



**US Army Corps
of Engineers®**
Engineer Research and
Development Center



Engineered Resilient Systems

Web-Enabled Interface for iRODS: Comparing Hydroshare and Metalnx

Kevin D. Winters, Mark A. Cowan, Glover E. George,
Megan E. Gonzalez, Brian Priest, Omar Morris,
and Jonathan Landrum

August 2020



The U.S. Army Engineer Research and Development Center (ERDC) solves the nation's toughest engineering and environmental challenges. ERDC develops innovative solutions in civil and military engineering, geospatial sciences, water resources, and environmental sciences for the Army, the Department of Defense, civilian agencies, and our nation's public good. Find out more at www.erdcenter.usace.army.mil.

To search for other technical reports published by ERDC, visit the ERDC online library at <http://acwc.sdp.sirsi.net/client/default>.

Web-Enabled Interface for iRODS: Comparing Hydroshare and Metalnx

Kevin D. Winters, Mark A. Cowan, Glover E. George, and Megan E. Gonzalez

*U.S. Army Engineer Research and Development Center (ERDC)
Information Technology Laboratory (ITL)
3909 Halls Ferry Road
Vicksburg, MS 39180-6199*

Brian Priest, Omar Morris, and Jonathan Landrum

*Geocent Jackson Mississippi Office
1000 Highland Colony Parkway
Suite 5203
Ridgeland, MS 39157*

Final Report

Approved for public release; distribution is unlimited.

Prepared for Headquarters, U.S. Army Corps of Engineers
Washington, DC 20314-1000

Under Work Unit 92L5D8

Abstract

The Integrated Rule-Oriented Data System (iRODS) software provides ample resources for managing data and collections thereof, but there are occasions where utilizing its command line interface (CLI) is impractical or not desirable. One such example is when it is required that the user authenticate using a common access card (CAC), which is more easily accomplished through a graphical user interface (GUI) than through a CLI. Furthermore, restricting the system to only offering a CLI can alienate users who would normally be averse to using a system in such a way, and there are users who are not averse to utilizing a CLI, but who would still benefit from a GUI until they are able to familiarize themselves with the iCommands provided by iRODS. Thus, it becomes imperative to either implement or utilize an existing GUI for the system.

DISCLAIMER: The contents of this report are not to be used for advertising, publication, or promotional purposes. Citation of trade names does not constitute an official endorsement or approval of the use of such commercial products. All product names and trademarks cited are the property of their respective owners. The findings of this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

DESTROY THIS REPORT WHEN NO LONGER NEEDED. DO NOT RETURN IT TO THE ORIGINATOR.

Contents

Abstract	ii
Figures and Tables	iv
Preface	v
1 Introduction	1
1.1 Background.....	1
1.2 Objectives.....	1
1.3 Approach	1
2 Benefits of a Web-Enabled GUI	2
2.1 Intuitive user interface	2
2.2 No need to compile for multiple platforms.....	2
2.3 Ability to access the system from remote devices	2
2.4 Does not require installation.....	3
2.5 Updated immediately when changes are made	3
3 Comparison of Features	4
3.1 HydroShare	4
3.2 Unique HydroShare features	5
3.2.1 Public cloud or local installation	5
3.2.2 Social interaction.....	5
3.2.3 Ability to publish data objects.....	6
3.2.4 Extendable via an API.....	7
3.3 Metalnx.....	8
3.3.1 Project details	8
3.3.2 Unique Metalnx features.....	8
3.3.3 iRODS administrator dashboard.....	8
3.4 Complete set of management features.....	9
3.4.1 Utilizes underlying iRODS credentials for authentication	10
3.4.2 User profiles and custom metadata templates	10
3.5 Analysis of comparison.....	11
3.6 Explanation of partial features.....	12
3.6.1 Group management	13
3.6.2 User management.....	13
3.6.3 Metadata templates.....	13
4 Recommendations and Conclusions	14
References	15
Acronyms and Abbreviations	16
Report Documentation Page	

Figures and Tables

Figures

Figure 1. HydroShare groups.	6
Figure 2. Example application available in HydroShare.	7
Figure 3. Metalnx dashboard.	9

Table

Table 1. Comparison of features.....	12
--------------------------------------	----

Preface

This is a deliverable product under the Engineered Resilient Systems (ERS) Program, Data Analytics Work Package, Collaborative Tradespace Analytics Work Unit 92L5D8. Dr. Owen J. Eslinger was the program manager, and Dr. Robert M. Wallace was the Technical Director of the ERS program.

