Globus Data Publication as a Service: Lowering Barriers to Reproducible Science

Kyle Chard, Jim Pruyne, Ben Blaiszik, Rachana Ananthakrishnan,
Steven Tuecke, Ian Foster
Computation Institute, University of Chicago and Argonne National Laboratory
Chicago, Illinois, USA

***Abstract*—Broad access to the data on which scientific results are based is essential for verification, reproducibility, and extension. Scholarly publication has long been the means to this end. But as data volumes grow, new methods beyond traditional publications are needed for communicating, discovering, and accessing scientific data. We describe data publication capabilities within the Globus research data management service, which supports publication of large datasets, with customizable policies for different institutions and researchers; the ability to publish data directly from both locally owned storage and cloud storage; extensible metadata that can be customized to describe specific attributes of different research domains; flexible publication and curation workflows that can be easily tailored to meet institutional requirements; and public and restricted collections that give complete control over who may access published data. We describe the architecture and implementation of these new capabilities and review early results from pilot projects involving nine research communities that span a range of data sizes, data types, disciplines, and publication policies.**

I. Introduction

The scientific method is based on a model of publication, reproducibility, verification, and extension. As research becomes increasingly data-intensive, researchers face new challenges in supporting this process in the era of "Big Data." We have well-established practices for text-based publications. Researchers can avail themselves of thousands of publication options (journals, conference proceedings, self-service repositories), each implementing processes that provide immutability, summarization, indexing, and persistent identifiers. However, few such systems support the publication of data distinct from text, and fewer still the publication of large datasets. Thus, many researchers lack viable methods for publishing data.

Many studies (e.g., [1], [2]) have shown that current research data management and publication practices are inadequate. These studies highlight challenges at every level of the data publication process and note that data is infrequently published, linkages between text-based publications and their data are rarely defined, and original datasets often cannot be obtained or located. This situation undermines core principles of scientific discovery: that research is reproducible and verifiable. For these reasons, much attention has been given, by researchers, funding agencies, research institutions, and publishers to the need for better data sharing and publication practices. Funding agencies and publishers have responded by placing strict requirements on data availability associated with grants and publications, although these requirements are often not met [2]. Even when researchers do publish data, it is often published in an *ad hoc* manner, making data discovery and contextualization challenging (due to a lack of metadata). Further, these solutions rarely make guarantees of persistent data availability, data completeness, or data immutability.

We have developed new data publication capabilities to address these challenges. These capabilities support arbitrarily large datasets, customizable publication pipelines and policies, and a software-as-a-service (SaaS) delivery model that simplifies user adoption and service use [3]. Institutions, publishers, and researchers can outsource to this system the difficult aspects of managing the publication process, including data submission, description, assembly, curation, and discovery. These capabilities form part of the Globus research data management service [4], which is professionally developed, hosted, and managed by an experienced team at the University of Chicago. Its implementation applies proven methods to achieve high availability, including a distributed, stateless, and replicated implementation deployed on elastic cloud computing infrastructure, continuous monitoring, and automated responses to many transient failures.

Our data publication capabilities provide a self-service interface through which users are able to define "collections" with associated policies regarding data storage, submission workflows, metadata formats, persistent identifier providers, and delegated group-based role assignment. We leverage Globus data access and sharing methods [5] to provide a novel "bring your own" storage model through which collections may use distributed storage (e.g., institutional or cloud object storage) to store published datasets. Our approach enables publication of data no matter its size or complexity, makes it easy to associate a wide variety of citable, persistent identifiers (e.g., DOIs) with data, provides guarantees regarding data immutability, and enables linkage with other publications. These capabilities are provided via human- and machine-accessible interfaces for discovering and accessing published data, at scales that correspond to the growth of Big Data.

The rest of this paper is as follows. Section II describes related work, Section III outlines requirements, and Section IV presents the SaaS-based Globus capabilities. Section V describes experiences in nine pilot communities. Finally, Section VI summarizes and outlines next steps.

II. Related Work

Many studies have identified requirements and principles for data publication, data citation, the linking of data publications with other elements of the scientific record [6], [7], [8].Space does not permit a detailed review of this literature.

A tremendous variety of systems have been developed or proposed to support various forms of data publication. For example, highly curated and carefully validated materials properties databases date back to the 1970s [9], and materials databases such as NIST's standard reference data are relied upon by many thousands of materials researchers. In biomedicine, dbGaP (database of Genotypes and Phenotypes) [10] organizes the results of studies of genotype and phenotype interactions, while the Universal Protein Resource (Uniprot) [11] contains protein sequence and functional information. The SIMBAD [12] astronomical database provides data and measurements for astronomical objects outside the solar system. Our work is differentiated from such structured databases by its focus on providing a general solution for creating, organizing, naming, and preserving arbitrary datasets, rather than aggregating data of a specific type.

Raw data derived from experimentation, simulation, and computation, while often more voluminous than data stored in curated databases, can also have considerable value, and indeed are often essential for reproducible science. There are many examples of such data repositories in various fields. For example, the National Database for Autism Research [13] and the Federal Interagency Traumatic Brain Injury (TBI) Research [14] informatics system store large autism and brain imaging datasets, respectively. NOAA's National Climatic Data Center [15] provides public access to national climate and historical weather data. The Collaborative Chemistry Database Tool [16] provides a model for storing large amounts of computational chemistry raw data. The Materials Atlas [17] is a repository for 3D experiments and simulations on material systems, organized by technique. None of these repositories provides a general-purpose solution for publishing arbitrary scientific data, as each is designed for a single domain. Some require installation of custom software to access the data and many require upload of data to the central repository server.

Other domains, in particular library sciences, have invested significant effort in general-purpose data publication solutions. Some institutions have developed their own repositories, such as the Purdue University Research Repository [18], the Data Repository for the University of Minnesota [19], and Pennsylvania State ScholarSphere [20]. These systems allow researchers from *any* domain to create and publish datasets. However, as such systems are operated by individual institutions, they typically support only institutional users and rely on institutional storage, thus negating the potential for easily publishing data within inter-institutional collaborations or the wider scientific community.

