A Cyberplatform for Sharing Scientific Research Data at DataCenterHub

Ann Christine Catlin
Purdue University

Chandima HewaNadungodage
Purdue University

Santiago Pujol
Purdue University

Lucas Laughery
Purdue University

Chungwook Sim
University of Nebraska-Lincoln

Aishwarya Puranam
Purdue University

Andres Bejarano
Purdue University

DataCenterHub is a new solution for preserving, sharing, and discovering data produced by scientific research. Datasets are organized by experiments, with a simple common structure for metadata, file collections, and key parameters. Researchers associate annotations, reports, media, and measurements to each experiment, and interactive viewers interpret data by type and use so that they can be investigated before downloading. Parameters are extracted for discovery of key data otherwise hidden in files. DataCenterHub provides an alternative discipline-neutral solution, with the goal of helping researchers classify and share data for easy discovery and exploration.

The preservation of scientific research data has long been a key need of the research community.[1] Whether generated by large multi-institutional groups collaborating on funded projects or by a single graduate student writing a thesis, any dataset has the potential to shape and direct future research. Preservation is only one element of the data solution, however. Data that are archived but not shared lose all value for the broader research community, and shared data that cannot be discovered and explored easily will not have the benefit or impact that is the essential aim of data sharing. Data upload that is user-friendly, flexible, and comprehensive is a critical requirement for data sharing; but ease of discovery and exploration is equally important. Data that are hidden inside zip files, repositories that must be downloaded to be investigated, and data

collections that are not organized to clearly represent research activities do not offer the kind of access and usability that communities need to realize the true benefits of shared research data.

Several areas of research have invested heavily in customized environments for data archiving and viewing. For example, sites like the Encode Genome Browser[2] and the National Center for Biotechnology Information (www.ncbi.nlm.nih.gov)[3] are customized for genomic research and are very useful for that community. But most disciplines do not have full-service, customized solutions for sharing research data. A discipline-neutral platform is needed to serve these communities. If the platform also provides powerful methods for data discovery, exploration, and comparison, it offers some advantages over customized solutions because it facilitates viewing and study across disciplines.

A broad body of research addresses both customized and discipline-neutral data sharing repositories. The Registry of Research Repositories (www.re3data.org) has cataloged more than 1,500 repositories. Our contribution to this field is a unique, comprehensive discipline-neutral representation for research data that encompasses both structured data and repository files. DataCenterHub provides a simple structure and feature-rich web platform for dataset upload, sharing, and discovery that addresses the following needs:

- Self-serve platform where data from any scientific discipline can be uploaded, shared, and explored.
- Standards for organizing research data based on experiments, which are collections of structured and unstructured data, repositories of files, metadata, and user-defined parameter sets. Experiments are assigned bibliographic metadata, keywords, and spatial and temporal information.
- Simple process for uploading research datasets, with features for easy assignment of metadata, bulk upload of data and files, annotation, data classification, and customizable parameter sets.
- Repository files connected to experiments so that each file inherits all experiment metadata. Uploaded files are automatically classified by type (media, reports, drawings, data), and keyword extraction based on file content can be enabled.
- Interactive web interfaces to view, navigate, and search published experiments through unique "dataviews," with support for comparing experiments both within and across datasets.
- Specialized viewers that are launched based on data type, such as media galleries, viewers to explore user-defined parameter sets, and maps for spatial data with markers that display metadata at that location.

Datasets are defined as annotated experiments, with direct linkage of structured data and file collections to the experiments from which they were generated or assembled. For structured data, we present an interactive tabular view of the data that supports numeric and text filtering, search, and navigation—with drill-down views that display additional dimensions of the data in more detail. File collections have data types that are interpreted by the platform to launch type-specific exploration tools, for example:

- File collections of type "media" (photos, drawings, videos, images, and so on) have interactive viewers which display, play, and download files, with thumbnails for navigation and annotations for investigation.
- File collections of type "data" (CSV files, text files, and so on) have interactive explorers to search, view, and download files. The data explorer can also launch computational tools that operate on data in the file and present results or graphs.

The goal of our data platform is to provide a simple, common structure for scientific data and metadata. We offer a self-serve, discipline-neutral platform available to all researchers for preserving, sharing, and exploring scientific data in a way that more accurately represents research activity organization.

RELATED WORK

There is a growing interest in data sharing platforms that support preservation and exploration of research data. A large number of resources and publicly available systems exist, with diverse focus, goals, and capabilities. We briefly discuss several widely used systems and compare their coverage of research data types to that offered by DataCenterHub.

Figshare[4] is a popular platform that provides excellent features for file upload and keyword searching, and it offers citable sources for its collection of research outputs. However, users cannot archive and associate searchable structured data to their figshare research outputs, and nearly half of uploaded outputs consist of a single file, most often a spreadsheet.

Dryad[5] is a platform where users can make research data publicly available and link those data to published literature. Data packages are assigned keywords, abstracts, and spatial and temporal information. However, data can only be downloaded, and online exploration is not supported. SQLShare[6] provides its users with interfaces to upload, share, and manipulate research data. Each dataset is essentially a spreadsheet; once uploaded, users can write and run SQL queries on these spreadsheets and share the results. DataHub[7] from MIT is an installable framework for research data sharing. Users can install, manage, and run their own data sharing platform using this framework. Users view data in tabular formats and can write, save, run, and share SQL queries. It also provides analytical tools and charts for data visualization. These three systems support data sharing for CSV, text, or JSON files only, whereas DataCenterHub places no restrictions on the file types that can be uploaded, shared, and explored.

CKAN[8] and DKAN[9] are two popular open source data management platforms. Users can install and customize these platforms to build their own data management websites. Both platforms offer a full suite of cataloging, publishing, and visualization features that allows users to easily share data with the public. However, the query interface requires several clicks to get to the stored data, and comparison across multiple datasets is limited. With DataCenterHub, scientific data, experiments, and file collections are linked to each other in a clear way, with instant, direct access to files, and user-friendly query, exploration, and comparison capabilities across datasets.

SciServer[10] is a fully integrated cyberinfrastructure system which enables researchers to share and analyze big data. It allows researchers to work with terabytes or petabytes of scientific data, without needing to download large datasets. This system provides significant data analytics capabilities, and users need specialized knowledge to use the tools and querying features.

BACKGROUND

For nearly two decades, Purdue University's HUBzero cyberinfrastructure[11] has enabled web-based collaboration, educational outreach, and sharing of tools and resources for more than 50 hubs across diverse knowledge domains, with a combined audience of more than three million visitors per year. HUBzero provides a complete development and deployment environment for creating and publishing web-based computational tools—with job submission to XSEDE,[12] DiaGrid,[13] and local clusters for high-performance computing. HUBzero users can also upload their own content and manage access, tagging, and presentation through user-friendly self-guided services. Content consists of educational and outreach resources, training courses, publications, and presentations. HUBzero also offers wish lists (for requesting features), ticketing (to report problems and track solutions), usage metrics (to identify usage patterns), discussion forums, and reviews.

In 2009, we developed data components for the HUBzero cyberinfrastructure to serve as a foundation for creating customized data solutions. The components provided advanced technologies to support a full range of data capabilities to define, create, populate, explore, and maintain custom-built medical and scientific databases that included structured data and repositories of files.

Our data technologies support

- *data collection*—customized forms, formatted spreadsheets, annotation, and tagging for data and files, processing to import and merge data from external sources; and

- *data exploration*—customizable joining and processing of data from across databases and file repositories for display as interactive tabular "dataviews," where users can search, audit, review, sort, and download data and analytics, with extensions for graphs and maps.

Our data solutions offer an integrated environment for databases, repositories, tools, and resources that support a collaborative end-to-end research workflow. The open architecture of the data components facilitates the creation and seamless integration of new services and technologies required for new kinds of research data. For example, we added data auditing and data-completeness algorithms to our collection framework when needed for clinical patient data, and we added map support to our data exploration framework for earthquake disaster data.