The work was performed by the Computational Analysis Branch (CAB) of the Computational Science and Engineering Division (CSED), Engineer Research and Development Center (ERDC), Information Technology Laboratory (ITL), Vicksburg, MS. At the time of publication, Dr. Jeffrey L. Hensley was Chief, CAB; Dr. Jerrell R. Ballard, Jr. was the Chief, CSED; Ms. Patti S. Duett was the Deputy Director of ITL and the Director was Dr. David A. Horner.

COL Teresa A. Schlosser was the Commander of ERDC, and Dr. David W. Pittman was the Director.

1 Introduction

1.1 Background

The Integrated Rule-Oriented Data System (iRODS) software provides ample resources for managing data and collections thereof, but there are occasions where utilizing its command line interface (CLI) is impractical or not desirable. One such example is when it is required that the user authenticate using a common access card (CAC), which is more easily accomplished through a graphical user interface (GUI) than through a CLI. Furthermore, restricting the system to only offering a CLI can alienate users who would normally be averse to using a system in such a way, and there are users who are not averse to utilizing a CLI, but who would still benefit from a GUI until they are able to familiarize themselves with the iCommands provided by iRODS. Thus, it becomes imperative to either implement or utilize an existing GUI for the system.

1.2 Objectives

This report serves as a comparison between Hydroshare and Metalnx with the intention of identifying which application will serve as the GUI for iRODS. It will also examine the features of Hydroshare and Metalnx with the intention of identifying the similarities and differences. These features will then be used to determine which application should serve as the GUI for iRODS.

1.3 Approach

This document examines the features of Hydroshare and Metalnx and highlights the differences between the two applications. These differences are then considered with the iRODS software to determine which GUI to implement.

2 Benefits of a Web-Enabled GUI

The two proposed GUI projects are both web-enabled, meaning the user interacts with them via a web browser. There are numerous benefits to this method of GUI development over the traditional method of creating a GUI that is native to the operating system.

2.1 Intuitive user interface

iRODS is a powerful platform and has several ways of interacting with data, such as a representational state transfer (REST), application programming interface (API), Java API, Python API, R API, and the CLI. However, providing users with a GUI can make data searching, metadata management, and bulk uploading much easier for beginner users, due to the intuitive user interface a GUI provides. For advanced users of iRODS familiar with the different APIs, a GUI would be a welcomed addition for browsing data and metadata.

2.2 No need to compile for multiple platforms

Traditionally, a GUI would be developed for each platform on which the software would run; for example, the same GUI may have to be implemented in .NET for Windows, Swift for Macintosh, and GTK+ for Linux. There are cross-platform alternatives, such as Java Swing and Qt, however, they do not offer the native look and feel that platform-specific toolkits offer. Thus, either more effort must be spent to cover all potential platforms, or a subset of potential users must be excluded with the selection of a single platform to support. This is mitigated with a web-enabled GUI since the web browser is already written.

2.3 Ability to access the system from remote devices

Many users have multiple devices from which they would like to be able to use the system. By utilizing a web-enabled GUI, any device with network access can reach the system. CAC Authentication can be used to secure the remote web-interface if desired. This makes in-person collaboration and telecommuting easier as users are not tied to their workstations. It also makes licensing easier, as the provider of the software does not have to rely on license keys for an installation, but can simply utilize the user's authentication credentials as a basis for license enforcement.

2.4 Does not require installation

When attempting to use the system from multiple devices, a user of a native GUI would have to install the system on every device they intend to use. This requires more overhead for information technology (IT) department staff, who have to administrate each of these installations and keep the installations updated. With a web-enabled GUI, there is nothing to install on the user's workstation, thus, obviating this staff overhead altogether. This also simplifies device upgrade procedures as there is nothing that needs to be installed on the new device for it to be compatible with a web-enabled GUI.