General-purpose data publication systems are also emerging, such as Dataverse [21], figshare [22], Zenodo [23], and Dryad [24]. These systems provide generic repositories, either offered as cloud-hosted services or in some cases (e.g., Dataverse) deployed and hosted by the institution. Each implements some notion of a dataset as a unit of data publication, with which metadata and persistent identifiers can be associated. Importantly, due to their generic focus across domains, they frequently support only common metadata schemas (e.g., Dublin Core) but not domain-specific or user-defined schemas that many researchers desire. Many of these systems have evolved from digital content repositories and therefore do not support the deposit and storage of large datasets. For example, Dataverse limits dataset uploads to 2 GB, and figshare to 1 GB. By current standards, these limits are insufficient for even moderately sized scientific datasets. Moreover, all of these systems require that data be uploaded to the service. While a sensible requirement in many settings, for example when data is modest in size and the overarching goal is persistence, it is not necessarily appropriate when data is too large to move or when the primary motivation for data publication is sharing.

III. Data Publication

While the term *data publication* may be used to refer to a variety of different activities, certain characteristics typically differentiate a data publication system from other less formal data sharing approaches, such as placing data in a Dropbox folder, website, or other network-accessible storage. Specifically, a data publication system should allow for data to be identified, described, curated, immutable and/or verifiable, accessible, and preserved, as we now discuss.

Identified: Data is uniquely identified via a persistent identifier (PID) [25]. While data need not remain in the same location for all time, the associated identifier must be resolvable to the current location throughout the data's lifetime. Association of a persistent identifier enables proper citation and linkage between identifiers (e.g., papers and dataset) and improves discovery.

Described: Data is often locked within large files or collections of files. This makes data difficult to discover and hinders understanding of the conditions under which data was produced. Published data should be well described, perhaps not in its entirety, but to a level that facilitates discovery and some understanding of its contents many years in the future. Where possible, domain-specific ontologies and taxonomies should be used to describe contents in standardized ways.

Curated: Often, organizations hosting publications require assurances that data is complete, that it is well described, that licenses and copyrights have been signed, and that it meets other institutional policies before it is made available publicly under the auspices of that organization. Reasons for such assurances include improved data preservation, minimized liability, and improved data validity and re-usability. Thus, it is often necessary that data be curated, either manually or automatically, before it is published.

Immutable and/or verifiable: While it may be desirable to know that data, once published, will never change, immutability is challenging to provide. In some situations, it may suffice that users can determine whether the data they have accessed

has been modified from its published state, i.e., that the data is verifiable.

Accessible: Authorized users must be able to access data in straightforward ways, such as by Web UI and REST interfaces. Access methods should be open, documented, and standard such that a wide range of potential viewers may access the data without significant burden.

Preserved: Published data must be maintained, ensuring that over time contents are not lost and identifiers are not orphaned. Though the degree of preservation may differ among uses, it is generally expected that data be preserved even in the case of unexpected occurrences such as hardware failure or natural disasters.

Furthermore, data must be *discoverable* once published, so that others may find and use it. This requirement leads to another three characteristic features of data publication systems: that published data must be searchable, browsable, and retrievable, as we describe in the following.

Searchable: Users should be able to find data intuitively and quickly, with only minimal information known about the dataset. To achieve this, methods are required to quickly drill-down through large collections of data, to sort collections by arbitrary attributes, and to match both structured and unstructured queries.

Browsable: Data should be available via intuitive and standardized (Web or REST) interfaces, organized in standardized ways, and described using standardized metadata formats. These methods should enable users to browse many datasets quickly, inspect or explore dataset attributes, and explore connections between datasets. Thus, browsing aids discovery of known data and also facilitates serendipitous discovery.

Retrievable: Data and associated metadata should be available throughout the discovery process whether through the same on-line interface used for other discovery operations or by retrieving a copy of either data or metadata for off-line use.

IV. Globus Data Publication

Our data publication capabilities implement the five core activities shown in Fig. 1:

1) *Transfer*: Move or replicate data from its source to a suitable storage repository. In some cases, data may already be located in a suitable storage repository, in which case this step is not necessary.
2) *Describe*: Generate metadata, via elicitation from a human expert, automated extraction, and/or synthesis based on deeper analysis. This metadata may be loaded into one or more catalogs (Step 5), for purposes of discovery. A complete copy of the metadata should be collocated with the data.
3) *Curate*: Quality control and sign off, by the author, other specified users such as a collection manager or other administrator, and/or automated methods.
4) *Identify*: Assign a persistent identifier (PID) [25]. As with metadata, this PID may ultimately be loaded into one or more catalogs, for purposes of discovery.
5) *Register*: Load the PID and metadata into a catalog for subsequent discovery.

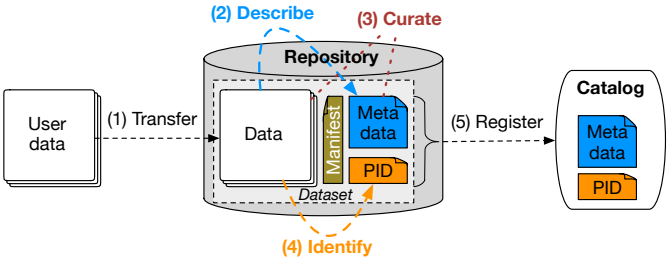

Fig. 1. The data publication process, showing the distinct transfer, metadata synthesis, curation, identification, and registration steps involved in publishing a dataset. A complete dataset comprises a set of files, associated metadata, a PID, and a manifest that allows easy verification of dataset contents.

Like other Globus capabilities, Globus data publication operates as cloud-hosted software-as-a-service (SaaS). That is, a single copy of the software is hosted for the entire user community, supporting multi-tenant access so that any authorized user can establish and manage their own data publication collection(s). A distinctive characteristic of our architecture is that the software used to manage the publication process is located on Amazon Web Services (AWS) cloud computers, but with data storage specified by the collection owner (e.g., on storage provided by the owner, by national facilities, or by cloud providers). To support a wide variety of use cases, data types, publication processes, data access policies, etc., we have designed the service to provide easy-to-use and self-service configuration. Fig. 2 shows the Globus data publication architecture.