Table 1 identifies some of the database systems we designed, developed, and delivered to support collaborative research. The database systems combine standard and customized components of our data technologies.[14–31]

In 2014, we began exploring the concept of self-serve databases for the HUBzero cyberinfrastructure. A prototype was developed as part of our work on NEEShub.[32] Our work with researchers on customized data solutions for medical and scientific databases, together with our work on the NEEShub self-serve database, was important preparation for development of our data sharing platform, now available at DataCenterHub (datacenterhub.org).

In Figure 1, a DataCenterHub dataview shows the integration of disaster data collected by reconnaissance teams on buildings impacted by the 2015 Nepal earthquake. The dataview offers building name "drill-down" to details for building properties and damage assessment (structured data); links to file repositories that include images, drawings, and other reconnaissance data (file collections); and analysis results (vulnerability parameters). This dataview joins heterogeneous data types together in a single interactive interface for investigating infrastructure resilience and disaster preparedness.

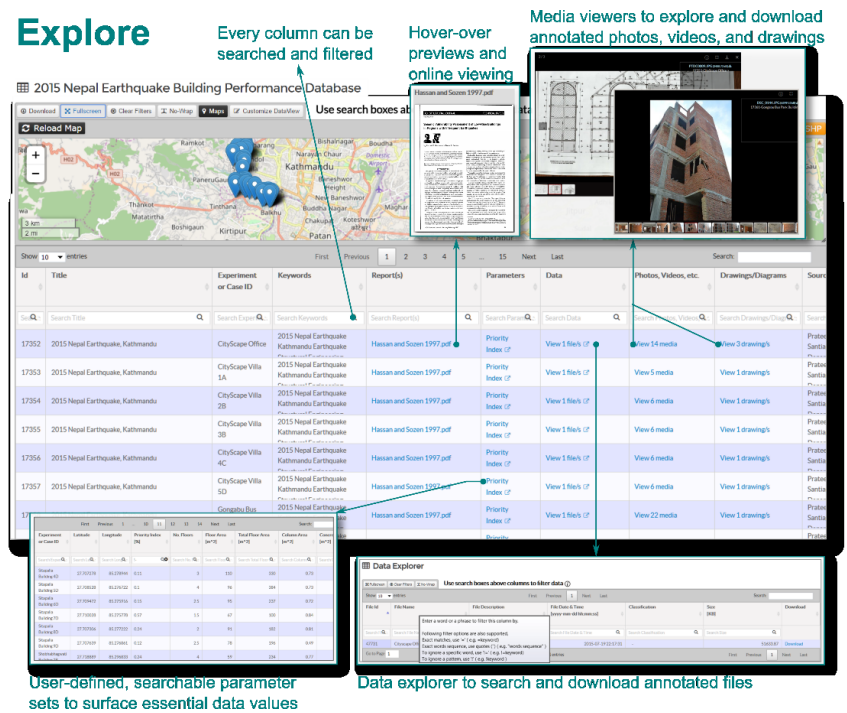

Figure 1. Exploring data collected by reconnaissance teams after the 2015 earthquake in Nepal. The dataview displays building information; links to launch viewers for files such as photos, videos, drawings, and reports; a map with markers that display metadata; and key building vulnerability parameters that are autolinked to building coordinates. DataCenterHub generates a merged view of heterogeneous data types for browsing and searching datasets.

Table 1. Some of the customized data environments built with the data platform at HUBzero.

Hub	Database	Community	Impact
Disaster-HUB	2010 Chile Earthquake, 2011 Joplin Tornado, 2013 Moore-Newcastle Tornado[14]	Civil engineering, NIST, NEES	NIST recommendations for improving preparedness and resilience of communities (US government building codes, standards, practices)
NEEShub	2010 Haiti Earthquake[15]	Civil engineering, NIST, NEES Academic and professional groups	Researcher access to raw and derived data for experiments, reconnaissance, and other scientific datasets, also providing search, analysis, and visualization
	ACI 369, ACI 445[16–18]		
	JEE and ACI journal publications[19,20]		
	Performance databases: SAC, Shear Wall, Shear Wave Velocity[21–23]		
cceHUB	Thymic malignancies Retrospective and prospective clinical data and analytics[24]	150 hospitals worldwide, thoracic surgeons, researchers, oncologists, clinicians	Staging, treatment, survival, outlier, prediction, and other studies for a rare cancer
	Pediatric HIV disclosure intervention[25]	HIV/AIDS research	Psychosocial interventions for affected children and their caregivers in resource-limited settings
	SafeRx database for investigation and research into adverse drug events[26]	Medication safety and pharmaceutical research groups	Adverse drug event analysis for data imported from the FDA and other sources to advance drug safety
pharmaHUB	Pharmaceutical excipients[27,28]	Drug manufacturers for quality of design and process	Material property variations for vendors and lots affecting manufacture of tablets
nanoHUB	Solar photovoltaic[29]	Solar PV researchers and engineering students	Optimization for design of residential distributed solar PV
CatalyzeCare	Infusion pump informatics (IPI)[30,31]	320 hospitals nationwide, clinical pharmacists, nurses, medication safety officers	Investigations of IV delivery errors, metrics, pattern, and trend analysis, comparative analytics. Hospital pump libraries are set and tuned based on analysis using IPI

THE DATA PLATFORM

The architecture and design of our data platform is based on technologies developed as Content Management System (CMS) components for the HUBzero cyberinfrastructure and used to create customized research databases. These components supported data-sharing projects across many medical and scientific disciplines, with advances to the data infrastructure as new features and capabilities were needed. Our experience building infrastructure for collecting, viewing, and analyzing data provided valuable insight during the design process for our DataCenterHub data platform. We reused key modules, features, and capabilities from our existing components. We also applied knowledge and understanding for what research data look like and how to make data usable across targeted research communities.

Several fundamental concepts drove our platform's design. The first was that research data are more than just collections of files with metadata. Most research activities involve numerous experiments in which an object or subject is exposed to input or stimuli and its response is recorded. Experiments are described by both structured and unstructured data. Experiments might be associated with collections of files of various classifications and types (photos, videos, reports, data, drawings, images, all stored in many different formats), with annotations and other associated metadata. Experiments might also include parameter sets with numeric or text data, links, and images that identify and describe properties, methods, equipment, outcomes, and computational results. To meaningfully preserve and present research data, the structured and unstructured data, metadata, files, and parameter sets should be organized to associate them clearly with experiments.

Second, a data sharing platform should offer a simple and straightforward way for users to contribute their data, with features that guarantee ease of use even for complex or big data. If data upload is difficult or time consuming, potential users might not want to use the system and valuable data will not be preserved for sharing and discovery.

The final concept governs the presentation of data for discovery and exploration. We believe the research community is best served when the distance between users and data is minimized. Research data should not be hidden behind keyword searches and high-level queries. Access to photos, reports, images, and parameter data should not require downloading zip files before content can be investigated and found useful. Instead, data should be displayed as it was organized during the research activity, as experiments, with associated data fully accessible via online exploration. Experiments should be connected to their data and file collections, with links that launch interactive tools for viewing files and drill-down capability for viewing experiment parameter details. All experiments belonging to a well-defined research activity should be collected together in a dataset, with restricted access during dataset creation and researcher-specified access for discovery at publication. The research community should then be able to navigate, search, and investigate published datasets though interactive web interfaces tailored for presentation of experiment data, metadata, parameters, and file types.

Our goal was to develop a platform for sharing scientific datasets that provides

- organization of data by experiments described by metadata, collections of files, and parameter sets containing key experiment data values;
- user customization of parameter sets to allow specialized uses of the platform, with requirements for metadata necessary to interpret the parameters, such as data field definitions and units;
- reuse of parameter sets to facilitate standardization and consistency across experiments conducted by independent researchers;
- organization of file collections by data type for file classification at upload and automatic interpretation of data type for launching viewers;
- easy ingest of large files and large collections of files;
- automatic metadata for files including classification by dataset, experiment, and data type; and extraction of keywords from content for use in search and filtering;
- annotation and other file-management operations in bulk;
- spreadsheets or web entry to upload and update metadata and parameter sets;
- dataset sharing by team members;

- publication for discovery as public or restricted datasets;
- direct access to view and explore datasets using an extensible tabular interface with interactive viewers that interpret data type for advanced search, preview, and comparison, both within and across datasets;
- BagIt archives for datasets, with Dublin Core metadata;[33] and
- a single, comprehensive interactive dashboard that helps users define experiments, connect them to research products, and publish them for discovery and exploration.