2.5 Updated immediately when changes are made

With a native GUI, when the IT department wants to roll out the next version of the software or updates to the current version, they may have to do so in waves, depending on the size of the user base. If these updates are instead pushed to the user base via system updates, there may be a delay between the time the update is available and the time the update is installed while the IT department staff evaluate all available updates and determine if any will cause system instability, including the updates to the native GUI in question. With a web-enabled GUI, updates are initiated on the user system immediately when they are released, with no need for assistance from IT Department staff, and there is a significantly less threat that an update to a web-enabled GUI will have adverse effects on the rest of the software on the user's device versus a native application.

3 Comparison of Features

In general, both products examined have significant merits and would make good additions to the iRODS architecture. They both offer the ability to browse existing data objects and upload new ones, they both offer the ability to add, modify, and remove metadata from data objects, they both offer the ability to search the metadata, and they both offer the ability to share the data and metadata with collaborators (Henderson et al. 2017; Conway et al. 2018). However, both products also include additional features that endear them to a particular type of user.

3.1 HydroShare

HydroShare is a hydrologic information system for sharing data, models, and code. The goal of the project is to enable researchers to easily discover and access hydrologic data and models, retrieve them to their desktop, or perform analyses in a distributed computing environment that may include grid, cloud, or high-performance computing. Researchers may also publish outcomes (data, results, or models) into HydroShare using the system as a collaboration platform for sharing these data, results, and models (Consortium of Universities for the Advancement of Hydrologic Science, Inc. (CUAHSI) 2016f).

HydroShare is an open-source project, licensed under the BSD 3-clause license, and the source code is available on GitHub (Henderson et al. 2017). The project is operated by CUAHSI, and the community surrounding its development is very active (Henderson et al. 2017; CUAHSI 2016a). A fully-featured installation of HydroShare is not a single application, but is rather a technology stack, and utilizing HydroShare would require administrating much more software in addition to iRODS. The user interface is comprised of a large Python/Django application, and the remainder of the features are provided by external projects (Henderson et al. 2017)

- Solr for searching
- Redis for caching
- RabbitMQ for concurrency and serialization
- iRODS for federated file storage
- PostgreSQL for database.

Thus, the project is not so much of a “front end” for iRODS as it is an extension to it, having an almost symbiotic relationship with iRODS. Users can upload data objects and add metadata to them, but cannot control iRODS administrative features, such as adding storage resources or modifying zones. Those administrative functions would need to be performed from within iRODS directly. However, for general use, the interface provided by HydroShare is more than sufficient.

3.2 Unique HydroShare features

HydroShare has many of the same features as Metalnx, but includes a few unique ones that set it apart. These unique features are not necessarily aimed at the hydrological sciences, and can be utilized by any generic user of the system.