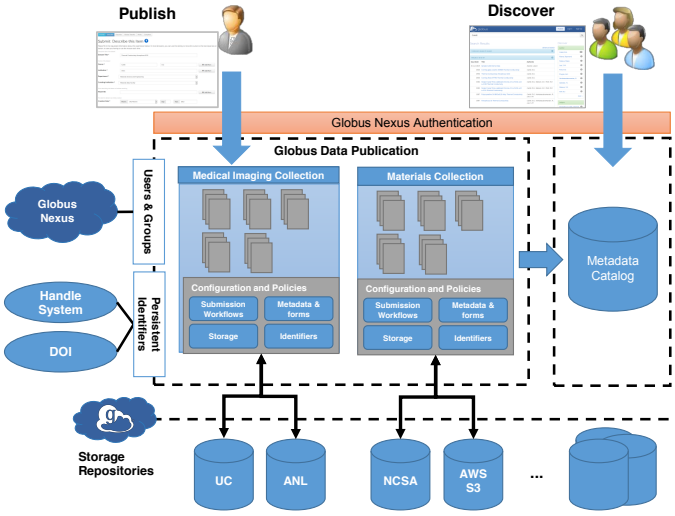

Fig. 2. Globus data publication architecture, showing (from top to bottom) the user interfaces for publication and discovery; a logical view of the system, with distinct collections and the metadata catalog; and the different repositories used to store data.

We base the Globus data publication model on a hierarchical structure composed of *communities*, optional *sub-communities*,

and member *collections*. A collection may map, for example, to a specific research project, to a department or group within an institution, or to an individual researcher. Within a collection, users publish *datasets* [26], each comprising:

1) One or more files and directories, located on accessible, but potentially distributed, storage repositories.
2) A set of metadata describing the nature, provenance, etc., of those files and directories; metadata is described as a set of text-based name-value pairs.
3) A unique PID, which can be resolved to locate the dataset even if the location of the dataset changes.

Datasets have an associated landing page—a web page hosted by Globus from which the dataset can be accessed—that is referenced by a URL relative to the Globus data publication service. Datasets are discoverable, based on metadata, via the data publication search interface. The PID and selected metadata may also be published in external catalogs, such as DataCite, so that users can discover and then resolve a PID to obtain a reference (URL) to the dataset's landing page. Dataset landing pages can also be indexed by web search engines such as Google, so that users can discover datasets by searching on public metadata, in the same way that they find web pages.

A. Publication service

We build our implementation on DSpace [27], [28], an institutional repository system designed for creating open digital repositories. DSpace provides an intuitive publication model, customizable publication workflows, granular access control, and easy-to-use and extensible web interfaces. While it has been widely adopted across research and library user communities, it is not designed to support data publication nor operate in a multi-tenant SaaS deployment.

DSpace supports flexible publication workflows, in which collection owners may prescribe different submission and curation steps required for publication. For example, owners may choose to require explicit curation for publications submitted to their collection. DSpace also offers customizable user interfaces to guide users through even complex publication workflows. While existing DSpace workflows, being designed to support publication of documents, are not perfectly suited for data publication, we found that only moderate modifications were required to apply them to data.

DSpace implements access control at all stages of the publication workflow as well as at each level of the publication model, including communities, collections, and datasets. This model allows for instance, different users to be given different roles in different contexts, such as administering one collection while being able to only view datasets in another. The extensible and granular nature of these permissions has allowed us to leverage and extend the access control model to manage access to remote data storage. In our model, permissions are applied to data and metadata separately, based on collection policies. Thus, for example, metadata can be made discoverable by all users while data (perhaps resident on a remote storage system) requires additional permissions.

DSpace is not without limitations for our purposes: it was not designed to handle large datasets, or data-oriented publications; it assumes that published files are collocated with the service; it relies on HTTP-based data submission and access; it requires static configurations for PID providers, submission workflows, metadata schema, and input forms; and it was designed for individual institution deployments rather than a scalable multi-tenant SaaS model. Thus, we made significant alterations to the DSpace user experience and architecture, as we describe below.

B. User interface

Globus data publication provides both general user interfaces and self-service administration interfaces. The latter allow authorized administrators to create communities and collections and to specify associated policies, with no direct operator involvement. For example, administrators can specify which users can perform actions on a collection, define submission and curation workflows, choose the storage repository to be used, and specify and configure PID providers.

The general user interfaces allow users to assemble and then submit datasets composed of files and metadata to collections, in a manner consistent with collection policies. They allow a user to perform any role assigned to them, such as data submission, curation, or discovery. Globus data publication renders context-dependent interfaces based on the context in which a user is acting. Thus, for example, metadata input forms are customized to match the forms defined by the collection, curation task lists are presented based on group membership, and search results are filtered to show only data that the user is authorized to view.

Building upon DSpace's user interface we have created a data-oriented interface for all user interactions. To address data publication needs we have extended and altered significant sections of the interface, including orienting around a data publication dashboard. The publication dashboard provides access to all activities performed by a user. We have significantly altered the administration interfaces, allowing a custom self-service model. Fig. 3 shows the publication dashboard and submission workflow.

C. Authentication and identity and group management

The ability to specify and enforce community- and collection-specific policies is an essential element of the architecture. Policies may control, for example, who can submit to a particular collection, who can access published data (metadata or files and directories), and who can perform workflow tasks such the creation of new collections.

To this end, our architecture leverages Globus identity, profile, and group management capabilities [29] for user authentication and to support the definition of rich data publication policies using flexible group-based authorization. Globus identity management allows users to create a unique Globus identity that supports single sign-on across services. Thus, users can authenticate when publishing data with the same identity that they use to transfer data or manage groups. As

