at Austin, Austin, TX, United States

These functions use different data attributes to establish identity, such as: metadata, checksums, local and global identifiers, file locations, results of data content comparisons, and the sources that data belong to. One or several of these attributes and the relations between them can uniquely distinguish files or groups of files as belonging to a specific class, such as a research stage or process (e.g., alignment data, analysis, data, and published data). Identity functions can contribute to making data findable, accessible, interoperable, and reusable (FAIR) (Wilkinson et al., 2016). Identifier Services (IDS) research is framed by traditional archival principle of authenticity in the digital domain (Guercio, 2001). It is also informed by the notion of postcustodialism, which proposes that there is no need for digital objects to reside under one custodian (e.g., a repository) as long as archival principles guide their management and long-term preservation (Henry, 1998). As growing and evolving data are used and stored in a distributed ecosystem, different organizations and people can become their stewards. IDS is intended to be an instrument that contributes to manage such complexities.

Our research methodology took a prototype implementation and evaluation approach in which identity functions were accessed and carried out via the IDS, a web portal through which research teams could test and evaluate them. In the portal, identity functions were realized through different computational analyses performed on the files (e.g., content comparison, integrity, and location checks).This prototype focused on genomic studies, because they encompassed many of the challenges we were trying to address: a single project can contain many and large (tens to hundreds of GB) files; many projects involve several researchers at different institutions; data analysis involves multiple steps; and cyberinfrastructure (CI) for data storage, sharing, analysis, and publication exists but as largely siloed resources. The goals of IDS are to:

- provide CI that significantly improves management of large distributed datasets at any point of their lifecycle;
- allow users (individuals and repositories) to organize their data and metadata according to their research workflows;
- relate and represent dispersed data, independent of where the data are located and whether data are partial or complete, duplicated, private or public, or active or static – at any point in the research lifecycle;
- schedule services to check location, integrity, and content similarity of data over time to verify its evolution;
- gather users' requirements and evaluate their adoption of IDS; and
- identify gaps and needs to advance large and distributed data management.

We operationalized the goals within a CI, encompassing the networks, web services and UI, cloud, databases, and high-performance computing (HPC) resources needed to develop, manage, and conduct our proposed identity functions. IDS is not a storage system but rather software and services that allow interactions with files in remote storage. The design and implementation of the prototype were informed by real-world genomic test cases, for which we listened to the needs of researchers working on projects conducted across remote teams and involving multiple storage and publishing locations. Because IDS had to accommodate their requirements within the envisioned functionalities, prototype implementation and evaluation of the test cases were accomplished in parallel. Researchers registered their datasets and metadata in IDS and provided feedback that was used to adjust development. This brought up research and technical challenges, both conceptual and practical. In addition, before ending the research project, we formed a focus group to assess users' adoption.

In IDS, users create custom data models that represent the materials (e.g., specimens, probes), processes (e.g., sequencing, genome assembly), and data as entities[1] that relate to specific lifecycle stages of their research projects. Users can register and associate large data that are distributed across storage systems, including similar data instances, with these entities and upload corresponding metadata via web forms or in bulk. Around the different entities, the metadata are used to differentiate the structure and components of the dataset. Once data are registered in IDS, researchers can use their project landing page to initiate or schedule identity functions on their dataset, such as verifying integrity and obtaining digital object identifiers (DOIs). To help manage large collections, users can keep registering new data and can also create subsets of the data that automatically preserve the modeled relations among all the entities. Functions requiring computation, such as checksum calculation and content comparison, are run on HPC resources and report results back to IDS. Repeated over time, the metadata and the results from the identity functions create a representation of a dataset's provenance and evolution. IDS can be used by researchers to track data stored at multiple locations,

[1] Entities are how we operationalize classes of files in a data model.

and vice versa, by repositories and data stores to continue tracking the evolution of the datasets.

This paper is organized as follows. In the Related Work section, we discuss authenticity as the conceptual framework for IDS, and review how different repository and data management projects support identity functions and implement data models. Along with each point, we clarify the novel contributions brought by IDS. In the Research Methodology section, we describe how we operationalized and implemented each identity function in the IDS prototype. We note each as a task including its evaluation through a data test case, and we describe the results of a final focus group. In the Conclusions section, we discuss the outcome and future needs.

2 Related Work

Authenticity is an archival principle that considers provenance, content, context, and integrity to attest that a record is what it claims to be over time and space (Hirtle, 2000). Mechanically, establishing authenticity for a digital object may involve verifying the integrity of the bits, its metadata, and its identifiers. Conceptually, an intelligent system could help determine and document an object's authenticity through series of related assertions about a digital object in connection to its provenance, versions, and derivatives (Lynch, 2000). However, while archival principles are making their way into the realm of digital data curation, they have not been developed into automated and scalable methods yet (Ray, 2012). IDS is designed to bridge this gap. In it, authenticity is operationalized as different identity functions within a scalable CI.

2.1 Data Identity and Repository Systems

As well as opportunities, open data movement introduces new challenges to establishing identity. In the current scientific environment, data are an "unruly and poorly bounded object" (Wynholds, 2011), existing as multiple versions, which may be stored in different places at different times and, in many cases, bear different identifiers. In this setting, a key identity function should allow files or groups of files to be distinguished from one another at any time in a dataset's lifecycle. As long-term custodians of data, existing repository systems fell short of performing this function. Most have poor or no mechanisms to continuously assess identity beyond validating integrity via checksums completed at ingest (Factor et al., 2009). Furthermore, checksum algorithms are of limited use for large data, because they cannot tell what has changed. Finally, repositories have limited mechanisms to link to external related datasets, and current initiatives are more focused on linking data to related publications (Hoogerwerf et al., 2019). This landscape suggests the need for solutions such as IDS that can track, validate, and document data identity in a continuum and at scale, regardless of where data are stored. In this postcustodial environment, IDS can be used by researchers and by repositories to join different instances, stages, and processes of data.

2.2 Infrastructure for Distributed Data

Several pieces of infrastructure exist to manage the lifecycle of life sciences data. Platforms such as Syndicate (Nelson & Peterson, 2014) and CyVerse (Merchant et al., 2016) are geared toward distributed data storage and management. Both allow researchers to scale their data storage, share data with others, and offer data publication services. Galaxy (Afgan et al., 2016), another popular life sciences data analysis platform, is not geared toward data storage or publication but does include metadata features. While projects such as these are crucial for supporting biological research, they do not offer identity functions for distributed data.

The Open Science Framework (n.d.) is a free, open-source web platform for managing research projects between multiple collaborators. In addition to storage within OSF, you can link data to commercial cloud platforms, add metadata, and organize data. OSF issues DOIs for self-publishing snapshotted versions of projects but neither includes data analysis tools nor supports direct publication to external repositories. Unlike IDS, OSF stores the data on its own platform and does not support distributed data management. For researchers working in a programmatic environment, Synapse (Bionetworks, n.d.) provides an open-source platform to carry out, track, and communicate their research in real time. It enables co-location of data, code, and results and narrative descriptions of that work. It can be connected to cloud computing resources, provides wikis for project management, and mints DOIs for published projects. Importantly, Synapse works with big data, but it also relies on a single storage location and does not provide a full suite of identity functions.

COPO (2015) is a portal focusing on plant scientists to store and retrieve data, making it easy to add metadata through a user interface that collects information about different aspects of a project (e.g., specimen, analysis, sequences). The COPO interface makes helpful suggestions regarding what information you might want to submit and normalizes metadata, which can be uploaded in bulk, to controlled vocabularies, ontologies, and community standards assisting with data integration. In addition, publications can be submitted through COPO to long-term storage repositories. IDS could be a complement to this project by tracking the evolution of the different components of a dataset.