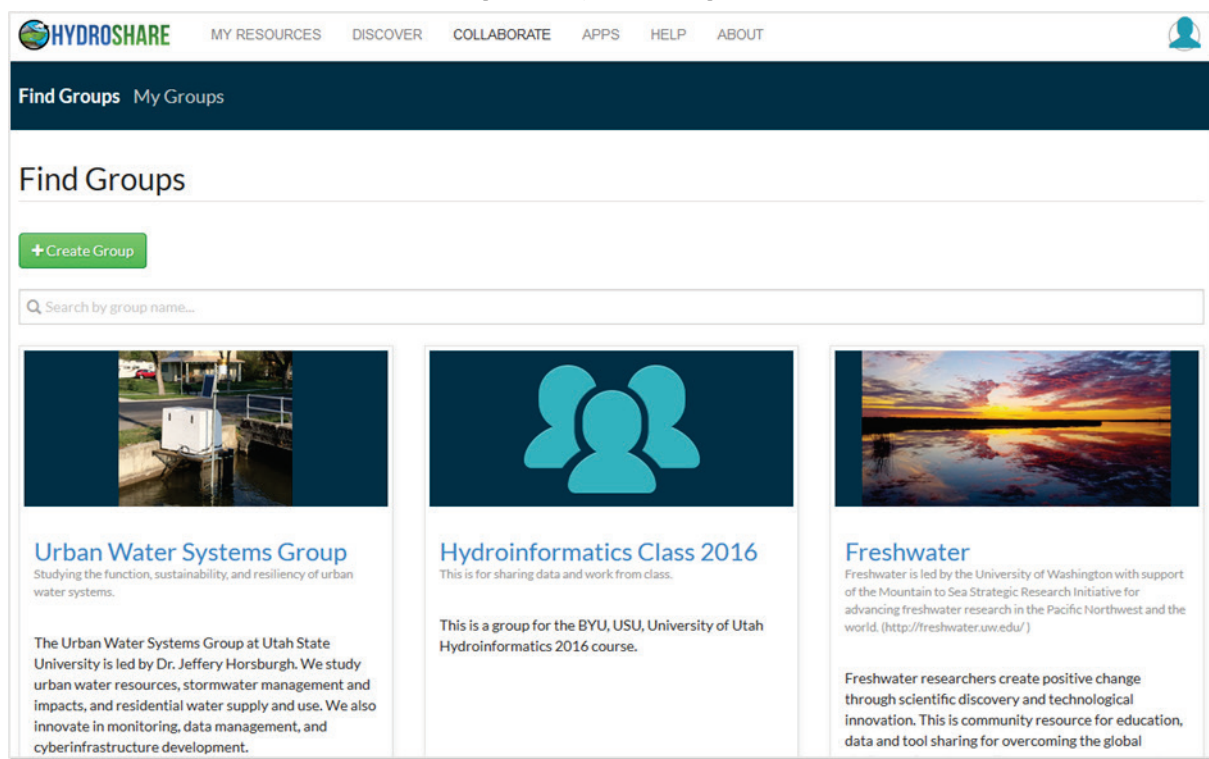
3.2.1 Public cloud or local installation

There is a public version of HydroShare available for anyone to access at the HydroShare website (CUAHSI 2019). Users are granted a small amount of storage space when they register for an account, and it is possible to link the account with an external iRODS installation so storage is not an issue. Furthermore, utilizing HydroShare in this way obviates the administration overhead associated with running a locally installed instance. Alternatively, it is also possible to set up HydroShare locally so that more fine-grained control of access can be realized.

3.2.2 Social interaction

The platform is designed with social functionality that enables the user to effectively collaborate with other HydroShare users. One way this is accomplished is by exposing to the GUI an underlying functionality in iRODS known as Groups (Bandaragoda and Leon 2018). The user can create a group for their research team and share resources within that group. Groups can be public or private, and the user has control over what is shared within the group. These groups can be browsed from the HydroShare interface and a user can request access from the group page (Figure 1).

Figure 1. HydroShare groups.



Another social feature offered is the ability to add comments and ratings to data objects. Commenting on and rating data objects is intended to add contextual value to the object; for example, adding a comment about the success or failure of a particular analysis on a particular data set so other collaborators can know how to extend the research (CUAHSI 2016f).