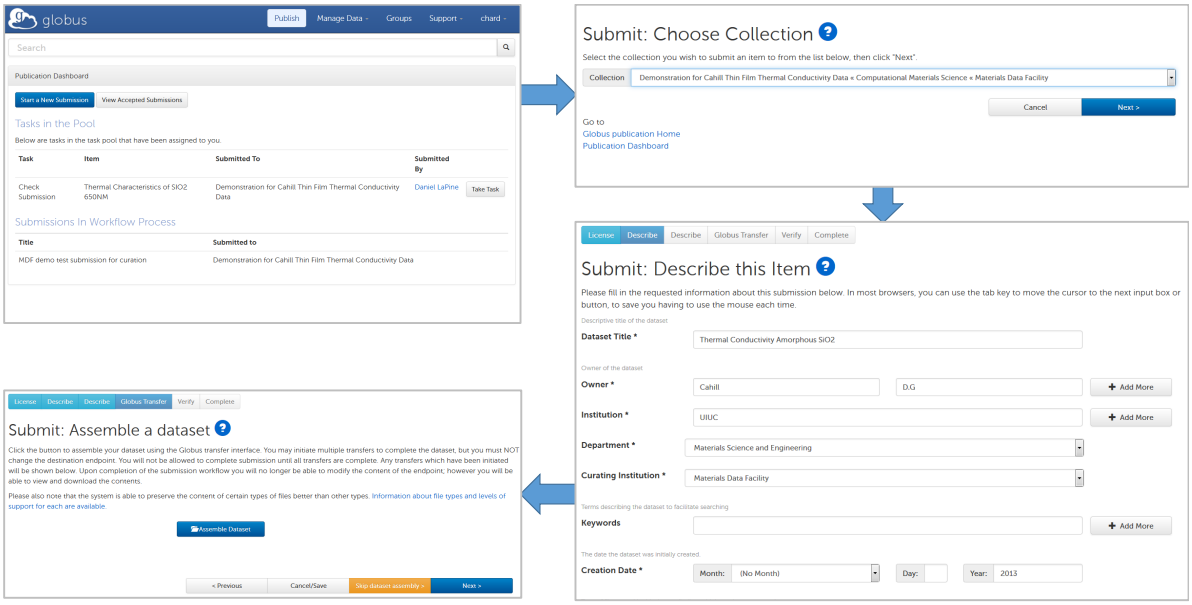

Fig. 3. An example submission workflow, showing (clockwise from upper left) the Publication Dashboard; selecting a collection; entering relevant metadata; and assembling a dataset.

researchers often have multiple identities (i.e., user accounts in different contexts), Globus allows users to link external identities (e.g., campus identities via InCommon [30], national cyberinfrastructure identities such as XSEDE [31], commercial accounts like Google) to their Globus identity. They can then authenticate with any linked identity.

We use Globus user-managed groups as a basis for specifying roles, policies, and workflows (e.g., invitation, membership approval). Users can leverage Globus group invitation workflows to create groups, invite users to a group via their Globus identity or email, and make changes to groups that are instantly reflected and enforced by the data publication system, for example to allow or restrict access to a dataset.

We have integrated Globus identity and group management capabilities with DSpace via Globus REST APIs and web-based workflows. We have added support for OAuth2 authentication workflows in DSpace allowing users to authenticate with DSpace using Globus identities or any of the supported linked identities. We have removed the standard DSpace group model and replaced it with more flexible Globus groups. Thus, all identity and group management is performed through existing Globus interfaces.

D. Data storage, access, and transfer

Researchers need to publish data of varying sizes and collection owners and administrators often desire that published data is stored on their own resources in order to maintain control for privacy and disaster recovery purposes. Thus, we implement a "bring your own storage" model that allows collection owners to define where datasets published in their collections are to be stored. With this model, data need not be copied to the data publication service in order to be published: collection owners can specify use of external storage or even cloud resources such as Amazon S3. As a result, our model assumes that storage providers are responsible for long-term data preservation.

To enable reliable high-speed transfer of large datasets when depositing and downloading data, as well as to support flexible data access, we use Globus transfer capabilities [32]. Globus provides high performance, secure, third party data movement and synchronization between "endpoints"—storage resources that implement Globus data access APIs. Globus handles the difficult aspects of data transfer; a user merely requests data transfers and Globus automatically tunes parameters to maximize bandwidth usage, manages security configurations, recovers from faults, and notifies users of completion and other events. We leverage these capabilities to allow users to assemble datasets for publication and also to access published datasets and transfer them to a desired location.

Globus supports in-place sharing of data directly from existing storage repositories. Sharing is accomplished by creating a virtual "shared endpoint" rooted at a specific path under a Globus endpoint. Users may add access permissions for users or groups to this shared endpoint. We build upon Globus sharing in the data publication service, so that each collection has an associated shared endpoint, hosted on selected storage resources, and used to store datasets. Each dataset is housed in a unique directory created within the shared endpoint. The data publication authorization framework can then manage access to the shared endpoint so that, for example, after submission, data are shared with (readable by) any defined curators, and at the conclusion of the curation process are readable by groups that have permission to view data in that collection.

We use Globus APIs for creating shared endpoints and managing access permissions to data. When configuring a

collection we require that users create a Globus endpoint on their storage repository and create a local user account from which Globus data publication can manage access to the storage repository. When creating a collection, we create an associated shared endpoint on a selected file path on which all datasets will be stored. We rely upon Globus web interfaces to allow users to assemble datasets. This approach has the advantage that assembled datasets may be composed of data from many different locations and can be structured (in terms of directories and naming) as the user wishes. Fig. 4 illustrates the interface for assembling a dataset.

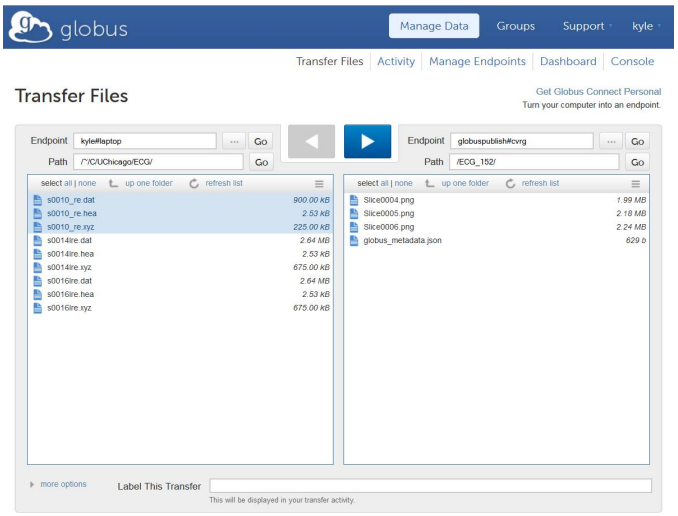

Fig. 4. Assembling a dataset by transferring data from several sources. The endpoint `globuspublish#cvrg`, on the right, is where files to be published are collected; the screenshot captures a moment when files are being added from a source endpoint `kyle#laptop`.

E. Persistent identifiers

Persistent identifiers (PIDs) are an important component of any publication model as they allow published objects to be uniquely identified for all time. This permits proper citation with confidence that the identifier will ultimately resolve to the current location of the data even if it should be moved over time. There are several types of identifiers commonly used, such as DOIs and Handles, and each has its own semantics governing use and infrastructure for creation. For instance, DOIs provide guarantees of persistence, consistency, and semantic interoperability. Identifier providers are the systems tasked with creating identifiers of a particular type. Providers typically require that clients are registered in advance before being able to "mint" new identifiers. Client credentials are used to authenticate client interactions with the provider. Often, providers support namespaces (e.g., "shoulders") in order to associate each identifier with its creator.