Terminology and Standards for Data Organization

The basic construct for scientific research data is an experiment or case. Users contribute data by creating a dataset and defining experiments that belong to the dataset. Each dataset has an identifier, title, creation date, and publication date. A dataset can be updated and managed by the dataset creator together with invited researchers who are assigned member roles.

Each experiment in the dataset is described by bibliographic metadata, file collections classified by data type, and parameters describing key properties and results.

Metadata supplied by users for each experiment consists of title, experiment identifier, source, keywords, start/end date, latitude, longitude, and compilation information. Definitions and examples of experiment metadata are given in Table 2.

File collections corresponding to an experiment are categorized by type: Report, Data, Photos/Videos, and Drawings/Diagrams. The categories were defined so that experiment file collections could be more easily understood and navigated. Figure 4 (top) shows how file organization is viewed by the user. Each experiment file is assigned an identifier and associated with a filename, upload timestamp, size, and extension. Files can be annotated by description and classification for more fine-grained file searches.

A parameter set of searchable structured data can be created for each dataset, and each experiment in the dataset can be assigned its own parameter values. Parameter sets are described in tabular (spreadsheet) format, with name, units, description, and data type for each parameter. Users can define their own parameter sets, or they can select a predefined parameter set. We expect standardized parameters to be defined by scientific discipline and experiment type. As research communities investigate data on DataCenterHub, standards for parameter sets could be discussed, vetted, and agreed upon, offering researchers better ways to compare data across experiments and datasets, with less variation in conditions, characterization, and measurement units, and better reproducibility.

DataCenterHub uses dataviews to present datasets for discovery and exploration. Dataviews are tabular displays with 1) column headers and descriptions, 2) search boxes for filtering and matching data in a column, 3) data types for each column that determine how data can be searched and viewed, for example, range search for numeric data, 4) navigation through datasets and experiments, 5) specialized view operations such as maps, download, and user customization, and 6) columns that can launch new dataviews for each data row so that further dimensions of the data can be explored.

The Discover dataview at DataCenterHub presents all published datasets for discovery. Dataviews are also used to present experiment parameter sets, and repository files and their metadata for collections of type Data and Report. The data platform offers type-specific viewing tools for file categories, such as media viewers for photos, videos, drawings, and diagrams. Online document viewers are provided for Reports. Files of type Data generally contain measurements, observations and other results. For some Data files, computational tools can be launched using the file as input.

All datasets are published with a *resource* (or splash) *page*. Users can add descriptions to their resource page, which is linked from the datasets Discover dataview through the dataset title. A resource page has its own dataview that presents only the experiments and data for that dataset.

Table 2. Experiment metadata definitions and examples.

Metadata	Definition	Dataset 1*	Dataset 2**
Dataset title	Name of project, group of experiments, simulations, or studies	2015 Nepal earthquake Building Performance Database	RNA sequences and assemblies of Varroa jacobsoni (honey bees varroa mites)
Experiment title	Name of experiment, simulation, or study	2015 Nepal earthquake, Kathmandu	Solomon Islands Vj mites
Experiment ID	Identifier assigned by source to sample, specimen, case, survey, site, experiment, or simulation	Kapan School 1	S7
Source	Names of the people who generated the data	Prateek Shah, Santiago Pujol, Department of Urban Development and Building Construction (DUDBC Nepal)	Denis Anderson
Keywords	Words describing the dataset	2015 Nepal earthquake, structural engineering, reinforced concrete, priority index, building performance, moderate structural damage, 5.5 stories	Ugi Island, Makira Province, Apis cerana, mite, V. jacobsoni, honey bee, RNA later
Start date/end date	Start and end dates of the period when the experiment took place	2015-06-18 / 2015-07-01	2010-04-08 / 2010-04-08
Location (latitude, longitude)	Decimal coordinates of location where data are collected, if applicable	27.697936, 85.312817	–10.520373, 161.827497
Compiled by	Names of people who compiled dataset on DataCenterHub	P. Shah, A. Puranam	G. Andino
Compiled date	Date the dataset was compiled on DataCenterHub	2015-07-14	2015-08-28

*Investigation of reinforced concrete buildings in the aftermath of the 2015 Nepal earthquake.

**Sample collection of mites from honey bees, with raw RNA sequence reads and newly discovered transcript assemblies.

Example Workflow for Data Upload and Discovery

This section describes a workflow for preserving data collected in the field following the 2015 Nepal earthquake. The research team surveyed 150 damaged and undamaged buildings, collecting hundreds of photographs and drawings in addition to damage reports and numerical models. We describe the steps followed by the team to create their dataset, and then show how the broader earthquake engineering community can use DataCenterHub to discover and explore the collected information.

Only a few of the many features of our data platform are covered in this example. A detailed workflow including more features is available at datacenterhub.org/resources/30.

Data Upload Workflow

To create a dataset, the user first registers at DataCenterHub. After logging in, the user clicks Upload to start a dataset and enters the dataset name. The user is presented with the dataset editor (Figure 2), which consists of three sections:

1. *Experiment or Case Information*, where users enter bibliographic metadata.
2. *Experiment or Case Files*, where users upload file collections.
3. *Experiment or Case Parameters*, where users define and enter searchable experiment properties, outcomes or other structured data.

The user can invite other researchers to help upload and vet the dataset using the Share feature.

Enter Experiment or Case Information

The research team organized the data they collected during the reconnaissance effort by building. Therefore, they organize their dataset by defining each building impacted by the earthquake as one "case."

For small datasets, users can enter cases and their metadata directly into the Experiment or Case Information web form (Figure 2). For the Nepal dataset, the Spreadsheet Upload feature is used to load information for all buildings at once. This feature provides users with a formatted spreadsheet template for entering cases and their metadata, and is especially useful when large amounts of data already exist in other spreadsheets or when metadata (bibliographic information such as source and temporal information) are repeated across many cases.

For the Nepal dataset, the team had already organized much of their building information in spreadsheets. They copy and paste data from their existing spreadsheets to the spreadsheet template, and this template is then uploaded to define all cases for that dataset. If changes are needed, users can either edit information directly in the web interface or use Spreadsheet Upload to get an updated spreadsheet template (now containing all currently uploaded data), revise the data, and upload again.

Upload File Collections

With dataset cases defined, the user is ready to upload files for each case. For this dataset, the file collection consists of a report about measures for building vulnerability, photos documenting damage to the buildings, numerical models for the buildings, and drawings of floor plans from the field survey. Users can upload folders and files through the web interface ("Local" upload) or use an SFTP client to transfer files to the DataCenterHub location designated for large-file and bulk-file uploads ("Server" upload). During the upload process, users can also annotate files with metadata such as description and classification.

For Nepal data collection, the researchers organized their files on their desktops in building-specific folders, with each folder divided into subfolders corresponding to the DataCenterHub file types (Reports, Data, Photos/Videos, Drawings). They use an SFTP client to transfer the entire collection to DataCenterHub, and the Server upload feature is then used to assign each building folder to the appropriate building case and file category. Because the final report is shared by all cases, the Shared Upload feature is used to upload the report and assign it to all cases at once.

The research team can use the many features of the Files Upload section to make their upload fast and easy by using a combination of Server Upload (when there were many files per case, such as photos) and Shared Upload (when a single report file was shared across cases).