At the end of a research project's lifecycle, data repositories continue to play a crucial role in providing FAIR data. The International Nucleotide Sequence Database Collaboration (INSDC) (2018) coordinates efforts among three large international repositories, including NCBI in the US. Most journals require researchers to submit their sequence data to an INSDC repository before a corresponding article can be published. Dryad (Vision, 2010), a repository heavily used by the life sciences community and now accepting all kinds of research data, partners with journal publishers to accept data connected to a publication. Like numerous other topic-specific repositories that exist in life sciences, they have shortcomings. Only a few, like Dryad and the CyVerse Data Commons, publish the full dataset associated with a study, and in the case of the former, there are size limitations. Most do not include a data model to support flexible data organization, providing general or domain-specific metadata standards. As a result, researchers often end up publishing their projects in more than one repository, leading to related data that are not linked in any way. These gaps indicate a need for IDS to manage and relate multiple copies or instances of datasets across different repositories and data stores.

2.3 Data Modeling

Data modeling is the process of describing the entities that are important to a system and how they relate to one another (West, 2011). Many researchers are familiar with modeling methods, which they use to operationalize research problems as code and workflows. However, they do not necessarily use them to organize their data. Instead, during active research, the vast majority of researchers use hierarchical file structures and file naming conventions to organize their data on local computers, so this is the method that gets adopted when they move into big data on remote storage systems (Gray, 2005). Unfortunately, simple hierarchical systems lack the flexibility to reorganize data as needed and do little to support data understandability and discovery. Thus, they are not well suited for managing big data.

Data modeling is the first step in building an information system. In repository systems, data models are the backbones of how data are packaged in relation to metadata and access functionalities. Repositories such as Dspace (Phillips & Koenig, 2008), Fedora (n.d.), and Dataverse (Gary, 2007) each use a unique data model that is tied to one or more metadata schemas to represent published datasets. Repositories that use data models for managing data across the lifecycle are not common, but an example outside of life sciences is worth reviewing. DesignSafe (Rathje et al., 2017), a CI for natural hazards engineering, uses four different data models to allow interactive curation across the lifecycle, for publicly representing data obtained by hybrid simulation, experiments, field reconnaissance, and simulation research methods used by the community.

Most disciplinary databases use a relational model to organize their data. For example, INSDC database uses models that relate projects, specimens (bio-samples), sequences, and genome features. COPO uses a model that allows users to connect specimens and experiments to specific data. Generalized data models for sharing and discovering data on the World Wide Web include the Portland Common Data Model (Duraspace, 2016) and the Open Resource Exchange (ORE) (2014) model. DataONE uses a modified version of ORE to describe the approximately 800,000 datasets it aggregates and indexes (2015). Several ontologies can also be used as general data models. The Provenance Ontology (PROV-O) (2013) is a high-level ontology based on the PROV data model for provenance (2013). Other ontologies have been developed to record and integrate scientific data, i.e., experiments or measurements by humans or sensors (Haller et al., 2018; Madin et al., 2007; Walls et al., 2014). A preliminary work showed that these ontologies are compatible and can be mapped to PROV-O (Semantic-Observations, 2016), suggesting that the core concept of linking entities and activities is useful for recording information about research. The DCC Curation Lifecycle Model indicates the stages of a research process, from data generation to reuse (DCC Curation Lifecycle Model, 2016). In IDS, we use the DCC model as a framework to allow users to build their own models to represent the steps and processes involved in their research. IDS links physical files stored elsewhere to the processes (as entities and metadata) that use and generate them. The IDS generic data model maps to PROV (see Figs. 1A, 1B, and 1C).

3 Design and Implementation of the IDS Prototype as a Research Methodology

IDS research was operationalized and implemented as a set of prototype services available through a web interface. The research methodology consisted of building seven tasks. Completion and results of the tasks were evaluated using genomic data test cases and involving the researchers that created them. Along with the task narrative, we report how we tested each.

Task 1: Gather Genomic Data Test Cases

Working with six research teams on their real genomics test cases, we were able to gather their data management requirements, build and test IDS functionalities, and adjust them as needed. The standard procedure for analyzing the test cases consists of multiple steps. First, IDS personnel worked with collaborators to describe their research workflows including the timeline, processes involved in the project (e.g., specimen collection, data analysis, and data preprocessing), data types (including which types need to be preserved or published), and expectations for transitioning from in-research to public data. All the test cases were documented on the IDS project wiki. In the following sections, we describe two cases that were fully executed in the IDS prototype and used to illustrate this paper.

3.1 Maize Methylation

This project (Li et al., 2015) performed whole-genome bisulfite sequencing (WGBS) for five maize (*Zea mays*) genotypes, resulting in an NCBI Sequence Read Archive (SRA) (n.d.) submission of five FASTQ files. The majority of the research and derived publications focus on analyses of one hundred base pair (100 bp) tile files that report the outcome of the alignment and analysis of methylation. Researchers would like to share them publicly. For this, they wanted DOIs for each maize genotype as subsets of the complete dataset that includes one 100 bp tile file, a copy of the corresponding sequence file deposited at SRA, and a description of how those were created. This was important because the same underlying sequence data may be reused to create a new 100 bp tile with altered algorithms or based on alignment to an updated reference sequence. The complete dataset was stored on Corral (n.d.) at the Texas Advanced Computing Center (TACC) (n.d.) during research, and it was later published in CyVerse Data Commons (Springer, N, 2017a; 2017b; 2017c; 2017d; 2017e).

3.2 High-Throughput in Situ Hybridization (HT-ISH)

Data for this case came from the LungMAP initiative (Ardini-Poleske et al., 2017) , which focuses on the development of the lung just before and after birth. The project collects HT-ISH data on gene expression in the brains of mice and humans. A thin slice of lung tissue is marked with a probe for a specific gene, and cells that contain that gene transcript show up with a dark purple marker. A gene could potentially have multiple different probes, each performing differently, and the RNA sequence for each probe must be tracked. The tissue specimens and the images produced need to be tracked along their metadata, and there may be multiple version of an image. Images and their metadata are collected at two locations, first to an account on CyVerse's BisQue online image viewer (Kvilekval, Fedorov, Obara, Singh & Manjunath, 2010). Once image data have been inspected, cleaned, and approved, they are sent to a LungMAP data coordinating center, which makes them publicly available at www.lungmap.net. At the moment, the project has collected ~20,000 images, 12,000 of which are publicly available. The data curator was concerned with managing thousands of images and related metadata so that they could be properly referenced in a publication but could not decide on a single hierarchy under which to organize them. Through modeling and using IDS, the data could be organized under the entities identified during data modeling (specimens, probes, or genes) for purposes of creating meaningful and manageable data subsets. Relationships between entities could be maintained, and researchers could choose to explore and provide DOIs to subsets based on any of the entities and their relationships.

The other test cases that served to generate requirements included genetic variation in rice (Duitama, 2015; Duitama et al., 2015), soil microbial community data from the National Ecological Observatory Network (Kao, Gibson, & Rachel, 2012), and the 1KP project, the first project to sequence 1000 plant genomes (Gitzendanner et al., 2018; Matasci et al., 2014).

Task 2: A Generic, Extensible Data Model

For IDS, we needed a data model that encompassed the lifecycle of genomic research data and specified the processes, material entities, and data involved at each stage. The data model serves as the basis for organizing data and metadata in all projects, so that users can track their data in a sensible manner, query for particular data files (e.g., which specimen was the source, or which process generated the output), and create custom subsets of the dataset. We began the project designing an all-encompassing genomics data model, but as we worked through our test cases, we realized that this model could be extended indefinitely with new processes and data types as the field of genomics grows and changes. Adding every possible research workflow variation to the model was not only beyond the scope of IDS, but was redundant with ontological efforts to describe biological data (Smith et al., 2007).