Our architecture is designed to support flexible self-service configuration, allowing administrators to configure what type of identifiers should be used for a given collection, what identifier provider should be used, and what namespace and credentials are used to create identifiers. Our architecture relies on published APIs for the various providers. We currently support the Handle System [33], DOIs [34], EZID [35] and Bitly [36]. DSpace is designed to support only a single identifier provider for all collections. Thus, we have extended this model to enable many such providers to be available and for individual collection owners to select and configure their chosen identifier provider via a web interface.

F. Metadata form registry

To ease the process of working with large amounts of metadata from different domains, we expose a model by which users can establish metadata definitions and create and manage "metadata input forms"—collections of metadata attributes that are associated with one another—that are rendered as part of the submission workflow. When configuring a new collection, users may create an XML schema file that describes unique metadata fields that can be associated with a dataset. They can also create an input form XML file that defines how individual metadata input fields are arranged. This file maps input fields displayed on individual submission pages to metadata fields defined in the schema. It also specifies field restrictions (e.g., multi valued, required, etc.) and enumerations of potentially valid values. To ease the process for users who do not wish to define custom metadata forms, we provide input forms for a set of commonly used metadata formats, such as Dublin Core [37] and the DataCite [38] mandatory, recommended, and optional schema.

G. Submission and curation workflows

Like metadata forms, users often have different requirements of the submission and curation workflows used during publication. To meet these needs we have extended DSpace's workflow model to enable self-service workflow selection. DSpace currently supports only a single workflow that must be predefined and is used across all collections. Using our model, users are able configure a collection with a customized submission workflow that may apply different steps in different orders. The steps supported are: publication lookup, license approval, metadata description, dataset assembly, and verification. We leverage the standard DSpace curation model through which collections may optionally enable data curation, metadata curation, and explicit approval.

H. Discovery

Efficient storage and search across extensible and schemaless semi-structured metadata is a complex research problem. Scientific metadata in particular pose difficult challenges, such as requiring typing and association between related metadata fields. As we must support arbitrary metadata schemas to meet the needs of a diverse user base, we designed our model to support new metadata fields dynamically. Thus, we must support schema mutability at runtime so that new attributes can be added and schema can evolve over time. Our implementation leverages a modified DSpace search and discovery model that enables intuitive browsing across communities,

collections, and datasets. This model is based upon Apache Solr [39] and supports rich, free-text search. Importantly, it also enables secure contextualized search results that ensure that access control on data publications is enforced.

We have customized the default facets shown to users and modified the results displayed to be more easily navigated. We have also modified the entire site's user experience to be more search oriented, incorporating search capabilities on every page and providing context-specific search results based on the page from which the search was conducted.

While these capabilities enable effective search over published data, we are working to improve them further to meet the unique requirements of large research metadata. In particular, we aim to support typed metadata, metadata objects, and arbitrary schemas, and to enable self-service, per-collection configuration of search interfaces. We also view it as critical to support sparse data models, as it is common to have both many attributes associated with particular entity types and to have many entity types. We have previously studied the impact of data model choices on query performance in different metadata storage environments [40]. In that investigation, we identified relational models that can be used to efficiently store and query such metadata. We plan to apply these approaches to create a highly scalable metadata catalog model in which sparse, schema-less data can be stored and accessed efficiently. Our goal is a strong entity model in which collections of attributes (called entities) are defined based on the metadata forms applied to a given collection. We will then use a decomposed storage model to associate custom or enhanced metadata attributes with entities when required.

I. Deployment

We deploy our data publication capabilities as part of the Globus service, which applies lessons learned from commercial cloud services to meet high availability targets. (Globus achieved 99.96% availability in 2013 [29].) Applying the same SaaS model, we operate a single instance of the data publication software for all users, and use a sophisticated multi-tenant model that supports scaling to many concurrent users. Our cloud-based, managed deployment model on AWS Elastic Compute Cloud (EC2) allows new instances to be provisioned quickly in the case of failure without data loss.

The DSpace implementation comprises a number of Java Servlets and an extensible web-based JSP interface. We host these components in an Apache Tomcat container on provisioned EC2 instances. To ensure a stateless deployment, we host community, collection, groups, and dataset metadata on AWS Relational Database Service (RDS)—a reliable and elastic cloud-based database. We co-locate the Solr indexes used for search and discovery with the publication service as data can be re-indexed if the instance is lost. Finally, we apply Globus monitoring practices to ensure that our operations team is notified of failures and other issues affecting the service.

V. Pilot Deployments

Over a three month period we have engaged in pilot deployments covering fields such as materials, climate change, and biomedical research. These pilots have helped to refine our data publication capabilities before making it generally available. Here, we describe nine such pilots. Some are anonymous, as not all wish their names to be shared at this time.

A. Materials Data Facility

The Materials Data Facility (MDF) is a collaborative project of the University of Chicago, the National Center for Supercomputing Applications (NCSA), the Chicago Center for Hierarchical Materials Design (CHiMaD), and other partners. It aims to accelerate the process of creating new materials by facilitating the publication, preservation, and sharing of experimental and simulation materials datasets.

We recently created, in collaboration with the Cahill lab at the University of Illinois at Urbana Champaign (UIUC), a first MDF collection. The datasets published into this collection are comprised of thin film thermal conductivity data (collected by the 3ω technique) for materials ranging from single crystal yttria-stabilized zirconia to polymeric polymethylmethacrylate. To meet description requirements, we developed new metadata schemas and input forms based on discussions with lab members, including Dublin core metadata [37] as well as collection-specific metadata (e.g., department and curating institution), domain-specific metadata (e.g., film classification, composition, film thickness, technique), and associated journal publication information (paper title, authors, DOI, etc.). Storage for the collection is operated by NCSA and Argonne. DOIs are allocated to each dataset from the UIUC DOI shoulder.

B. CardioVascular Research Grid

The CardioVascular Research Grid (CVRG) [41] is a national resource for researchers studying cardiac diseases. It supports the storage and analysis of complex datasets composed of time-series, imaging, and genomic data. CVRG hosts a broad range of tools and services for storing and managing cardiac data, querying combinations of these data, applying statistical learning methods for biomarker discovery, analyzing cardiac imaging, and analyzing and annotating ECG data. CVRG users also have requirements for publishing both raw and analyzed data for dissemination to collaborators and more widely to the cardiovascular research community.