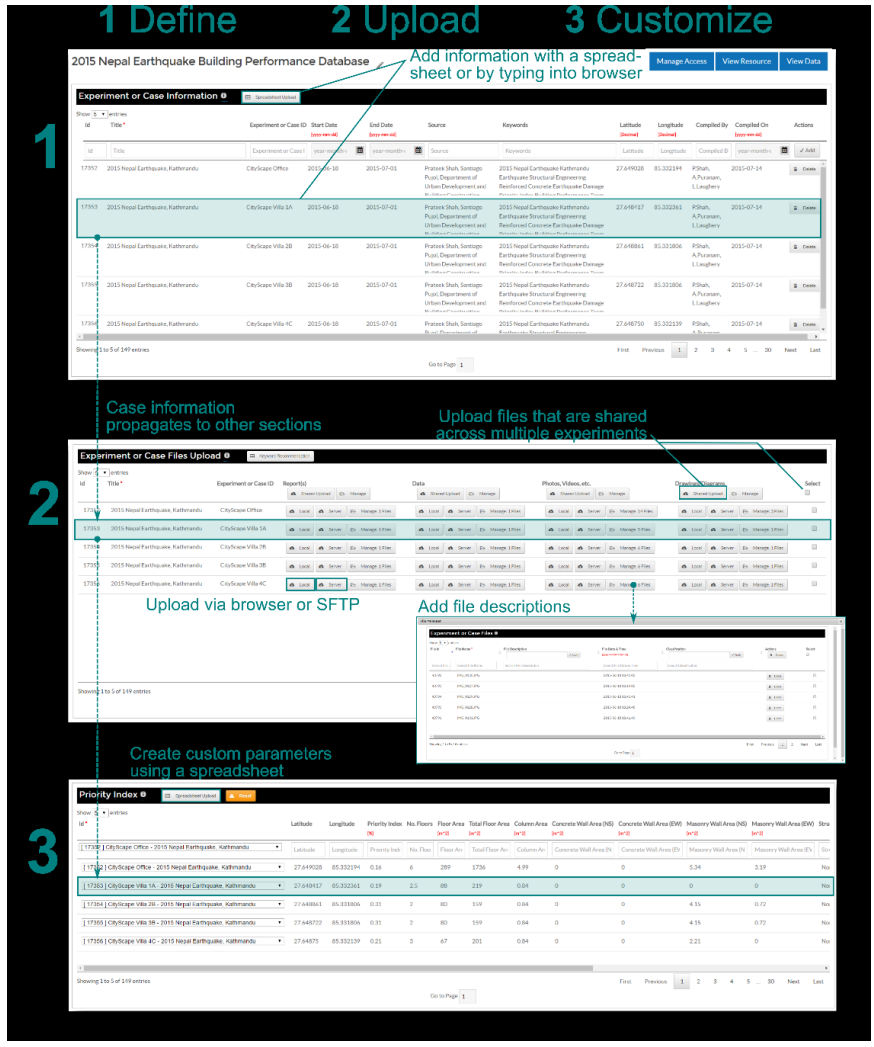

Figure 2. Workflow for uploading and editing a dataset: (1) Use a spreadsheet to define your experiments; (2) upload your data, reports, and media files for each experiment; and (3) use a spreadsheet to create custom searchable parameters.

Enter Parameters

The users are now ready to define the parameter set where important data values assigned to each building case can be "surfaced" and searchable. Parameters of interest include building characteristics (for example, number of floors, floor area, areas of columns and walls), calculated values (for example, column area as a percentage of total floor area), level of structural damage, and a computed measure for a building's vulnerability to damage named "priority index." Users can create their own parameter set or choose a predefined one. A research group at DataCenterHub had already defined a "Priority Index" parameter set for earthquake datasets, and it contains all the parameters of interest. This parameter set is selected for the Nepal dataset.

As with Experiment or Case Information, users can type their data values directly into the Parameters web form or they can upload data using a Parameters spreadsheet template. For our example dataset, the users fill the Priority Index spreadsheet template with parameter set values, one row of parameter values per building case. This template is then uploaded to the dataset using Spreadsheet Upload.

Publish

When completed, the dataset is submitted for publishing. As part of submission, users specify access restrictions (if any), create a splash page with a short abstract, and select a license. After a brief review process, the dataset is made available for discovery.

This dataset was available worldwide for researchers to investigate only a few weeks after the Nepal earthquake reconnaissance was completed.

Discover and Explore Data with Dataviews

Any visitor to DataCenterHub can click Discover to launch the Datasets dataview. All published datasets are displayed in the view, with one experiment or case per row. The dataview presents dataset title, experiment title, experiment metadata, links to file collections by category, and links to experiment parameters. Users can understand from the start how datasets are organized. Experiment data are immediately visible and the ways to navigate, search, and explore data are intuitive.

For our example discovery workflow, earthquake engineers interested in finding an experiment from any public dataset listing “earthquake” can type this as a search phrase in the keywords search box. At the same time, search text can be entered in the dataset or experiment title search boxes. For example, users can type “Nepal,” “Haiti,” “Taiwan,” or “Chile” to investigate whether datasets related to other earthquake reconnaissance efforts have priority index data available in their parameter set.

There are two datasets at DataCenterHub for the Nepal earthquake. Searching the Datasets dataview for “Nepal” in the dataset or experiment title and “priority index” as a keyword locates this dataset.

In the Nepal dataset, engineers can navigate to the parameter set and investigate building parameters. They can sort the buildings by number of floors and filter for severe damage, or they can search the priority index values for specific ranges to identify the most vulnerable buildings. For the buildings of interest, they can then view annotated photos with the media viewer to see the actual damage sustained.

Characteristics of the buildings, including wall and column dimensions, number of stories, and level of damage are all available in the parameters view for comparison. Building plan drawings that have more information about the structural members can be investigated with the media viewer.

The organization of the dataset as cases (buildings) provides engineers with a straightforward way to navigate, investigate, and analyze reconnaissance data, case by case (Figure 1). Dataviews and attached specialized viewers (media viewer for photos, data file explorer, parameter viewer) offer unique ways to search, view, study, and compare the structured data, unstructured data, and repository files across multiple cases and across datasets

ARCHITECTURE AND IMPLEMENTATION OF THE DATA PLATFORM

In this section, we present the data platform architecture, capabilities, and implementation. The data platform is built on top of the HUBzero cyberinfrastructure, which uses the LAMP (Linux, Apache, MySQL, PHP) stack. Figure 3 depicts the system architecture at DataCenterHub.

The User Interface layer is supported by the Data Platform layer, which consists of Create and View components and the organizational structure of the repository. At the bottom is the HUBzero system infrastructure. The Create and View components consist of operational building blocks that support the features of the user interface for data collection and data exploration.

From the data upload point of view, our platform operates on a collection of datasets. Each dataset consists of a set of related experiments that are grouped and managed together. For data discovery, the platform allows more fine-grained access at the experiment level, which facilitates easy search, sort, and comparison capabilities among experiments across datasets.

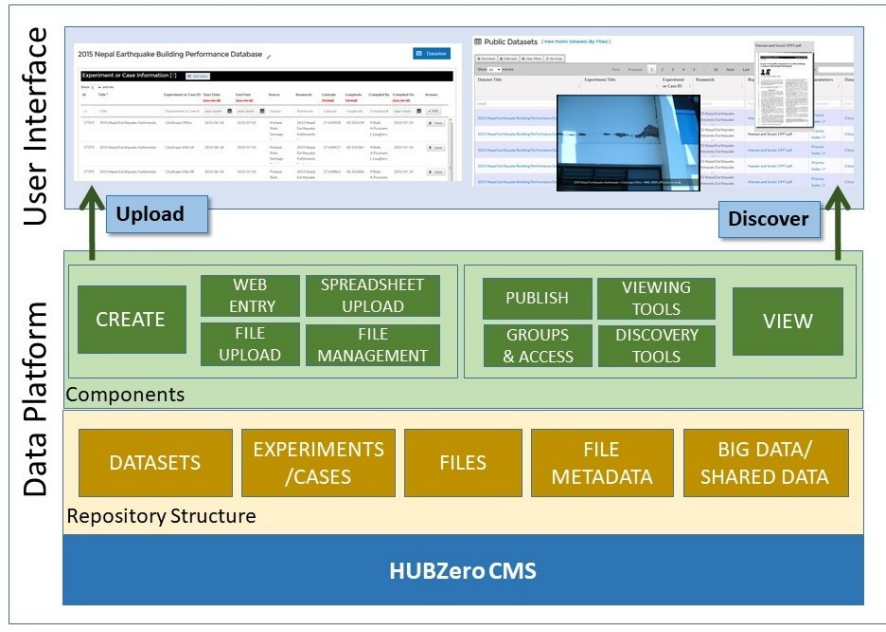

Figure 3. Architecture of the data platform.