3.2.3 Ability to publish data objects

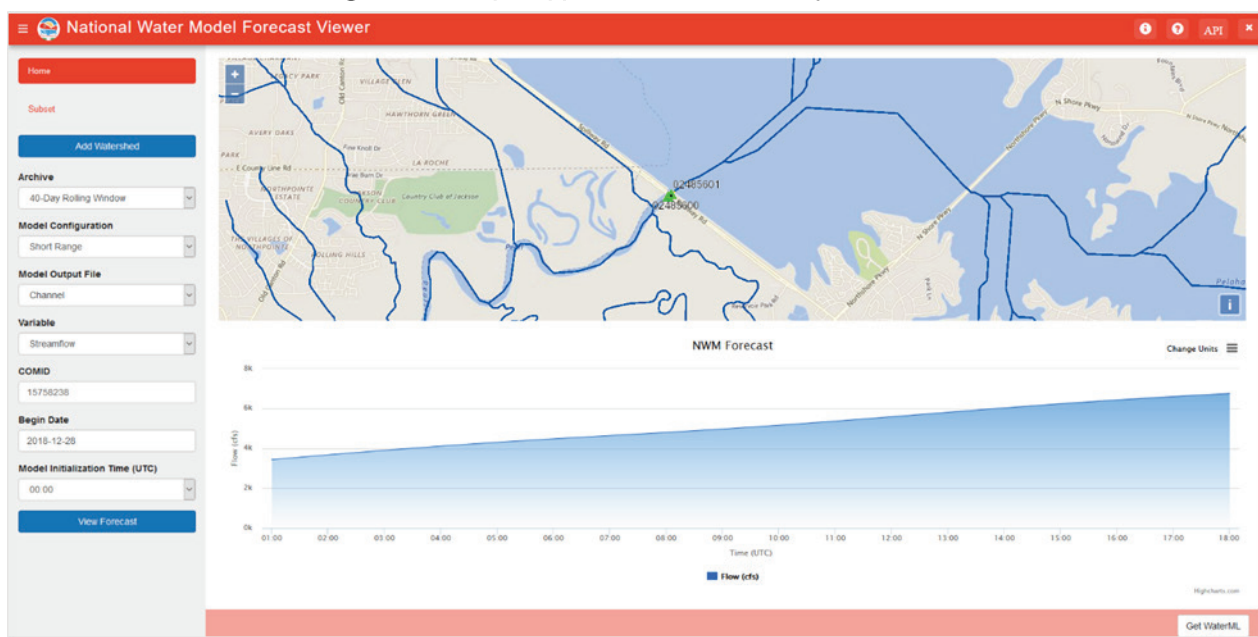
Once data objects are uploaded into HydroShare, the user can choose to publish the data object, which converts it into a permanent, immutable resource and registers a unique data object identifier (DOI) to it (CUAHSI 2016b). This feature can be compared to the act of uploading an electronic preprint to arXiv where a finished product is made available to other users in perpetuity. Indeed, a notice is given to the user when converting a data object to a published work that this action should be thought of with the same care as publishing a paper, and retraction of the published object can be laborious (CUAHSI 2016b). The DOI assigned to the object can be referenced from a journal article as a way of providing not just the synopsis of the research in the form of a paper, but also the complete data set, models, and software used to get the results, so that replication and extension can be more easily accomplished. It should be noted that

publishing a data object sets its access rights to be the same as those that are publicly accessible. Community data sets available to the public include those from the National Aeronautics and Space Administration, the Environmental Protection Agency, the National Oceanic and Atmospheric Administration, and others (CUAHSI 2016d).

3.2.4 Extendable via an API

HydroShare can be extended by way of a REST API. This API enables users to create their own applications and extend existing ones to visualize, analyze, and work with data objects from within HydroShare. One example is the National Water Model Forecast Viewer, which utilizes publicly available data to predict flow rates of streams (CUAHSI 2016e).

Figure 2. Example application available in HydroShare.



Thus, this API enables the user to create a Platform as a Service (PaaS) application to offer to other researchers. Metadata for an application instance describes the application, sets the permissions (who is given access to the application), and defines the types of data objects upon which the application can act. These applications run on their own web server, either on the same physical machine that is powering HydroShare, or on a separate, network-accessible one. Applications can be used to operate on original data and models or be used to access community data sets

(CUAHSI 2016d). Metalnx does not offer an API for direct interaction in this way, as iRODS offers similar APIs natively.

3.3 Metalnx

Metalnx is a web application designed to work alongside iRODS. It provides a GUI that can help simplify most administration, data collection, and metadata management tasks, removing the need to memorize the long list of iCommands. Metalnx was originally developed by engineers at EMC, and after Dell acquired the company, the project was open sourced and turned over to the iRODS Consortium.

3.3.1 Project details

Metalnx is an open-source project, licensed under the BSD 3-clause license, and the source code is available on GitHub (Conway et al. 2018). The project is governed by the iRODS Consortium and is intended to “provide an iRODS-centric, white-label, easily installable, usable front-end to an iRODS Zone” (Russell 2018). The Metalnx community is very active, with frequent updates to the codebase, and this development is produced by many of the same programmers writing iRODS itself.

Metalnx is written in Java as a Tomcat application, with development centered on the concept of a Maven build lifecycle. After a successful build of the software, a web application resource (WAR) file is produced, which is easily deployed to the web server. As the project is run by the iRODS Consortium, very few dependencies are needed beyond what it required by iRODS itself, most notably Java and Tomcat (Conway et al. 2018).