Our solution was to devise a generic data model to accommodate all our test cases and many more. Our generic data model (Fig. 1A) has three key components: 1) processual entities, which include things such as collecting specimens, carrying out assays, and analyzing data; 2) material entities such as physical specimens, reagents, and probes; and 3) data entities, including both individual data objects and datasets. We include a project entity as the umbrella under which the other entities are grouped. Processual entities link material and data entities in a graph, through input and output relations. Instances of such entities constitute different stages of the research process.

Using the generic model as a template, custom types of genomic projects can be represented based on the specific entities used during research and by their data management needs. Correspondence between the generic data model and the custom ones is shown for the maize methylation project (Fig. 1B) and for the HT-ISH project (Fig. 1C). The test case descriptions in Task 1 and the project-specific models drawn by the researchers are used to determine which entities need to be instantiated in IDS for which we will need to create metadata, and for which entities or groups of entities we need to provide identifiers. As illustrated in the figures, the positive evaluation of this task was achieved when we verified that the research processes of our use cases mapped to the generic model.

Task 3: IDS Web Interface and Architecture

We developed a web interface to support identity functions using Django to build the front end, the Agave APIs (Dooley, 2012) for web services, and a MySQL database to implement the generic data model and gather metadata (Fig. 2). Through the course of our project, we changed technologies and methods in response to research needs. Here, we report on the technologies that provided the most successful outcomes, noting that they may change in the future in lieu of new advancements. The Agave APIs provide a set of web services for managing and analyzing data, allowing users to register remote storage systems for storing and interacting with data, as well as HPC systems for remote data analysis. A system in Agave represents a server or a collection of servers that can be physical, virtual, or exposed through a single hostname or IP address. Systems are identified and referenced in Agave by a unique ID. The first iteration of our interface was used to implement the maize methylation test case and the second for the HT-ISH test case. Following are the functionalities we developed.

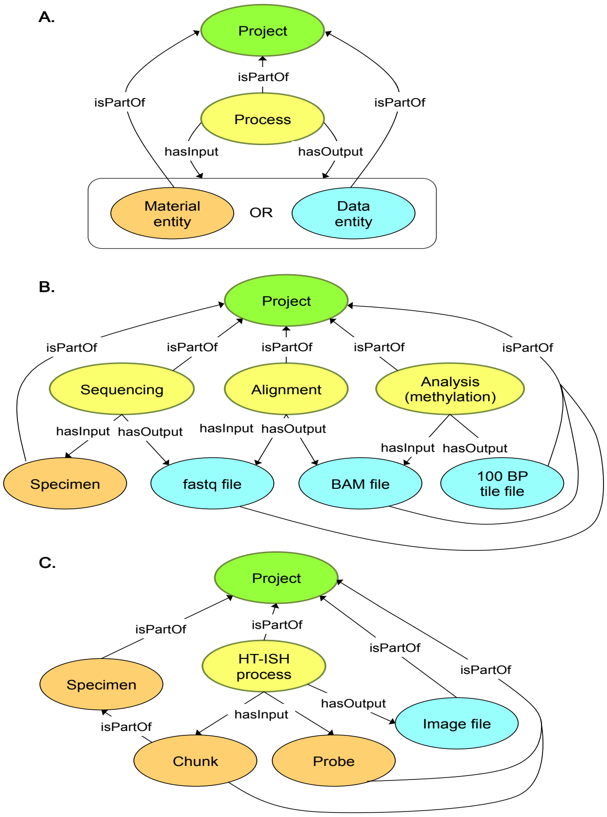


Figure. 1. ***(A)*** A generalized data model for Identifier Services (IDS) that describes the entities in a research project. Projects have as parts processes, which have either material or data entities as inputs and outputs. Actual research projects adapt this model to their specific needs. **(B)** For the maize methylation test case, a project consists of three processes (sequencing, alignment, and analysis), one type of material entity (specimens), and two types of data entities (BAM files and 100BP tile files). **(C)** For the high-throughput in situ hybridization (HT-ISH) test case, information is recorded about a single process type (HT-ISH), which has two types of material entities as input (chunks and probes) and one type of data entity as output (image files).

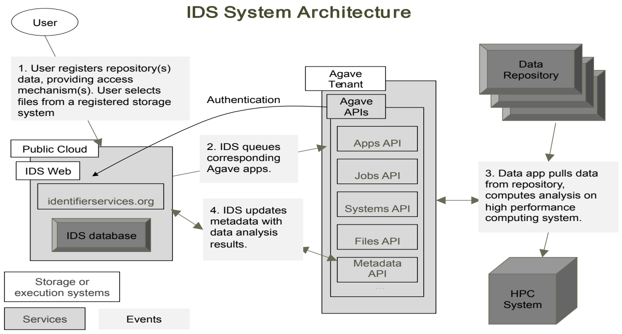

IDS System Architecture
User
1. User registers repository(s) data, providing access mechanism(s). User selects files from a registered storage system
Authentication
Agave Tenant
Agave APIs
Data Repository
Public Cloud
IDS Web
identifierservices.org
IDS database
2. IDS queues corresponding Agave apps.
Apps API
Jobs API
Systems API
Files API
Metadata API
...
3. Data app pulls data from repository, computes analysis on high performance computing system.
4. IDS updates metadata with data analysis results.
HPC System
Storage or execution systems
Services
Events

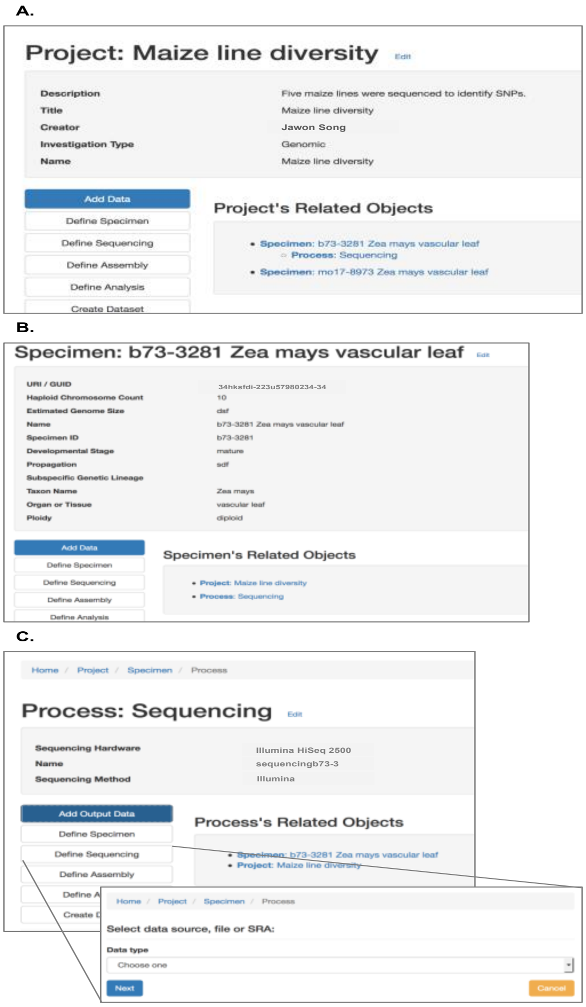
Figure. 2. IDS system architecture. Users interact with the IDS services via a web front end to register files and metadata. These interactions trigger actions by the Agave APIs such as fetching data from a repository and moving it to an execution system where a checksum is calculated. New metadata are pushed back to the IDS database by Agave and are visible to users via the web front end.