Dataset Structure and Implementation

Each dataset is assigned a unique ID that is used to locate content in the database and files in the repository. At the highest level, a dataset consists of a list of experiments, a collection of files, and a parameter set. Each experiment defined for a dataset is assigned an ID—this identifier is unique across the entire data platform, since an experiment can be individually accessed and explored through the discover dataview, not just as part of the dataset where it was defined. Similarly, files are assigned unique IDs when uploaded to experiments.

The identifiers for datasets, experiments and files are the principal locators for

- updating data and metadata in the database;
- storing, accessing, and managing files in the repository;
- presenting and linking information in the dataviews; and
- downloading datasets as annotated archives.

File collections (categorized by data type) can be associated with a single experiment or shared across multiple experiments. A parameter set is defined for the dataset, and each experiment in the dataset is assigned its own values for each parameter.

Repository structure

The file repository is organized by datasets. Each dataset directory consists of three subdirectories: Files, File Metadata, and Definitions (Figure 4, middle). The “Files” directory stores the files, first grouped by category (Reports, Data, Photos/Videos, Drawings) and then by the experiment ID according to the following structure:

```
<DatasetID>/Files/<FileCategory>/<ExperimentID>/<FileName>
```

Storage for each experiment directory is "flattened," and therefore, filenames within a single experiment directory must be unique. The flattened structure makes the assignment of file metadata more user-friendly in the data collection interface, and also ensures that the operation of discovery dataviews and tools is efficient and effective for data exploration.

An additional directory above the experiment level supports the sharing of files across the experiments. As shown in Figure 4 (middle), files shared across experiments are physically stored in the Shared Files folder and linked symbolically from the experiment directories.

The File Metadata directory stores any metadata generated for the files, such as thumbnails and video previews. The Definitions directory stores the JSON format description of dataset constructs used to generate dataviews, including experiment views, file views, and views for predefined and user-defined parameter sets.

Database structure

There are three main tables that store data related to the three main components of a dataset: experiments, files, and parameters. Contents of these tables are populated with the information entered or uploaded by users and/or automatically extracted and assigned by the data platform. A new MySQL database is created for each new dataset, and data relevant only to that dataset is mapped using a MySQL view linked to the respective table in the main database. The database's organization is shown in Figure 4 (bottom). The contents of the main tables are as follows:

- *Experiments table:* dataset ID, experiment ID, and experiment/case bibliographic information such as title, source, and keywords.
- *Files table:* dataset ID, experiment ID, file ID, and other file information such as filename, file category, description, keywords, classification, size, and upload timestamp.
- *Parameter table:* this table schema is automatically defined from fields and values specified by the user when creating the parameter set for that particular dataset; therefore, parameter table schema and contents vary from dataset to dataset.

When creating or updating datasets, the platform will access the corresponding dataset level database to read and store data. When viewing and exploring datasets, the platform will access the main database tables to read data, since data discovery is done at the experiment level.

Data Entry and File Upload

For small datasets with a limited number of experiments and files, it is easy to enter the data and upload the files through the web interface. Using the Experiments or Case Information web form, experiments can be added to the dataset one at a time. For each new experiment, an entry is added to the experiments table, the experiment is assigned a unique system-generated ID, and repository space is allocated to store the files.

Using the Experiments or Case Files web form, users can upload files from their local machine to a desired experiment in the appropriate file category. They can upload multiple files or folders at a time by dragging and dropping the files/folders using the Local button (Figure 2, Upload).

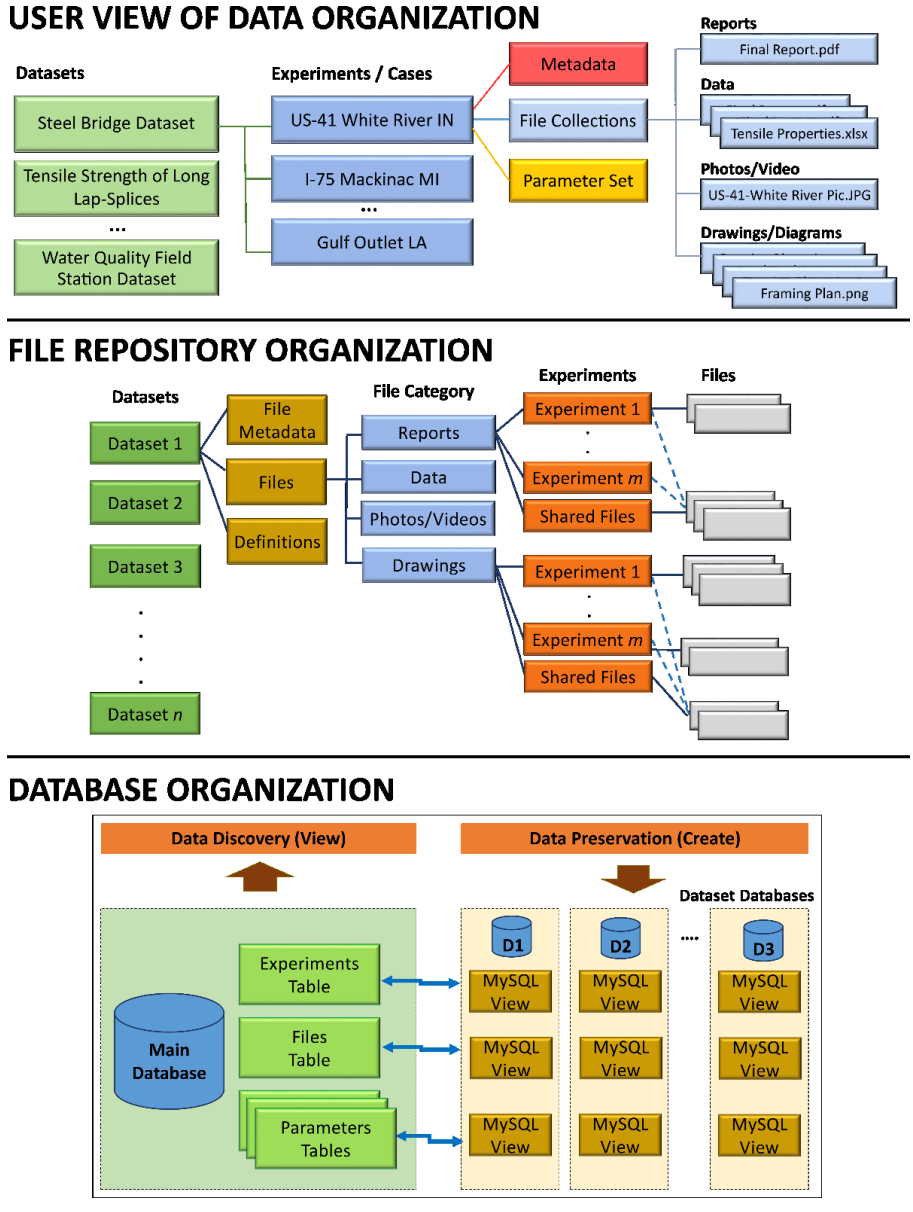

Figure 4. Dataset organization: User view of the data (top), file repository organization (middle), and database organization (bottom).

Parameter values for the experiments can be entered using the Experiment or Case Parameters web form. If none of the predefined parameter sets correspond to the dataset experiment parameters, users can create custom parameter sets that correspond to their data.

Although the database schema and repository structure for storage of files, data, and their relationships are simple and straightforward, complexities arise as a result of data platform support for variations in the ways researchers need to upload data and specify metadata. This is particularly true for uploading and associating files to multiple experiments or associating metadata to multiple files, especially when the number of experiments or files is very large.