To explore the data publication needs of CVRG users we have created two pilot collections to store text-based and ECG-based publications. CVRG researchers have developed a specialized ECG publication metadata schema that is used to describe ECG publications and includes additional information about investigators and grants. The two collections both utilize storage repositories hosted on CVRG resources. Both collections also use a modified submission workflow that first enables PubMed search to discover associated publications that are then used to populate dataset metadata.

C. Accelerated Climate Modeling for Energy

The Accelerated Climate Modeling for Energy (ACME) project involves eight national laboratories, the National Center for Atmospheric Research, four academic institutions, and one commercial company. It aims to develop and apply climate and earth system models to climate change research. ACME uses an existing data cataloging system called THREDDS (Thematic Realtime Environmental Data Distributed Services) [42] to publish a variety of earth systems data. Each participating location hosts a THREDDS node that can be manually populated with published data. This process typically requires that an administrator receive data from submitters, copy it to a shared location, execute an extraction script, and then manually register the dataset and metadata in THREDDS.

To improve and automate the ACME publication process, we have piloted the use of Globus data publication to provide standard submission and curation workflows for creating ACME data publications and a federated index across distributed THREDDS catalogs. We have also used this pilot to explore and prototype a model for automated extraction of metadata during the submission process. As future work we plan to provide this capability to all Globus data publication users. ACME data is typically in NetCDF format [43], with much useful metadata embedded in NetCDF headers [44]. Thus, we developed a modified submission workflow that extracts embedded metadata by invoking an ACME-supplied extraction tool at the data location. This approach relies on remotely executing an extraction tool via a GridFTP server [45] at the dataset location. As part of the submission, researchers transfer data to the extraction server after which the automated extraction process is invoked. As this process may take some time, the submission workflow is blocked until completion. After metadata is extracted, the dataset and its metadata are registered in both Globus and THREDDS catalogs.

D. Robust Decision Making on Climate and Energy Policy

The Center for Robust Decision Making on Climate and Energy Policy (RDCEP) aims to improve computational models for evaluating climate and energy policies. RDCEP researchers often wish to publish large datasets of a wide variety types, and while they have access to significant storage capacity on institutional resources they are not equipped to provide persistent identifiers, public data access, or descriptive metadata.

In collaboration with RDCEP researchers we created a collection for storing Global Gridded Crop Model Intercomparison [46] datasets. RDCEP researchers developed a customized metadata schema that can represent, among other things, climate variables (e.g., relative humidity, precipitation), spatial resolution, temporal resolution, and various masks. Data is stored on the University of Chicago Research Computing Center and uses the Globus-operated Handle server to associate persistent identifiers with published data.

E. Genetic Research

In collaboration with a national genetic research lab we have created a data publication collection for publishing large genomic datasets. This research group investigates the affects of different environments on living tissue and has amassed large collections of genetic data.

The collection that we have created for this group uses storage provided by the University of Chicago, Handles minted by a Globus-operated Handle server, and the MINiML (MIAME Notation in Markup Language) metadata schema [47], developed by the NCBI for describing microarray gene expression data and other high-throughput molecular abundance data. MINiML includes attributes such as organism, manufacturer, catalog number, and technology (e.g., spotted DNA/cDNA).

F. Compute Canada

Compute Canada is a federated national cyberinfrastructure provider for the Canadian research community. Composed of four partner providers, Compute Canada integrates compute, data storage, and research facilities across the country. With a mandate to support computational Canadian research, and already a storage provider for large research data, Compute Canada is exploring methods to provide scalable data publication capabilities to their partners.

Compute Canada have developed pilot data collections related to high energy physics (HEP) simulations and the marine biology of Pacific salmon. In each case, researchers have developed custom metadata schema and forms that describe domain-specific attributes such as (in the HEP case) ECM energy and geometry version. Both collections are configured to use storage provided by Compute Canada. These collections are differentiated by their data sizes, with one HEP simulation dataset being $\sim$100GB while Pacific herring datasets are $\sim$100MB. This pilot is also exploring linkages between the readily accessible data described in Globus and long-term archival systems which often require that data be transformed into new formats and moved to appropriate storage systems.

G. Research Institutions

We are working with three US research institutions to investigate the potential of outsourcing campus data publication needs. Each pilot focuses on a different scientific domain and uses different storage and persistent identifier mechanisms.

The first institution used the system to publish real-time cardiovascular MRI acquisition and rapid image reconstruction data. It used storage resources provided by the institution's research computing center and relies on the Globus-operated Handle server to create persistent identifiers. We have developed a collection for this group that contains $\sim$5GB cardiac MRI k-space datasets. One of these datasets has since been referenced from a PLOS ONE publication.

In the second case, we worked with the institution's Library to deploy a geospatial data publication collection. The required metadata blends values from the Dublin Core and extended Dublin Core Terms schema with emerging public schemas related to the domain. We adapted the domain-specific schema for use in Globus data publication and provide identifiers using the Globus-operated Handle server, with the goal of transitioning to the institution's existing Handle server as

the pilot progresses. Storage is provided by the institution's information technology organization.

In the third case, we created an organic chemistry collection that includes NMR and crystallographic information. Many of these datasets encompass a large number of relatively small files, a configuration that creates somewhat different challenges than the fewer, larger files typical of other pilots. However, the use of Globus transfer to assemble datasets avoids most difficulties that would otherwise be inherent in dealing with many files. As in other pilots, the metadata schema builds upon the Dublin core schema for describing related publications and extends it with domain-specific information provided by the domain experts and includes linkage to researchers via their ORCID. For use in this pilot, the institution has issued new credentials with EZID which are used directly by Globus to create DOIs (under the institution's shoulder) when datasets are published in the collection.

H. Discussion

These pilots cover a wide range of use cases including national cyberinfrastructure, large scale consortiums, institution research computing and library services, and small to medium research projects.They exercise a range of capabilities provided by the Globus publication service. For example, they encompass both small and large data; varying numbers of files and directories; different metadata requirements, from standard publication and domain-specific schema through to custom, collection-specific schema; a range of storage systems, distributed across North America; different persistent identifiers, including DOIs and Handles from both Globus and institutional managed accounts; and different input and curation workflows.

While these pilots have proved successful, they have already prompted modifications to our service and will also influence future development. For example, we quickly learned that different research groups have significantly different requirements with respect to metadata, workflows, and persistent identifiers. While we had already incorporated some support for such differences, we determined that additional support for customization was needed in some areas. For example, pilot users required different persistent identifiers, some required DOIs issued from DataCite registers while others required light-weight identifiers such as Bitlys. In addition, in many cases groups expressed needs to use existing institutional identifier accounts. Another example of such flexibility related to the need to modify submission workflows, where pilots required different input pages presented in different orders.