3.3.2 Unique Metalnx features

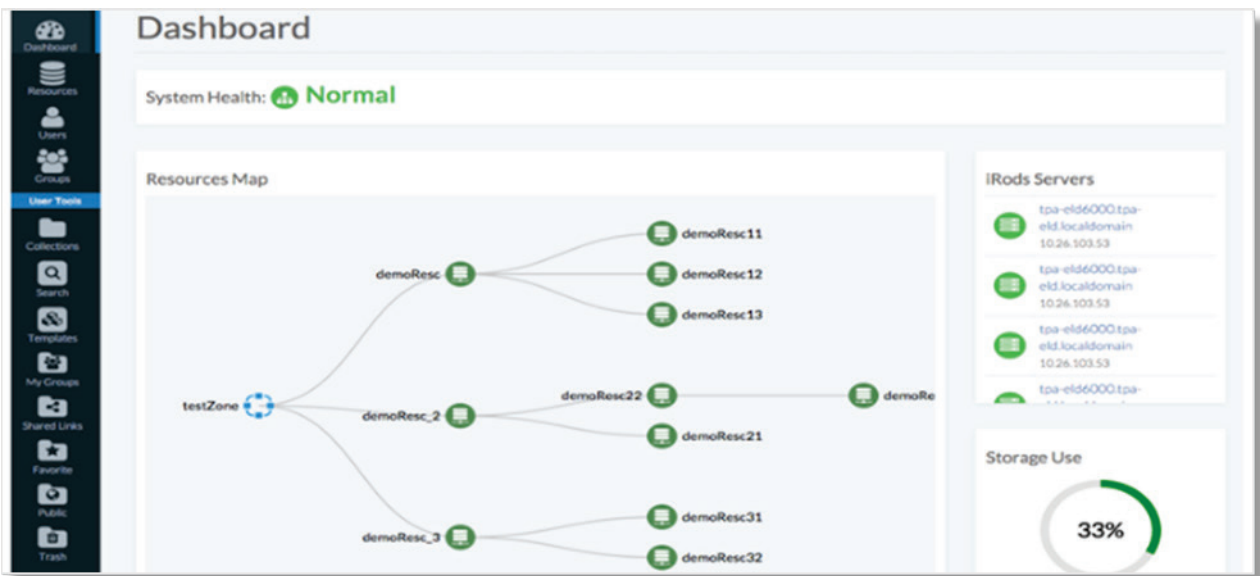
Because Metalnx is a project of the iRODS Consortium, great effort has been placed on ensuring its compatibility with iRODS and on its feature breadth (Russell 2018). As such, there are features offered by Metalnx that are not available in any other extant iRODS GUI project.

3.3.3 iRODS administrator dashboard

The Metalnx interface includes a dashboard page for administrators to see the resources, servers, and storage use of the installation at a glance (Conway et al. 2018). On this page, administrators can see the state of the grid as a “System Health” score. They can also see a map of the resources

in the installation and how the defined zones interact with those resources. There is a list of the servers powering the iRODS installation, as well as a pie chart showing the total usage of the installed disk storage capacity. Administrators can drill down to individual storage devices and servers to get more details about the health and capacity of the entire system. Another unique feature of the dashboard is the ability to see if all iRODS microservices have been successfully deployed to all iRODS servers. Non-administrators do not have access to the dashboard and will see the collections page when they log in.

Figure 3. Metalnx dashboard.



3.4 Complete set of management features

For both administrators and regular users, full access to management iCommands available to the respective user type are accessible from the Metalnx GUI (Conway et al. 2018). For regular users, this includes collection management (add, edit, upload, download, copy, move, and delete data objects), and metadata management (add, edit, and remove). This set of features is extended for administrators; in addition to the previously stated abilities, administrators also have user management (add, edit, and remove), group management (add, edit, and remove), and resource management (add, remove, and view information about storage resources in the iRODS grid). This level of management is unparalleled among current iRODS GUI platforms.

3.4.1 Utilizes underlying iRODS credentials for authentication

Since Metalnx is tightly coupled with iRODS, a user only needs valid iRODS credentials in order to log into and utilize Metalnx (Guerra 2017). The user's permissions are identical to those of the underlying iRODS system, and any changes to permissions and group membership that happens in Metalnx are made in iRODS as well. The opposite is also true, where changes made in iRODS are recognized immediately in Metalnx. Furthermore, if an administrator creates, modifies, or removes a user from within Metalnx, that change is made in iRODS, just as if the administrator had made the change from the CLI (Conway et al. 2018). It is also possible to configure Metalnx and iRODS to utilize the lightweight directory access protocol (LDAP) for authentication, so long as the authentication type in iRODS is set to pluggable authentication modules (PAM) Guerra 2017.