For small datasets, direct web entry of data and web upload of locally stored files work well for creating and managing datasets, but small datasets account for less than 25 percent of the data at DataCenterHub. The next two sections discuss the spreadsheet and shared file upload processes that support the creation of large datasets with hundreds of experiments and thousands of files.

Spreadsheet Processor

Scientists collect and store their research data using spreadsheets; this format is nearly universal across all scientific disciplines. It makes sense to support a data preservation workflow where researchers' spreadsheets can be uploaded and processed to populate database tables that store searchable structured data.

The spreadsheet processor at DataCenterHub supports spreadsheet upload for

- bibliographic metadata for experiments in Experiment/Case Information, and
- parameter set data for Experiment/Case Parameters.

For Experiment/Case Information, the spreadsheet headers and data types are determined by our fixed bibliographic metadata. Users download a spreadsheet template for the dataset with all existing experiment metadata, and then, this spreadsheet can be used to add new experiments, update existing experiments, or delete existing experiments. The spreadsheet processor validates the file type, file format, and data types before storing the data to the experiment table.

Spreadsheet processing for user-defined Experiment/Case Parameters is more complex. Users provide a name for their parameter set, then upload their own spreadsheet with column labels naming their experiment parameters. The data platform creates a new database table for their data with column labels as fields. The spreadsheet must contain rows for parameter names (column labels), descriptions, and units. This should be followed by one or more rows of data, since the data are used by the spreadsheet processor to detect data type for each parameter (Figure 2, Customize). Once the table is created, the process for add/update/delete is the same as for Experiment/Case Information. The data platform provides interfaces to review the parameter set (labels, definition, units) and users can remove and redefine the entire parameter set for their dataset as necessary.

Upload Support for File Collections

In DataCenterHub, there are two ways to upload files to a dataset. Using the Local option, users can interactively upload files and folders through the web interface. Using the Server option, users first upload the files and folders to a temporary location at DataCenterHub via an SFTP client and later transfer the files to the desired datasets. The Local option is used when files are small (<5 GB) and the number of files to be uploaded at a time is not large. The Server option is used when the number of files to be uploaded is very large or when file sizes exceed the upload limit supported by browsers (for example, genomic data). Both methods follow the file repository structure shown in Figure 4 (middle).

When compiling large datasets, it is sometimes necessary to associate files with multiple experiments. Using the Shared Upload feature, users can upload files across multiple experiments. After selecting some (or all) experiments, users either click the Shared Upload button to select the files or drag and drop files and folders to the Shared Upload button (Figure 2, Upload). The file collections loaded to the dataset through the Server option can also be assigned to one or more experiments selected by the user. Files shared across multiple experiments are stored in the shared files directory and associated with the selected experiments via symbolic links (Figure 4, middle).

File metadata such as file size, file type, and upload timestamp are automatically captured and saved in the files table. If the files are organized into a folder hierarchy when uploaded, the file path information will also be automatically captured as file classification metadata. Users can enable automatic keyword processing to extract keywords from uploaded report files and associate them as metadata to the experiments. After uploading the files, the File Manager can be used to enter or modify additional file information such as annotations. Adding or modifying file metadata can be done for single experiments or for multiple experiments at a time when files are shared (Figure 2, Upload).

Implementation and Extensibility of the Dataview

The View component is designed to access data stored in a MySQL database and present the data in an interactive, tabular display (Figure 1). The dataview presentation is determined by applying the rules of a "data definition" language that defines each column of the tabular display by identifying database table and field for the source of the data, and imposing data format, display type, display text, and display operations that are given as arguments to the column in the data definition. The data definitions and dataviewer component at DataCenterHub are a customized version of our general dataviewer CMS component.

The data definition generally identifies a principal database table with optional primary key, and permits any number of table joins that identify and match how selected table fields are to be linked to the principal and/or joined table fields. Column definitions can then refer to any of the primary or joined table fields to place data in the columns. Duplicates are handled with the "group by" rule. The data definition language is extremely flexible and has been designed in an extensible way that allows the introduction of new features as needed. Among the many features of the data definition language are the following:

- specification of MySQL queries (of any complexity) as part of any column definition or as a final where clause in the data definition;
- introduction of any number of "raw" data columns to the view that do not exist in the database, but instead are computed using data from other columns;
- formatting of columns with user-customizable labeling, clickable hyperlinks, and data alterations for display purposes; and
- data typing that determines the display properties or operations that can be performed on the column data when presented in the browser.

It is the last feature that gives the dataviewer its unique and powerful character. Data typing can specify that a filename (data from the MySQL database) should be presented as type "image"—this displays the image thumbnail in the dataview column with clickable options to view the full image or download the file. Data typing can specify that a collection of image and video files should launch a media viewer, including calculation of file count and presentation of "View <#files> Media" in the column as a clickable link (media viewer in Figure 1). Columns can present clickable text that generates a new drill-down dataview for each row in the column, extending the original tabular display to any number of dimensions for data through drill-down linkage (for example, parameter set dataview in Figure 1). Columns can be defined to launch visualization tools (for data files), present hover-over PDF images (for report files), and link to external sites. Data definitions can be written by a developer or autogenerated by the system according to the inputs obtained through the user interface.

Data definitions are stored in two formats: a PHP version to support ongoing definition updates, and a JSON version that is interpreted by the dataviewer for secure presentation in a web browser. Dataviews can use client- or server-side processing for gathering the data to present in the browser. This can be set in a data definition or configured as a default value across all dataviews, such as switching to server-side processing when 50,000 or more rows are present in the dataview. The performance of the dataview (based both on number of rows and complexity of column operations) is used to determine whether client- or server-side processing should be used.

At DataCenterHub, data definitions are used to describe and display the datasets "Discover" dataview, where all columns are predefined by the experiment metadata; and also the user-defined Parameter Set dataviews, where columns and data types are defined in real time by end users. Parameter Set dataviews are autogenerated when any new parameter set is created.

All dataviews have interactive exploration features that are automatically part of every tabular display. Columns can be sorted and searched according to the column data type. A column search box above and below each column supports text, numeric or date search across all rows of data in the column. Text search supports filtering of column data by patterns to match or ignore. Numeric search supports arithmetic and pattern filtering, including =, >, >=, <, <=, !, =! and range ("<low> to <high>"). Dataview buttons provide functions such as Clear Filters (to remove

multiple column filtering) and Download (to save the dataview data to a spreadsheet). Dataviews have titles, a cross-column search capability, and construct for placing configurable text and/or links to display on the dataview web page (such as for instructions.)

Access Control

At DataCenterHub, we establish and control access privileges separately for dataset creation/update and for discovery/exploration of dataset experiments.

Users need to be registered in DataCenterHub to create datasets, and the creator of a dataset can share dataset editing and file upload privileges with other users. During the creation stage only invited members can make changes to the dataset or view the dataset. Users can be invited as managers or as members. The manager role has all the privileges of the owner of the dataset, with the exception of deleting the dataset. Any manager can edit, share, and publish the dataset. A member can edit the dataset, but does not have permission to invite other members or to publish the dataset. Although multiple users are allowed to collaborate on editing a dataset, no concurrent modifications are permitted; only one user can edit the dataset at any given time. All managers and members can view the current state of the dataset at any time.

When a dataset is complete, the owner or managers can publish it with either unrestricted public access or with restricted access. If a dataset is published with unrestricted access, the dataset is added to the list of public datasets and any visitor can view it without registering or logging in at DataCenterHub. If the dataset is published with restricted access, viewing privileges will be restricted to the members of a specified group of users. These users must be logged in to view the dataset.

Data Download and Archive Package

Any visitor browsing the public datasets at DataCenterHub can download individual files from dataset experiments. Users can also download experiment metadata and parameter sets as spreadsheets using the Download button on the dataview. In addition, registered users at DataCenterHub can download the complete dataset as a BagIt “bag.”