3.3 Project Creation and Data Model Input

In IDS, data are organized by project, and projects are defined by an investigation type based on their custom data models and metadata. Users create projects, which then become the project's landing page, and create one or select an existing investigation type. Our investigation type test cases included genomic sequencing, methylation, and imaging genomics. Investigation types can be reused by multiple projects, but each project is unique. For a project that defines the process of sequencing, with specimens as input and sequence data files as output, every unique specimen, sequencing process, and data file gets a UUID. The IDS database stores the UUIDs, metadata associated with them, and the relationships among entities based on the investigation type's data model. There are two ways for users to customize the generic data model. First, users can create a project interactively through the web interface and label the generic entities to correspond to the specific processes, materials, or data used in their research (Fig. 3A). As they label entities, they also define the metadata needed to describe them (e.g., specimen id, species, and collection location for specimens, Fig. 3B). To encode relationships among entities, users select what input is related to what output for each instance (Fig. 3C). This method is useful for relatively small projects like the Maize methylation project. The second way is for users to configure an investigation type by creating a YAML file (2009) that defines the entity types, their corresponding metadata, and relationships among entities (Supplemental Document 1). Users then upload the YAML file via the IDS portal (Fig. 4A). This method is useful for large-scale projects whose data model is fairly stable, as creating the YAML file requires some level of coding ability. Once a project is created, users can upload a single spreadsheet with all the metadata for the project as well as the locations of the data files (Fig. 4B). IDS then ingests the information into the database that automatically triggers Agave for registration of the data files (refer to the Data Registration section). This bulk registration and metadata upload method was used with the HT-ISH test case during which the curators gather detailed metadata about each image.

3.4 Data Registration

IDS does not store data but rather allows researchers to manage all their data from a central location. This is accomplished through registration, in which users supply the location of the data which may be stored at multiple locations including open repositories, cloud services, and data centers. IDS automatically fetches the files, calculates checksums in an HPC system, and associates each checksum with the UUID of the registered file. Checksums are presented in the user's dashboard, along with the project's metadata and in relation to the corresponding entity (Fig. 4D). For IDS to access files at locations that require authentication, users must first register the systems with Agave, so that IDS does not need to handle users' credentials for external systems. To work with Agave, storage systems must be accessible through one of several methods (grid ftp, sftp, scp, or ssh) and have open ports available for the IDS IP address. We tested this option with the Corral storage resource and the CyVerse Data Store. When data are available via a public download URL, IDS can register them without authentication. To demonstrate how files can be registered from repositories that serve data via their own APIs, we built an application that is executed using Agave to register data from NCBI's SRA. In the case of HT-ISH, bulk registration was useful to evaluate differences between manual data recordkeeping and what is actually stored. When registering a set of 239 images noted in the spreadsheet file provided by the curators, we verified that only 223 of the recorded images were on CyVerse BisQue, allowing to identify completeness and consistency between what is published in a repository and the researcher's recordkeeping system. This method works well but has the limitation that different code is needed to access each repository or storage system, limiting its generalizability and sustainability.



Project: Maize line diversity

Description	Five maize lines were sequenced to identify SNPs.
Title	Maize line diversity
Creator	Jawon Song
Investigation Type	Genomic
Name	Maize line diversity

Project's Related Objects

- Specimen: b73-3281 Zea mays vascular leaf
  - Process: Sequencing
- Specimen: mo17-8973 Zea mays vascular leaf

Specimen: b73-3281 Zea mays vascular leaf

URI / GUID	34hksfdi-223u57980234-34
Haploid Chromosome Count	10
Estimated Genome Size	dsf
Name	b73-3281 Zea mays vascular leaf
Specimen ID	b73-3281
Developmental Stage	mature
Propagation	sdf
Subspecific Genetic Lineage	
Taxon Name	Zea mays
Organ or Tissue	vascular leaf
Ploidy	diploid

Specimen's Related Objects

- Project: Maize line diversity
- Process: Sequencing

Process: Sequencing

Sequencing Hardware	Illumina HiSeq 2500
Name	sequencingb73-3
Sequencing Method	Illumina

Process's Related Objects

- Specimen: b73-3281 Zea mays vascular leaf
- Project: Maize line diversity

Select data source, file or SRA:

Data type: Choose one

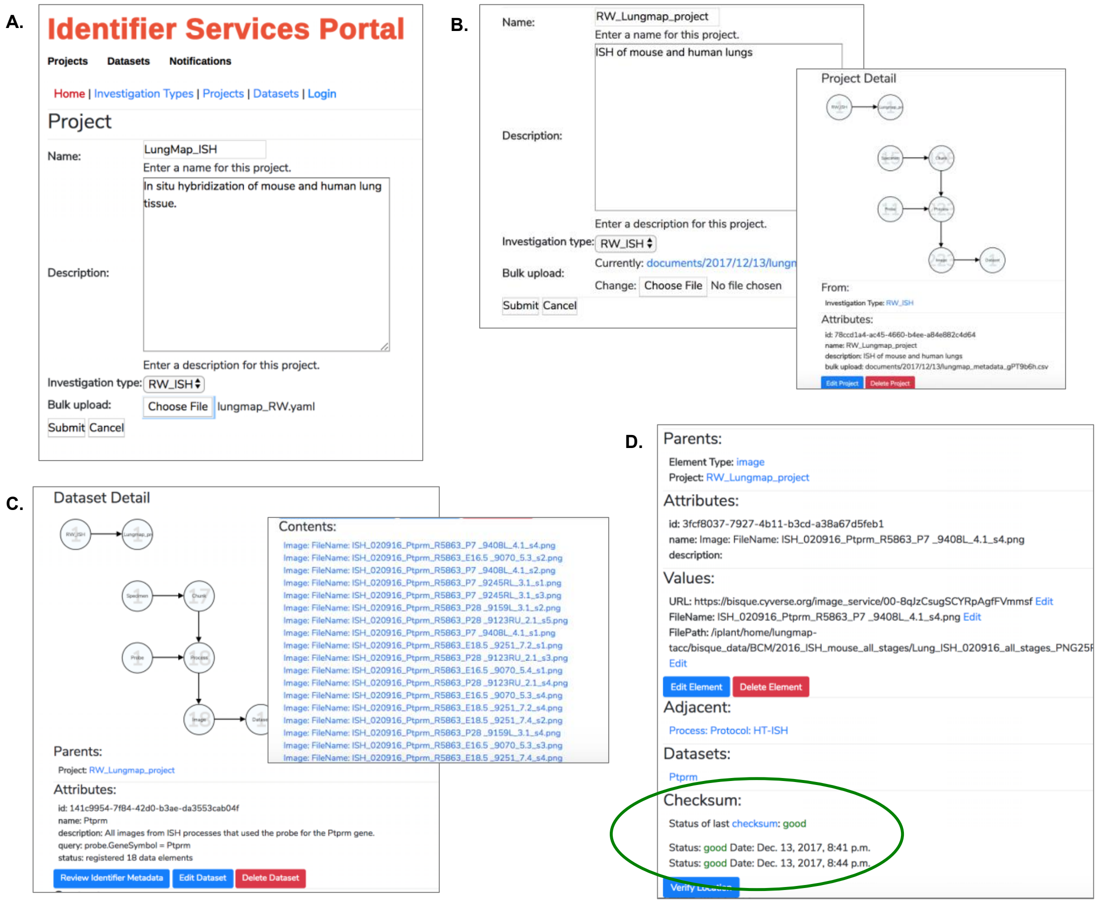
Figure. 3. Manual creation and entry of a project using a web interface. **(A)** Users can create a project directly in a web interface that is preconfigured to genomic sequencing projects. Users interactively label the entities as processes (i.e. sequencing, assembly, analysis), specimens, or data used in their research. **(B)** As they create labeled entities, they also define the metadata needed to describe these entities, in this case, corresponding to the standard metadata required for submission to National Center for Biotechnology Information (NCBI). The URI/GUID field in this example contains the UUID assigned by Identifier Services (IDS). **(C)** To encode relationships among entities, users manually select what input is related to what output for each instance. In this case, the sequencing process has as output a FASTQ file from the NCBI Sequence Read Archive (SRA).