The pilot also identified the need for more flexible metadata support. While manual, form-based metadata entry is a useful start, general use will require the ability to mix automatically derived and human-supplied metadata. We will build upon the ACME prototype to make such automated extraction capabilities broadly available.

Similarly, while our attribute-value metadata model has the advantage of simplicity, we find that it is often important to be able to associate a type and other attributes, such as units, with metadata, for effective search and discovery. We also find that multiple metadata elements must often be bound together for appropriate context and effective search, so that for example an author is associated with their institution and a value with its unit. Thus, we require support for defining, using and searching metadata objects.

Finally, we found that while our current metadata model is oriented around datasets, users in domains such as climate and earth science also need to be able to associate metadata with dataset members (i.e., individual files and directories). Often, each file in a dataset cover a unique aspect of the investigation (e.g., a particular data type such as precipitation, a particular experimental condition such as energy density, or a geographical area). Thus, these individual files (a subset of the dataset rather than the entire dataset) are often of interest to researchers and therefore we require methods to be able to describe and find individual files within a dataset.

VI. Summary

Data publication is important for both everyday scientific activities and the preservation of research outputs. To meet these needs we have developed unique SaaS data publication capabilities that enable small and large datasets to be identified, described, curated, accessed in a controlled manner, and preserved. These capabilities, incorporated and delivered within the Globus research data management service, provide a powerful, self-service model for creating collections, describing datasets, and defining policies, such as access control. Users can submit data for publication into any collection to which they have been granted access and can use a powerful search interface to discover published data.

Key features of Globus data publication include: support for large datasets, with appropriate policies for all types of institutions and researchers; the ability to publish data directly from locally owned storage or from cloud storage without needing to upload it to cloud storage; extensible metadata that can describe the specific attributes of different scientific domains; flexible publication and curation workflows that can be easily tailored to meet user requirements; public and restricted collections that give complete control over who may access published data; support for different persistent identifier providers (e.g., DOI, Handle, Bitly) using public and institutional credentials; and a rich discovery model that allows others to search, access, and use published data.

A set of nine pilot projects have established the value of these capabilities and provided valuable feedback. We are now focused on full deployment and operations, with the goal of general availability in the near future. Future work will focus on enhancing metadata capabilities; enabling intuitive and self-service definition of input forms based on complex scientific metadata; developing a scalable, secure and configurable multi-collection search mechanism; developing a machine-accessible REST API; and improving multi-tenant self-service capabilities. We will develop a form registry to enable the creation and sharing of metadata schema and forms, and import from existing standards. In so doing we will

expand upon our current search capabilities, moving towards an optimized sparse-data search model that supports strongly typed attributes and metadata objects.

ACKNOWLEDGMENTS

We thank the Globus team for implementing and operating Globus services, and the participants in our publication pilots for their invaluable contributions. This research was supported in part by DOE contract DE-AC02-06CH11357; NIH contract 1U54EB020406-01, Big Data for Discovery Science Center; and NIST contract 60NANB15D077.

REFERENCES

[1] T. H. Vines, A. Albert, R. Andrew, F. Debarre, D. Bock, M. Franklin, K. Gilbert, J.-S. Moore, S. Renaut, and D. Rennison, "The availability of research data declines rapidly with article age," *Current Biology*, vol. 24, no. 1, pp. 94–97, 2014.

[2] A. A. Alsheikh-Ali, W. Qureshi, M. H. Al-Mallah, and J. P. Ioannidis, "Public availability of published research data in high-impact journals," *PLoS ONE*, vol. 6, no. 9, p. e24357, 2011.

[3] B. Waters, "Software as a service: A look at the customer benefits," *J Digit Asset Manag*, vol. 1, no. 1, pp. 32–39, 2005.

[4] I. Foster, "Globus Online: Accelerating and democratizing science through cloud-based services," *Internet Computing, IEEE*, vol. 15, no. 3, pp. 70–73, May 2011.

[5] K. Chard, S. Tuecke, and I. Foster, "Efficient and secure transfer, synchronization, and sharing of big data," *Cloud Computing, IEEE*, vol. 1, no. 3, pp. 46–55, 2014.

[6] M. J. Costello, "Motivating online publication of data," *BioScience*, vol. 59, no. 5, pp. 418–427, 2009.

[7] J. Kratz and C. Strasser, "Data publication consensus and controversies," *F1000Research*, vol. 3, no. 94, 2014.

[8] L. Candela, D. Castelli, P. Manghi, and A. Tani, "Data journals: A survey," *Journal of the Association for Information Science and Technology*, 2015.

[9] J. Nardone, "Computerized Material Property Data Information System," Plastics Technical Evaluation Center (PLASTEC), Tech. Rep. N31, 1976, http://1.usa.gov/1yqtBDg.

[10] M. Mailman, M. Feolo, Y. Jin, M. Kimura, K. Tryka, R. Bagoutdinov, L. Hao, A. Kiang, J. Paschall, L. Phan, N. Popova, S. Pretel, L. Ziyabari, M. Lee, Y. Shao, Z. Wang, K. Sirotkin, M. Ward, M. Kholodov, K. Zbicz, J. Beck, M. Kimelman, S. Shevelev, D. Preuss, E. Yaschenko, A. Graeff, J. Ostell, and S. Sherry, "The NCBI dbGaP database of genotypes and phenotypes." *Nature Genetics*, vol. 39, no. 10, pp. 1181–1186, 2007.

[11] UniProt Consortium, "Uniprot: a hub for protein information," *Nucleic Acids Research*, vol. 43, no. D1, pp. D204–D212, 2015.

[12] "SIMBAD Astronomical Database," http://simbad.u-strasbg.fr/simbad/, web site. Accessed: June 8, 2015.

[13] "National Database for Autism Research (NDAR)," https://ndar.nih.gov/, web site. Accessed: June 8, 2015.

[14] "Federal Interagency Traumatic Brain Injury Research (FITBIR)," https://fitbir.nih.gov/, web site. Accessed: June 8, 2015.

[15] "National Climatic Data Center (NCDC)," http://www.ncdc.noaa.gov/, web site. Accessed: June 8, 2015.