BagIt is a hierarchical file packaging format introduced by the Library of Congress and is used primarily for storage and transfer of preservation-quality digital content.[33] BagIt bags consist of a payload (the dataset encapsulated in the bag) and tags (the metadata used to record bag transfer and storage.) The payload of a BagIt bag created for a DataCenterHub dataset consists of experiment bibliographic metadata (CSV file), all experiment parameter values (CSV file), and all files uploaded to that dataset. Files are packaged in the same repository structure as when stored in the system. A document describing the organization of the dataset experiments, parameter set, and file types is included in the payload. The bag also contains an XML format file (tags) describing the experiment bibliographic metadata according to the Dublin Core Metadata Initiative (DCMI) metadata terms.[34] Once a bag is generated, users can download it to their desktop or client machine via SFTP. Downloaded dataset bags can also be stored to repositories designed primarily for longevity of data with long-term preservation and support for data formats.

SUPPORT FOR COMMUNITIES OF RESEARCH AND PRACTICE

The data platform at DataCenterHub was designed as a solution for uploading, sharing, and discovery of datasets across scientific disciplines. Although the platform is discipline-neutral, it has proven to be very effective at organizing engineering and scientific data.

Currently, more than 250 datasets are available on the platform, comprising nearly 50,000 experiments/cases and 8,000,000 files. Some of the topics covered are RNA sequencing , steel bridges, datasets from the network for earthquake engineering simulation, strong ground motions, earthquake reconnaissance, structural engineering experiments, geotechnical engineering experiments, agronomy, and forestry.

As part of the DataCenterHub development effort, we have worked with research groups at workshops, forums, and conferences to understand

- how they currently manage their data,
- how their data are organized,
- how their data and files are formatted,
- what challenges they face in sharing their data, and
- what features would be useful in a data sharing platform.

Current data management methods vary widely among researchers. Methods range from backups on external hard drives or NAS devices to commercial cloud storage. Data format, organization, and metadata also vary widely from discipline to discipline. Whereas our bibliographic metadata might be sufficient to describe an experiment for some disciplines, others require time- or methodology-dependent information to understand the data. For example, in agronomy, the history of crop rotation on a plot might be essential to understanding current production. To meet these metadata needs using DataCenterHub, researchers append additional experiment metadata to their user-defined parameter sets.

The number and sizes of files uploaded for experiments also vary across and within disciplines. Engineering readme files are generally less than 1 Kbyte, while RNA sequencing files (that is, FASTQ files) can be 100 Gbytes or more. Some civil engineering datasets consist of a single report and a small number of data files for each experiment, while other datasets have thousands of files.

To illustrate our ongoing engagement with user communities, we will describe how DataCenterHub is being used by the civil engineering community. The American Concrete Institute (ACI) uses DataCenterHub to improve the design of reinforced concrete members. Two ACI subcommittees use DataCenterHub to share datasets of past structural tests of reinforced columns and walls. The datasets include information about specimen dimensions, material properties, reinforcing details, and peak capacities as well as photographs and reports. ACI datasets are used by structural engineers to evaluate current expressions for the strength and deformability of reinforced concrete elements under different loading conditions. When new expressions are proposed, the datasets are used to evaluate them. Earthquake engineers use DataCenterHub to evaluate and improve indices and methods that measure building vulnerability to earthquakes. ACI sent teams to conduct field surveys of affected buildings after the earthquakes in Kathmandu (2015), Taiwan (2016), and Ecuador (2016). More than 400 buildings were surveyed, and the collected data were published at DataCenterHub in a matter of weeks. These data included building locations, structural drawings, measurements, photographs, and damage descriptions.

The Earthquake Engineering Research Institute (EERI) uses field data on DataCenterHub to supplement its earthquake clearinghouse GIS maps, which contain information about areas affected by earthquakes and reconnaissance data. Faculty researchers at universities in Auckland, Nebraska–Lincoln, and Purdue have contributed earthquake simulation datasets. The structure and bridge engineering laboratory at Yonsei University in Korea has contributed years of experimental data on perfobond rib shear connectors that can be used in structural members for bridges and buildings in steel–concrete composite applications.

DataCenterHub development focuses on serving the needs of its user community. Table 3 lists some features that were developed in response to our user community. New features continue to be developed in response to user requests and new types of research data.

Table 3. DataCenterHub features developed in response to requests from its user community.

User request	Feature added
Share creation and editing of datasets with selected users	**Dataset sharing**: Dataset creator can add user to dataset as either a manager or member. When being edited, dataset is locked for editing by other users but is viewable.
Restrict access to published datasets to selected group	**Restricted access**: Dataset publication allows users to identify a group for restricted access. A user must be logged in and a member of the selected group to see the dataset. Owners can make the dataset public at any time.
Search across all columns for public datasets	**Search all**: Search box above dataview offers text search across all dataview columns.
Support upload of 100-Gbyte files too large to be uploaded through the web interface	**Server upload**: File upload for dataset experiments allows users to transfer data to DataCenterHub with an SFTP client of their choice and then assign the files to one or more experiments. There is no restriction on file size.
Support upload of documents (such as reports) to many experiments at one time	**Shared upload**: Users can assign any number of experiments and upload files to all selected experiments. These files are uploaded just once and then associated with each selected experiment.
Shared file manipulation (view, sort, file deletion, and file annotation) for multiple selected files in one experiment and shared files across multiple experiments	**File manager**: When editing, users can assign metadata to multiple files simultaneously, sort files by name to find files more easily, and delete multiple files simultaneously, both within and across experiments for selected files.
Group discovery datasets by experiment titles	**Group by title:** On discovery page, users can toggle between grouping experiments by title (dataset experiments with same title appear once with link to show all) and showing each dataset experiment as a separate entry.
See photos and videos before downloading with toggle for annotations on viewing	**Media viewer:** Under the drawings and photos/videos data categories, users can view photos, videos, and annotations. Under the report category, thumbnail previews of the first page of a report are provided on hover-over.
Automatically extract files' folder hierarchy as they are uploaded, and store as file metadata	**Folder hierarchy extraction**: A file's parent directories can be useful classification metadata. When users upload files, file hierarchy is captured and stored in the dataset and then displayed in the discovery dataview. Users can also assign classification metadata to their files in lieu of folder hierarchy.
Sort files using metadata assigned by the dataset creator	**File explorer**: Users can search and filter experiment files in an experiment using descriptions, file extensions, sizes, and file classification assigned in the file manager by the dataset editors.

CONCLUSION

Cyberplatforms that enable the preservation and sharing of data provide an invaluable service for the advancement of research. We presented a new solution for the organization, upload, sharing, and discovery of data produced by scientific research. Datasets are organized by experiments, with a simple common structure for metadata, file collections, and key parameters. Researchers associate annotations, reports, media, measurements, observations, and outcomes to each experiment so that the relationship between an experiment and its data is clearly understood and the data can be accessed quickly and easily for search and exploration. Spreadsheet and bulk upload features make contribution straightforward and user friendly, even for datasets with large data. Interactive web interfaces with specialized viewers interpret data and file collections by type and use, so that researchers can investigate and compare data before downloading. Key parameters describing experiments are extracted and searchable within and across datasets for discovery and comparison.

DataCenterHub provides an alternative to existing discipline-neutral solutions, with the goal of helping as many researchers as possible classify and share their data and files for discovery and exploration. Nearly 50,000 experiments, 8,000,000 files, and 30 Tbytes of data have been contributed at DataCenterHub. Our extensible architecture allows new features to be added as needed, and our platform continues to evolve and adapt to an increasingly diverse community of users.

ACKNOWLEDGMENTS

This project is funded by the NSF CIF21 DIBBs Program (award 1443027). We would like to thank the following people for their valuable contributions to the project: NSF Program Director Amy Walton; members of the development team with major contributions: Sudheera Fernando, Sumudinie Fernando, George Howlett, Merve Usta, and Sai Chowdary Samineni; and senior project personnel of the DIBBs project: Line Pouchard, Ayhan Irfanoglu, Mohammad Reza Jahanshahi, and Lisa Zilinski.