3.5 Model-based Queries, Dataset Creation, and Publication

We developed a virtual project configuration method that allows users to create subsets of data in a project based on their defined data model and metadata values. Having provided the metadata as explained in data registration, users can query the IDS database in the interface by any of the entity labels and or metadata fields in order to decide what a "subset" is. Because the relationships between entities and metadata expressed in the custom data model are maintained, the query result can be graphically represented as multirelational. Information about all the files in the subset is also provided. Once a subset is created, the user can request a DOI for it through an automated pipeline using the EZID API (n.d.). Metadata for the identifier (e.g., title, creator, date) are automatically pulled from the project description. The user then verifies or edits the metadata, and the subset receives a DOI, which is printed on the landing page. This virtual project configuration exists in IDS and may or may not reflect the organization of the files on the system where they are stored. Because files are registered and their location is established, IDS can use Agave to conduct any of the needed identity functions on the actual, remotely stored files. Our evaluation with the HT-ISH test case allowed us to build different data subsets. This is particularly useful for large datasets whose structure and relationships can be identified through the graphical representation. For example, we queried for data containing all image files that were derived from a probe for a certain gene to create a set and queried for all files from a single specimen. Results of the queries and the resultant representation were successful as exemplified in Fig. 4C.

Tasks 4 and 5: Implement Automated Location and Integrity Functions

During research and beyond, data may be moved from one storage resource to another, resources may be decommissioned, data content may be changed intentionally or not without notifying all team members, and data may get corrupted. We developed functions to check that registered data are where they were expected, and to calculate checksums over time and report back to IDS for comparison of results. In digital repositories where files are not supposed to change, checksums are used at ingest to establish the integrity of the transfer and the authenticity of a file. In IDS, this function is powered by an Agave app that fetches the file/s, calculates checksums in the HPC resource Wrangler (Jordan et al., 2015), and adds the new checksum and date verified to the file's metadata in the database and on the project's landing page (Fig. 4D). If the file is not present at its registered location or the checksum does not match the previous one, a warning is provided to the user who will have to resolve the origin of the inconsistency. Ideally, this service should be scheduled to run at regular intervals and produce a continuous report. Depending on the



A. Project

Name: LungMap_ISH
Enter a name for this project.

Description: In situ hybridization of mouse and human lung tissue.
Enter a description for this project.

Investigation type: RW_ISH

Bulk upload: lungmap_RW.yaml

B.

Name: RW_Lungmap_project
Enter a name for this project.

Description: ISH of mouse and human lungs
Enter a description for this project.

Investigation type: RW_ISH

Bulk upload: Currently: documents/2017/12/13/lungm
Change: No file chosen

Project Detail

From:
Investigation Type: RW_ISH

Attributes:
id: 78ccd1a4-ac45-4660-b4ee-a84e882c4d64
name: RW_Lungmap_project
description: ISH of mouse and human lungs
bulk upload: documents/2017/12/13/lungmap_metadata_gPT9b6h.csv

C. Dataset Detail

Contents:
Image: FileName: ISH_020916_Ptprm_R5863_P7 _9408L_4.1_s4.png
Image: FileName: ISH_020916_Ptprm_R5863_E16.5 _9070_5.3_s2.png
Image: FileName: ISH_020916_Ptprm_R5863_P7 _9408L_4.1_s2.png
Image: FileName: ISH_020916_Ptprm_R5863_P7 _9245RL_3.1_s1.png
Image: FileName: ISH_020916_Ptprm_R5863_P7 _9245RL_3.1_s3.png
Image: FileName: ISH_020916_Ptprm_R5863_P28 _9159L_3.1_s2.png
Image: FileName: ISH_020916_Ptprm_R5863_P28 _9123RU_2.1_s5.png
Image: FileName: ISH_020916_Ptprm_R5863_P7 _9408L_4.1_s1.png
Image: FileName: ISH_020916_Ptprm_R5863_E18.5 _9251_7.2_s1.png
Image: FileName: ISH_020916_Ptprm_R5863_P28 _9123RU_2.1_s3.png
Image: FileName: ISH_020916_Ptprm_R5863_E16.5 _9070_5.4_s1.png
Image: FileName: ISH_020916_Ptprm_R5863_P28 _9123RU_2.1_s4.png
Image: FileName: ISH_020916_Ptprm_R5863_E16.5 _9070_5.3_s4.png
Image: FileName: ISH_020916_Ptprm_R5863_E18.5 _9251_7.2_s4.png
Image: FileName: ISH_020916_Ptprm_R5863_E18.5 _9251_7.4_s2.png
Image: FileName: ISH_020916_Ptprm_R5863_P28 _9159L_3.1_s4.png
Image: FileName: ISH_020916_Ptprm_R5863_E16.5 _9070_5.3_s3.png
Image: FileName: ISH_020916_Ptprm_R5863_E18.5 _9251_7.4_s4.png

Parents:
Project: RW_Lungmap_project

Attributes:
id: 141c9954-7f84-42d0-b3ae-da3553cab04f
name: Ptprm
description: All images from ISH processes that used the probe for the Ptprm gene.
query: probe.GeneSymbol = Ptprm
status: registered 18 data elements

D.

Parents:
Element Type: image
Project: RW_Lungmap_project

Attributes:
id: 3fcf8037-7927-4b11-b3cd-a38a67d5feb1
name: Image: FileName: ISH_020916_Ptprm_R5863_P7 _9408L_4.1_s4.png
description:

Values:
URL: https://bisque.cyverse.org/image_service/00-8qJzCsugSCYRpAgfFVmmsf Edit
FileName: ISH_020916_Ptprm_R5863_P7 _9408L_4.1_s4.png Edit
FilePath: /iplant/home/lungmap-tacc/bisque_data/BCM/2016_ISH_mouse_all_stages/Lung_ISH_020916_all_stages_PNG25F
Edit

Adjacent:
Process: Protocol: HT-ISH

Datasets:
Ptprm

Checksum:
Status of last checksum: good

Status: good Date: Dec. 13, 2017, 8:41 p.m.
Status: good Date: Dec. 13, 2017, 8:44 p.m.

Figure 4. Bulk project creation and metadata registration. **(A)** Users configure an investigation by uploading a YAML file that defines the entity types, their corresponding metadata, and relationships among entities. **(B)** Projects are created based on the investigation type. Users upload a single spreadsheet with all metadata for the project as well as the locations of the data files. Identifier Services (IDS) then ingests the information into the database and automatically triggers registration of the data files. The IDS web portal then displays the project data in a graph based on the specified data model (inset). **(C)** Using information in the model and database, users can create datasets via query. In this case, the query is searched for all images derived from the probe for gene Ptprm. This query yields a dataset of 18 files, structured according to the data model. **(D)** Users can view the details of any file in a project and do location/integrity checks. The resulting checksums are displayed (green oval), and if they do not match, or the file is not where it is supposed to be, an error is reported.

system that serves the files, size and number of files, and whether computations are done sequentially, checksum calculations can be network and throughput intensive. A test with 223 images took 15 minutes to check locations, compute checksums, and register files serially. When the task is properly distributed to many nodes (parallelized), completion time can be reduced to one-tenth of the serial execution time. Future implementations should also allow checksum comparisons of more than two copies of the same file in multiple locations. These functions are complemented by task 6.