[16] M. Chen, A. C. Stott, S. Li, and D. A. Dixon, "Construction of a robust, large-scale, collaborative database for raw data in computational chemistry: The Collaborative Chemistry Database Tool (CCDBT)," *Journal of Molecular Graphics and Modelling*, vol. 34, pp. 67–75, 2012.

[17] "Materials Atlas," http://bit.ly/1FmMjQk, web site. Accessed: June 8, 2015.

[18] "PURR: Purdue University Research Repository," http://purr.purdue.edu, web site. Accessed: June 8, 2015.

[19] "Data Repository for the University of Minnesota (DRUM)," https://www.lib.umn.edu/datamanagement/drum, web site. Accessed: June 8, 2015.

[20] "ScholarSphere," https://scholarsphere.psu.edu/, web site. Accessed: June 8, 2015.

[21] "Dataverse," http://thedata.org, web site. Accessed: June 8, 2015.

[22] "figshare," http://figshare.com, web site. Accessed: June 8, 2015.

[23] "Zenodo," https://www.zenodo.org, web site. Accessed: June 8, 2015.

[24] "Dryad," https://www.datadryad.org, web site. Accessed: June 8, 2015.

[25] International Council for Science: Committee on Data for Science and Technology, "Data citation standards and practices," http://www.codata.org/task-groups/data-citation-standards-and-practices, 2010.

[26] A. H. Renear, S. Sacchi, and K. M. Wickett, "Definitions of dataset in the scientific and technical literature," in *Proceedings of the 73rd ASIS&T Annual Meeting on Navigating Streams in an Information Ecosystem - Volume 47*, ser. ASIS&T '10. Silver Springs, MD, USA: American Society for Information Science, 2010, pp. 81:1–81:4.

[27] "DSpace," http://dspace.org, web site. Accessed: June 8, 2015.

[28] M. Smith, M. Barton, M. Bass, M. Branschofsky, G. McClellan, D. Stuve, R. Tansley, and J. H. Walker, "DSpace: An open source dynamic digital repository," *D-Lib Magazine*, vol. 9, no. 1, 2003.

[29] K. Chard, M. Lidman, J. Bryan, T. Howe, B. McCollam, R. Ananthakrishnan, S. Tuecke, and I. Foster, "Globus Nexus: Research identity, profile, and group management as a service," in *IEEE 10th International Conference on e-Science*, vol. 1, Oct 2014, pp. 31–38.

[30] W. Barnett, V. Welch, A. Walsh, and C. A. Stewart, "A roadmap for using NSF cyberinfrastructure with InCommon," 2011.

[31] J. Towns, T. Cockerill, M. Dahan, I. Foster, K. Gaither, A. Grimshaw, V. Hazlewood, S. Lathrop, D. Lifka, G. D. Peterson, R. Roskies, J. R. Scott, and N. Wilkins-Diehr, "XSEDE: Accelerating scientific discovery," *Computing in Science and Engineering*, vol. 16, no. 5, pp. 62–74, 2014.

[32] B. Allen, J. Bresnahan, L. Childers, I. Foster, G. Kandaswamy, R. Kettimuthu, J. Kordas, M. Link, S. Martin, K. Pickett, and S. Tuecke, "Software as a service for data scientists," *Communications of the ACM*, vol. 55, no. 2, pp. 81–88, 2012.

[33] S. Sun, L. Lannom, and B. Boesch, "Handle system overview," Internet Engineering Task Force, Network Working Group, RFC 3650, November 2003. [Online]. Available: https://www.ietf.org/rfc/rfc3650.txt

[34] "Digital Object Identifier System," http://www.doi.org, web site. Accessed: June 8, 2015.

[35] "EZID," http://ezid.cdlib.org/, web site. Accessed: June 8, 2015.

[36] "Bitly," https://www.bitly.com, web site. Accessed: June 8, 2015.

[37] S. Weibel, J. Kunze, C. Lagoze, and M. Wolf, "Dublin core metadata for resource discovery," *Internet Engineering Task Force RFC*, vol. 2413, no. 222, p. 132, 1998.

[38] "DataCite," https://www.datacite.org, web site. Accessed: June 8, 2015.

[39] "Apache Solr," http://lucene.apache.org/solr/, web site. Accessed: June 8, 2015.

[40] T. Malik, K. Chard, and I. Foster, "Benchmarking cloud-based tagging services," in *IEEE 30th International Conference on Data Engineering Workshops (ICDEW)*, March 2014, pp. 231–238.

[41] R. L. Winslow, J. Saltz, I. Foster, J. J. Carr, Y. Ge, M. I. Miller, L. Younes, D. Geman, S. Graniote, T. Kurc, R. Madduri, T. Ratnanather, J. Larkin, S. Ardekani, T. Brown, A. Kolasny, K. Reynolds, M. Shipway, and M. Toerper, "The CardioVascular Research Grid project," in *Proceedings of the AMIA Summit on Translational Bioinformatics*, 2011, pp. 77–81.

[42] B. Domenico, J. Caron, E. Davis, R. Kambic, and S. Nativi, "Thematic Real-time Environmental Distributed Data Services (THREDDS): Incorporating interactive analysis tools into NSDL," *Journal of Digital Information*, vol. 2, no. 4, p. 114, 2002.

[43] Unidata, "netCDF: An access interface for self-describing portable data," 2002, http://www.unidata.ucar.edu/packages/netcdf/.

[44] B. Eaton, J. Gregory, B. Drach, K. Taylor, S. Hankin, J. Caron, R. Signell, P. Bentley, G. Rappa, H. Hck, A. Pamment, and M. Juckes, "NetCDF Climate and Forecast (CF) metadata conventions, version 1.5," 2010.

[45] W. Allcock, J. Bresnahan, R. Kettimuthu, M. Link, C. Dumitrescu, I. Raicu, and I. Foster, "The Globus striped GridFTP framework and server," in *SC'2005*, 2005.

[46] J. Elliott, C. Müller, D. Deryng, J. Chryssanthacopoulos, K. Boote, M. Büchner, I. Foster, M. Glotter, J. Heinke, T. Iizumi *et al.*, "The global gridded crop model intercomparison: data and modeling protocols for phase 1 (v1. 0)," *Geoscientific Model Development*, vol. 8, no. 2, pp. 261–277, 2015.

[47] "MINiML (MIAME Notation in Markup Language)," http://www.ncbi.nlm.nih.gov/geo/info/MINiML.html, web site. Accessed: June 8, 2015.