REFERENCES

1. Science Board, “Long-Lived Digital Data Collections: Enabling Research and Education in the 21st Century,” National Science Foundation, Arlington, VA, Rep. NSB-05-40, Sep. 2005.
2. K. Rosenbloom et al. (2010). ENCODE whole-genome data in the UCSC Genome Browser. *Nucleic Acids Research, 38(suppl 1)*, pp. D620-D625. [Online]. Available: http://nar.oxfordjournals.org/content/38/suppl_1/D620
3. NCBI (2016). National Center for Biotechnology Information. *Website*. [Online]. Available: http://www.ncbi.nlm.nih.gov
4. Figshare. *Website*. [Online]. Available: https://figshare.com
5. Dyrad. *Website*. [Online]. Available: http://datadryad.org/
6. SQLShare. *Website*. [Online]. Available: https://sqlshare.uw.edu/
7. DataHub. *Website*. [Online]. Available: https://datahub.csail.mit.edu/
8. CKAN. *Website*. [Online]. Available: https://ckan.org/
9. DKAN. *Website*. [Online]. Available: http://getdkan.com/
10. SciServer. *Website*. [Online]. Available: http://www.sciserver.org/
11. HUBzero (2014). HUBzero powered sites. [Online]. Available: https://hubzero.org/sites
12. K. Towns et al. (2014, September). XSEDE: Accelerating Scientific Discovery. *Computing in Science and Engineering*. [Online]. *16*. pp.62-74. Available: http://ieeexplore.ieee.org

13. DiaGrid (2016). A Distributed Research Computing Grid Provided by Information Technology at Purdue RCAC. *Website*. [Online]. Available: https://diagrid.org
14. NIST (2016). NIST Disaster and Failure Studies HUB. *Website*. [Online]. Available: https://disasterhub.nist.gov/
15. N. Sedra et al. (2010, December). The Haiti Earthquake Database. *NEEShub Database*. [Online]. Available: https://nees.org/resources/1797
16. W. Ghannoum et al. (2012, September). ACI 369 Circular Column Database. *NEEShub Database*. [Online]. Available: https://nees.org/resources/3658
17. W. Ghannoum et al. (2012, September). ACI 369 Rect. Column Database. *NEEShub Database*. [Online]. Available: https://nees.org/resources/3659
18. C.E. Ospina et al. (2011, October). ACI 445 Punching Shear Collected Databank. *NEEShub Database*. [Online]. Available: https://nees.org/resources/3660
19. A. Elnashai, N.N. Ambraseys, S. Dyke (2010, July). The Journal of Earthquake Engineering Database. *NEEShub Database*. [Online]. Available: https://nees.org/resources/3166
20. J. Browning et al. (2012, March). The ACI Publications Database. *NEEShub Database*. [Online]. Available: https://nees.org/resources/3825
21. SEAOC, ATC, CUREE (2011, January). SAC Steel Project Database. *NEEShub Database*. [Online]. Available: https://nees.org/resources/2264
22. X. Lu, Y. Zhou, J. Yang, J. Qian, C. Song, Y. Wang (2010, December). Shear Wall Database. *NEEShub Database*. [Online]. Available: https://nees.org/resources/1683
23. NEES@UTexas (2011, January). The Shear Wave Velocity Profiles Database. *NEEShub Database*. [Online]. Available: https://nees.org/resources/2262
24. J. Huang, J et al. (2014, October). Development of the International Thymic Malignancy Interest Group International Database: An Unprecedented Resource for the Study of a Rare Group of Tumors. *Journal of Thoracic Oncology*. [Online]. *9(10),* pp.1573-1578. doi:10.1097/JTO.0000000000000269.
25. A.C. Catlin et al. (2015, November). Sankofa Pediatric HIV Disclosure Intervention Cyber Data Management: Building Capacity in a Resource-limited Setting and Ensuring Data Quality. *AIDS Care, 27(1)*, pp. 99-107. doi:10.1080/09540121.2015.1023246 2015
26. SafeRx Dashboard (2016). *cceHUB Database*. [Online]. Available: https://ccehub.org/saferx
27. K. Kuriyan, A.C. Catlin, and G. Reklaitis (2009, June). pharmaHUB: Building a Virtual Organization for Pharmaceutical Engineering and Science. *Journal of Pharmaceutical Innovation*, *4(2)*, pp. 81-89. doi: 10.1007/s12247-009-9061-7
28. T. Wang et al. (2013, July). The Creation of an Excipient Properties Database to Support Quality by Design (QbD) Formulation Development. *American Pharmaceutical Review, 16(4)*. pp. 16-25. [Online]. Available: http://www.americanpharmaceuticalreview.com/
29. X. Liu, D. Kearney, A.C. Catlin, and J. Pekny. (2014, August). Solar PV. *nanoHUB Database*. [Online]. Available: https://nanohub.org/resources/solarpv
30. A.C. Catlin et al. (2015, February). Comparative analytics of infusion pump data across multiple hospital systems. *American Journal of Health-System Pharmacy, 72(3),* pp. 317-324. doi: 10.2146/ajhp140424 2015
31. S. Witz et al. (2014, September). Using Informatics to Improve Medical Device Safety and System Thinking. *Association for the Advancement of Medican Instrumentation Horizon, 48(2),* pp. 38-43. doi: 10.2345/0899-8205-48.s2.38
32. T. Hacker et al. (2011, July). The NEEShub Cyberinfrastructure for Earthquake Engineering. *Computing in Science and Engineering*, *13(4)*, pp. 67-75.
33. A. Boyko, J. Kunze, J. Littman, L. Madden, B. Vargas (2009). NDIIPP Content Transfer Project: The BagIt File Packaging Format. [Online]. Available: https://confluence.ucop.edu/display/Curation/BagIt
34. S. Weibel, J. Kunze, C. Lagoze, and M. Wolf (1998, September). Dublin core metadata for resource discovery. RFC 2413, IETF. [Online]. Available: dl.acm.org/citation.cfm?id=rfc2413

ABOUT THE AUTHORS

Ann Christine Catlin is a senior research scientist at the Rosen Center for Advanced Computing at Purdue University. Her research interests include problem-solving environments, knowledge-based systems, architecture and technologies for web-based medical and scientific research databases, and mathematical software. Catlin has an MS in mathematics from Notre Dame University. Contact her at acc@purdue.edu.

Chandima HewaNadungodage is a senior software engineer at the Rosen Center for Advanced Computing at Purdue University. Her research interests include database management, data stream mining, big data analysis using GPGPUs, and machine learning. HewaNadungodage has a PhD in computer science from Purdue University. Contact her at chewanad@purdue.edu.

Santiago Pujol is a professor of civil engineering and the academic director for research computing at Purdue University. His research interests include seismic vulnerability of existing structures, displacement-based seismic design, and instrumentation and evaluation of existing structures. Pujol has a PhD in civil engineering from Purdue University. He is a Fellow of the American Concrete Institute and a member of the Earthquake Engineering Research Institute. Contact him at spujol@purdue.edu.

Lucas Laughery is a postdoctoral researcher in civil engineering at Purdue University. His research interests include the behavior of structural concrete under extreme demands, instrumentation and testing of structures, and research data management. Laughery has a PhD in civil engineering from Purdue University. He is a member of the American Concrete Institute and the Earthquake Engineering Research Institute. Contact him at llaugher@purdue.edu.

Chungwook Sim is an assistant professor of civil engineering at the University of Nebraska–Lincoln. His research interests include the modeling and testing of structural concrete, the response of structural members under extreme loads, and infrastructure health monitoring. Sim has a PhD in civil engineering from Purdue University. He is a member of the American Concrete Institute and the Earthquake Engineering Research Institute. Contact him at csim@unl.edu.

Aishwarya Puranam is a PhD candidate in civil engineering at Purdue University. Her research interests include the behavior of structural concrete under extreme demands and instrumentation and testing of structures. Puranam has an MS in civil engineering from Purdue University. She is a member of the American Concrete Institute and the Earthquake Engineering Research Institute. Contact her at apuranam@purdue.edu.

Andres Bejarano is a research associate at the Rosen Center for Advanced Computing and a PhD candidate in computer science at Purdue University. His research interests include computer graphics, scientific visualization, UI/UX design, and web-based platforms. Bejarano received an MS in computer science from Purdue University and an MS in systems engineering and computation from the Universidad del Norte. Contact him at abejara@purdue.edu.