Task 6: Implement a Data Content Comparison Function

This task considers differential content as an identity attribute. While a checksum comparison can determine if two files stored at different locations are identical, it cannot determine that two nonidentical files have similar content. For example, many genomic researchers will create a genome assembly file and store it locally. When they publish their sequence data to SRA, it takes the raw reads and reassembles the genome according to its own pipeline, so the local copies and the public file will not have the same checksum. Still, for all purposes, these are two instances of a same work. Likewise, two collaborators in a project may each have a copy of a file, but one would have added some additional header information, leaving the rest of the content intact. Losing track of files and of their changes is very common when working in distributed environments and large datasets, and using the sequence comparison tool Blast for verification is inefficient. We developed a workflow for content comparison of genomic sequence data, described in (Xu et al., 2016), and applied it to three common scenarios found in genomics data management. These include: a) copies of a dataset stored in different repositories with metadata and content discrepancies, b) two seemingly identical components of a dataset stored and published that show content differences (this was tested with the maize methylation sequencing files), and c) two published datasets with similar metadata that show significant content differences. In all cases, researchers were not sure of why and what had changed, and the existing metadata for some of the files were not enough to make informed decisions.

Our algorithm detects content differences in sequencing files, first by comparing the unique identifiers that precede each sequence and then the sequences themselves. Each of these content elements is considered an identity attribute. The method allows identifying discrepancies between the files stored at different locations. Because content-based comparisons can be computationally expensive, an important requirement of this function is scalability. We used the Agave API to transfer data to the Wrangler HPC resource. To provide a point of reference, comparing a pair of files of ~3 GB each including up to ~19,000 pairs of identifiers and sequencing records using this method took less than 2 minutes (2016).[2]

An important goal of this task was to convey information for users to understand the nature of the discrepancies. This information will help them make decisions about the files' provenance, enhance metadata, and decide how to assign DOIs. However, there are challenges in reporting the results of large-scale comparisons, as the amount of differences identified by the algorithm can be of the same order (tens of thousands) of the sequencing records being compared. After consulting with our collaborators, we decided on a report consisting of three layers of information including a) statistical summaries of the results, b) examples of pairs of sequencing records randomly sampled from each compared file, and c) the complete copy of the results organized by identical sequencing records, non-identical ones, and records missing in one of the files. With this information, users were able to infer the reasons for the differences and whether the files could be considered the same work or significantly different. For example, the comparison of two maize methylation sequencing files, one stored in Corral and one in SRA, showed that by mistake an untrimmed file was submitted to SRA; the researchers considered that the difference between both was not significant. Comparisons can be repeated over time to continuously manage the identity of evolving data.

Task 7: Evaluate IDS from the User's Perspectives

We used the test cases and a focus group to evaluate researchers' understanding of identity in lifecycle data management, how useful the functions developed by IDS were to them, and to estimate whether or not they would adopt them. Researchers affiliated to each test case guided us throughout development by providing extensive feedback on what was needed to support their research and by testing. In the Maize methylation test case, the team was able to attain a complete record of the dataset and its current identity status, including the files stored in Corral and the sequencing files at SRA. Creating meaningful subsets based on multiple relevant identity attributes was the main takeaway for the HT-ISH project curator.

Before the end of the project, we conducted a three-segment focus group with five early career biologists. In the first segment, we asked the researchers to draw their data workflows to verify that they would be able to customize a generic data model in IDS. Next, we demonstrated IDS identity functions corresponding to tasks 2, 3, 4, and 5. Task 6 was presented but not demonstrated. Finally, we conducted a semi-structured discussion to assess how well users grasped the utility of the data model and the identity functions.

Following the modeling method described in Task 2, we asked researchers to draw their workflows, which they completed in the allotted time. After we demonstrated IDS, all saw the utility of modeling their workflows to customize the generic data model for organizing, describing, relating, querying, and sub setting their own data. All participants appreciated the services for checking identity across time, especially those facing the challenges of managing distributed datasets across many collaborators. The same researchers could see the immediate utility of the checksum and content comparison methods. All researchers became aware of the work involved in maintaining metadata throughout the project, which many do not experience until they prepare their data for publication. Although it was not the original purpose of IDS, researchers wanted to see the services integrated with workflow managers. This suggests that IDS might be more adopted if integrated not only with repositories that publish data but with infrastructure used for analyzing data (e.g., CyVerse or Galaxy).

[2] The results of this task were published as a conference proceeding at **DOI:** 10.1109/BigData.2016.7840987. Readers can refer to this publication for complete narrative, results, and evaluation.

There were also some points of confusion. For researchers with less bioinformatics experience, it was at first difficult to grasp that IDS does not store data but rather metadata, while the data remain in various remote locations. Other doubts were at what point of a project lifecycle IDS would be used, and how projects and data would be versioned over time.[3] Assessment of the overall value added by IDS varied depending on each researcher's project. Scientists working on small data projects saw little advantage over their current method of managing data on their personal computers, whereas researchers working on big distributed data projects with extensive metadata felt that the IDS services would adequately make up for any additional effort involved.

4 Conclusions

Current infrastructure for life sciences data management does not fully support the data lifecycle. In general, scientists manage active data as part of their research, and once they publish it, they lose control of their dataset within an institutional repository. From the other side, institutional repositories just manage a static copy of data and do not account for evolution of the project outside the repository boundaries. As a result of these contrasting practices, data often remain siloed and unaccessible – not FAIR. The goal of this research was to experiment with identity functions to manage the full lifecycle of genomics data. For this, we researched the conceptual and technical state of the art of scalable identity functions through a prototype and using real genomic data test cases. We also evaluated users' perspectives of such functions through a focus group. In the process, we found solutions to challenges and identified gaps that must be overcome in order to advance in the space.

The difficulty of managing datasets that contain many hundreds to thousands of files is a problem that researchers and data managers are very concerned with. Traditional hierarchical folder structures and meaningful file names are too fragile for managing large, distributed data collections, and applying metadata to many files can be extremely tedious. The IDS generic but flexible data model combined with bulk registration of data and metadata allows more automatic and thus efficient management of large datasets as the first step to continuously track their evolution over time through identity functions. In the process of developing these functions, we learned that we needed better graphical interfaces for users to interact with larger datasets. This led to prototyping functions to query, subset, and represent complex relationships between dataset components. Further research and design are required to ease the creation of custom data models and facilitate data management and understandability via interfaces as well as testing the user experience.

A key feature of this project is that data remain distributed across many locations and that teams working remotely can manage them using IDS. In order to perform identity tasks such as content comparison or checksum calculations in bulk, the data have to be moved to an execution system with large computing and throughput capacity. Bandwidth remains a significant limitation. To transition IDS to production, we need to test grid/distributed transfer protocols within the storage/repositories, IDS, and Agave. In addition, we can work toward scaling through further parallelizing the tasks. Scheduling remote open-science and shared HPC resources where computing takes place will need to be adjusted to scale workflows and improve run times.[4]

To perform identity functions on data stored in repositories, IDS must use the web services that are unique to each repository. Access to data by IDS is problematic for many repositories, a few of which provide a direct link to data files. While the security concerns associated with providing direct anonymous access to data are understood, current services inhibit distributed data management and access. A set of shared APIs that let repositories register as storage systems with a service like Agave would diminish this problem.

Through IDS, we gained insight into what identity means for researches and its potential for data management. We realize that projects are idiosyncratic and that it is difficult to generalize modes of identity functionality. Providing the opportunity to customize a generic data model is a method to sort this difficulty. To go beyond the notion of identifying individual files, something that is practiced consistently by biology researchers, we tested applying labels to all entities in a project to indicate the provenance of the different dataset components. We learned that identifiers gain and lose functionality over time in a project, and it is rare that a single type will serve over the entire lifecycle. Following archival principles, we strengthen the concept of identity

[3] We did not prototype versioning as many repository systems have that problem already solved. See the DataVerse Project at https://dataverse.org/.

[4] Use of open HPC resources implies setting up allocations and adapting to specific runtimes that allow many users to share the resources efficiently.

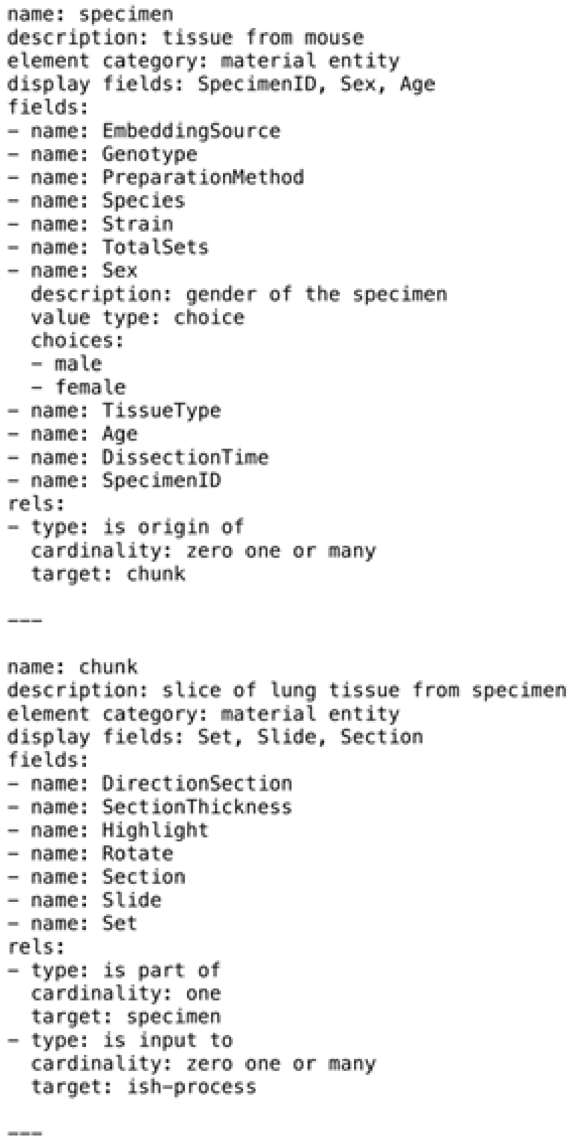
by using a combination of identity attributes and by offering the possibility to assign DOIs.

Finally, the ability to constantly update the metadata, regardless of where data are located, is a major shortcoming of current infrastructure. For example, it is difficult to link new derived datasets or correct metadata and having the corrections propagate to related files within their respective systems. The need for mechanisms that can push and pull metadata before, during, and after publication, while controlling for different levels of trust in relation to systems where data are published, is a key gap. In our research, we envisioned a system that allows multiple agents managing open data in a distributed environment. This necessarily entails opening gates, as well as coordination and trust, indispensable to postcustodialism. Identity functions take care of tracking evolving data in a way that assures their reliability and continuous maintenance of provenance. The reality of open, interoperable data cannot come to fruition until some of the technical mechanisms we manage are better resolved. Our expectation is that the outcomes of this project will directly inform the development of CI supporting distributed data management across scientific domains.

Acknowledgments: This research was funded by the NSF EAGER Collaborative Research (No. 1555458). The authors acknowledge the TACC at The University of Texas at Austin (URL: http://www.tacc.utexas.edu) and CyVerse (NSF DBI-0735191, DBI-1265383, URL: http://www.cyverse.org/) for providing HPC, database, and cloud resources that have contributed to the results reported within this paper.

Supplemental Documents

Supplemental Document 1. YAML file used to configure the IDS investigation type for the high-throughput in situ hybridization (HT-ISH) test case. The configuration file defines types of entities used in the investigation (specimen, chunk, probe, ISH process, and image), how they map to entities in the generic data model, metadata elements for each entity, and relations among entities, including cardinality of relations. Value types and choices of values may be defined for metadata attributes. Which metadata attributes to display in the portal is configurable.



```
name: specimen
description: tissue from mouse
element category: material entity
display fields: SpecimenID, Sex, Age
fields:
- name: EmbeddingSource
- name: Genotype
- name: PreparationMethod
- name: Species
- name: Strain
- name: TotalSets
- name: Sex
  description: gender of the specimen
  value type: choice
  choices:
  - male
  - female
- name: TissueType
- name: Age
- name: DissectionTime
- name: SpecimenID
rels:
- type: is origin of
  cardinality: zero one or many
  target: chunk

---

name: chunk
description: slice of lung tissue from specimen
element category: material entity
display fields: Set, Slide, Section
fields:
- name: DirectionSection
- name: SectionThickness
- name: Highlight
- name: Rotate
- name: Section
- name: Slide
- name: Set
rels:
- type: is part of
  cardinality: one
  target: specimen
- type: is input to
  cardinality: zero one or many
  target: ish-process

---
```

References

Afgan, E., Baker, D., van den Beek, M., Blankenberg, D., Bouvier, D., Čech, M., Goecks, J. (2016). The Galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2016 update. *Nucleic Acids Research,* 44(W1), W3–W10.

Ardini-Poleske, Maryanne E., et al. (2017, August). LungMAP: The Molecular Atlas of Lung Development Program. *American Journal of Physiology-Lung Cellular and Molecular Physiology,* 313(5), pp. L733–40. physiology.org (Atypon),

Bionetworks. (n, d). Website. Retrieved from https://www.synapse.org/

Corral - Texas Advanced Computing Center. (n.d.). Retrieved from https://www.tacc.utexas.edu/systems/corral

COPO - Earlham Institute Documentation. (2015). Retrieved from https://documentation.tgac.ac.uk/display/COPO/Overview

DataONE, Data Packaging — v2.0.1. (2015). Retrieved from https://releases.dataone.org/online/api-documentation-v2.0/design/DataPackage.html

DCC Curation Lifecycle Model | Digital Curation Centre. (2019, January 24). Retrieved from http://www.dcc.ac.uk/resources/curation-lifecycle-model

Dooley, R., Vaughn, M. Stanzione, D., Terry,S., Skidmore.E,. (2012). Software-as-a-Service: The iPlant Foundation API, *5th IEEE Workshop on Many-Task Computing on Grids and Supercomputers (MTAGS)*. IEEE, 2012.

Duitama, J. (2015). Genomic variation for 104 rice elite cultivars, landraces and wild relatives [Data set].

Duitama, J., Silva, A., Sanabria, Y., Cruz, D. F., Quintero, C., Ballen, C., … Tohme, J. (2015). Whole genome sequencing of elite rice cultivars as a comprehensive information resource for marker assisted selection. *PloS One,* 10(4), e0124617.

Duraspace. (2016). Retrieved from https://github.com/duraspace/pcdm

EZID (n.d.). Retrieved from https://ezid.cdlib.org/

Factor, M. Henis, E., Naor, D. Rabinovici-Cohen, S., Reshef, P., Ronen, S. Michetti, G., and Guercio, M. (2009). Authenticity and provenance in long term digital preservation: modeling and implementation in preservation aware storage. In *First Workshop on Theory and Practice of Provenance (TAPP'09)*. USENIX Association, Berkeley, CA, USA, Article 6.

Fedora Digital Object Model - Fedora 3.8 Documentation - DuraSpace Wiki. (2019, February 6) Retrieved from https://wiki.duraspace.org/display/FEDORA38/Fedora+Digital+Object+Model.

Gary King. (2007). An Introduction to the Dataverse Network as an Infrastructure for Data Sharing. *Sociological Methods and Research,* 36, pp. 173–199. Retrieved from http://j.mp/2owjuRr

Gitzendanner, M. A., Soltis, P. S., Wong, G. K.-S., Ruhfel, B. R., & Soltis, D. E. (2018). Plastid phylogenomic analysis of green plants: A billion years of evolutionary history. *American Journal of Botany,* 105(3), 291–301.

Gray, J., Liu, D. T., Nieto-Santisteban, M., Szalay, A., DeWitt, D. D., & Heber, G. (2005). Scientific data management in the coming decade. *SIGMOD Record,* 34(4), 34–41.

Guercio, M. (2001). Principles, methods, and instruments for the creation, preservation, and use of archival records in the digital environment. *The American Archivist,* 64(2), 238–269.

Haller, A., Janowicz, K., Cox, S. J. D., Lefrançois, M., Taylor, K., Le Phuoc, D., … Stadler, C. (2018). The modular SSN ontology: A joint W3C and OGC standard specifying the semantics of sensors, observations, sampling, and actuation. *Semantic Web,* 10(1), 9-32.

Henry, L. (1998). Schellenberg in Cyberspace. *The American Archivist*, 61(2), 309–327.

Hirtle, P.B (2000). Archival Authenticity in a Digital Age. In C. T. Cullen, P.B. Hirtle, D. Levy, C.A. Lynch (Ed.), *Authenticity in a Digital Environment* (pp. 8–23). Washington, D.C: Council on Library and Information Resources.

Hoogerwerf, M., Losch, M., Schirrwagen, J., Callaghan, S. Manghi, P., Iatropoulou, K., Keramida D., Rettberg, N. (2019) Linking data and publications: towards a cross disciplinary approach. *International Journal of Digital Curation.*

International Nucleotide Sequence Database Collaboration | INSDC. (2018). Retrieved from www.insdc.org/

Jordan, C., Walling, D., Xu, W., Mock, S.A., Gaffney, N. Stanzione, D. (2015). Wrangler's user environment: A software framework for management of data-intensive computing system, In *Big Data (Big Data), 2015 IEEE International Conference on,* 2015, pp. 2479–2486.

Kao, R. H., Gibson, C. M., Gallery, R. E., Meier, C. L., Barnett, D. T., Docherty, K. M., & Thibault, K. M. (2012). NEON terrestrial field observations: designing continental-scale, standardized sampling. *Ecosphere,* 3(12), art115.

Kvilekval, K., Fedorov, D., Obara, B., Singh, A., & Manjunath, B. S. (2010). Bisque: a platform for bioimage analysis and management. *Bioinformatics,* 26(4), 544–552.

Li, Q., Song, J., West, P. T., Zynda, G., Eichten, S. R., Vaughn, M. W., & Springer, N. M. (2015). Examining the Causes and Consequences of Context-Specific Differential DNA Methylation in Maize. *Plant Physiology,* 168(4), 1262–1274.

Lynch, C. A. (2000). Authenticity and Integrity in the Digital Environment: An Exploratory Analysis of the Central Role of Trust. In C.T. Cullen, P.B. Hirtle, D. Levy, C.A. Lynch (Ed.), *Authenticity in a Digital Environment* (pp. 32–50). Washington, DC: Council on Library and Information Resources.

Madin, J., Bowers, S., Schildhauer, M., Krivov, S., Pennington, D., & Villa, F. (2007). An ontology for describing and synthesizing ecological observation data. *Ecological Informatics,* 2(3), 279–296.

Matasci, N., Hung, L.-H., Yan, Z., Carpenter, E. J., Wickett, N. J., Mirarab, S., … Wong, G. K.-S. (2014). Data access for the 1,000 Plants (1KP) project. *GigaScience,* 3(1), 17.

Merchant, N., Lyons, E., Goff, S., Vaughn, M., Ware, D., Micklos, D., & Antin, P. (2016). The iPlant Collaborative: Cyberinfrastructure for Enabling Data to Discovery for the Life Sciences. *PLoS Biology,* 14(1), e1002342.

National Center for Biotechnology Information. (2018, September 4). Retrieved from https://www.ncbi.nlm.nih.gov/

Nelson, J., & Peterson, L. (2014). Syndicate: Virtual cloud storage through provider composition. In *BigSystem 2014 - Proceedings of the 2014 ACM International Workshop on Software-Defined Ecosystems, Co-located with HPDC 2014* (pp. 1–8). Association for Computing Machinery, Vancouver, BC, Canada.

ORE User Guide - Resource Map Overview. (2014). Retrieved from http://www.openarchives.org/ore/1.0/toc

OSF. (n, d). Retrieved from https://osf.io/

Phillips, S., Koenig, J. (2008) DSpace Administration AndUse. Retrieved from https://www.tdl.org/wp- content/uploads/2009/04/DSpaceAdministrationAndUse.pdf

PROV-DM: The PROV Data Model. (2013). Retrieved from https://www.w3.org/TR/prov-dm/

PROV-O: The PROV Ontology. (2013). Retrieved from https://www.w3.org/TR/prov- o/

Rathje, E. M., Dawson, C., Padgett, J. E., Pinelli, J.-P., Stanzione, D., Adair, A., … Mosqueda, G. (2017). DesignSafe: new cyberinfrastructure for natural hazards engineering. *Natural Hazards Review,* 18(3), 06017001.

Ray, J., (2012). The rise of digital curation and cyberinfrastructure: from experimentation to implementation and maybe integration, *Library Hi Tech,* 30(4), pp.604-622, https://doi.org/10.1108/07378831211285086

Semantic-Observations. (2016). Retrieved from https://github.com/Semantic-Observations.

Smith, B., Ashburner, M., Rosse, C., Bard, J., Bug, W., Ceusters, W., … Lewis, S. (2007). The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration. *Nature Biotechnology,* 25(11), 1251–1255.

Springer, N. (2017a). B73 WGBS [Data set]. https://doi.org/10.7946/P2SG6Z

Springer, N. (2017b). CML322 WGBS [Data set]. https://doi.org/10.7946/P2NP41

Springer, N. (2017c). Mo17 WGBS [Data set]. https://doi.org/10.7946/P2J012

Springer, N. (2017d). Oh43 WGBS [Data set]. https://doi.org/10.7946/P2D59K

Springer, N. (2017e). Tx303 WGBS [Data set]. https://doi.org/10.7946/P28G69

SRA - NCBI. (n.d.). Retrieved from https://www.ncbi.nlm.nih.gov/sra

Texas Advanced Computing Center. (2018, September 4). Retrieved from https://www.tacc.utexas.edu

Vision, T. (2010). The Dryad Digital Repository: Published evolutionary data as part of the greater data ecosystem, (713). https://doi.org/10.1038/npre.2010.4595.1

Walls, R. L., Deck, J., Guralnick, R., Baskauf, S., Beaman, R., Blum, S., ... Wooley, J. (2014). Semantics in Support of Biodiversity Knowledge Discovery: An Introduction to the Biological Collections Ontology and Related Ontologies. *PloS One, 9*(3), e89606.

West, M. (2011). *Developing High Quality Data Models*. Burlington, USA: Morgan Kaufmann Pub.

Wilkinson, M. D., Dumontier, M., Aalbersberg, I. J., Appleton, G., Axton, M., Baak, A., ... Mons, B. (2016). The FAIR Guiding Principles for scientific data management and stewardship. *Scientific Data,* 3, 160018.

Wynholds, L. (2011). Linking to scientific data: identity problems of unruly and poorly bounded digital objects. *International Journal of Digital Curation,* 6(1), 214–225.Xu, W., Huang, R., Esteva, M., Song, J., Walls, R. (2016). Content-based comparison for collections identification. In *2016 IEEE International Conference on Big Data (Big Data)* (pp. 3283–3289). DOI: 10.1109/BigData.2016.7840987

YAML Ain't Markup Language (YAML™) Version 1.2. (2009). Retrieved from http://yaml.org/spec/1.2/spec.html