3.4.2 User profiles and custom metadata templates

Having to repetitively apply the same information to multiple items can be tiresome. Metalnx is working to solve this problem in two areas. For regular users, metadata templates can be created, edited, and removed (Conway et al. 2018). These templates define multiple metadata tags with default values, which can then be applied across one or many data objects from within the collection management page.

This feature is different than the feature in HydroShare that is also called metadata templates. In HydroShare, the notion of a metadata template would more appropriately be named metadata requirements. These requirements are based on the data object type, whereby the user is prompted for metadata values for files uploaded of the given type (CUAHSI 2016c). The ability to prompt for required metadata is a feature that is fully implemented in both HydroShare and Metalnx, and is in fact a feature of the underlying iRODS software itself. In contrast, the feature in Metalnx, known as metadata templates, is the ability to create a custom set of metadata attribute-value-units *triples* (AVUs) with default entries for the attribute, value, and units, which can be applied to data objects that have already been uploaded. For example, a template can be created to add new metadata, such as the location from which the data was gathered and the tools used for gathering it, to a set of data objects. This feature is currently a work in progress and will be available after resolution of a feature update within the underlying iRODS software.

For administrators, in addition to metadata templates, user profiles can be created, edited, and removed (Conway et al. 2018). These profiles contain pre-defined groups and permissions settings that can then be applied to any iRODS user as needed. For example, onboarding new employees or creating new research groups can be accomplished in a single step after the user accounts are created. Also, permission changes that are system-wide can be made to all users simultaneously with this method.

3.5 Analysis of comparison

HydroShare provides a sufficient GUI experience for non-administrative iRODS use, and considering that most of the utilization of the system will fall within this realm, Metalnx extends its GUI to a more robust and complete set of abilities for controlling the underlying iRODS installation.

In Table 1, the major features of each product are compared. A feature that is labeled “Full” indicates the respective platform has fully implemented the stated feature. A feature that is labeled “None” indicates the respective platform has not implemented the stated feature at all. Features that are labeled “Partial” will be discussed further in the following sections.

As Table 1 shows, the major features for day-to-day utilization of the iRODS system by a typical user via the respective GUI are fully implemented in both platforms. Determining which platform to use will be based on the critical points of the extraneous features of one platform over another and the partially implemented features of one platform versus another.

Table 1. Comparison of features.

Feature	HydroShare	Metalnx
Add metadata to data objects	Full	Full
Automatic metadata extraction during upload (depending on data type)	Full	Full
Collection management (add, edit, upload, download, copy, move, and delete data objects)	Full	Full
Drag-and-drop interface	Full	Full
Manage who has access to the content shared	Full	Full
Metadata management (add, edit, and remove metadata tags)	Full	Full
Publish data, models, and code to a public-facing website to be easily found by other researchers with access rights to the site	Full	Full
Search by data objects by their metadata	Full	Full
Share data, models, and code with collaborators	Full	Full
Group management (add, edit, and remove iRODS groups)	Partial	Full
User management (add, edit, and remove iRODS users)	Partial	Full
Metadata templates (apply multiple metadata tags with default values)	None	Partial
Dashboard that displays information on system health	None	Full
Resource management (add, remove, and view information about storage resources in the iRODS grid)	None	Full
User profile template management (pre-defined groups, permissions, etc. that can be applied to a user)	None	Full
Utilize underlying iRODS credentials for authentication	None	Full
Ability to create a citable Digital Object Identifier (DOI) for data objects	Full	None
Group discovery (browse, request membership, invite others)	Full	None
Social platform for commenting on and rating uploaded data objects	Full	None
Web services API (for programming additional features/apps)	Full	None

3.6 Explanation of partial features

The three rows in Table 1 that contain references to “Partial” implementations require further explanation. These three features are either completely implemented within the underlying iRODS software or will be in an upcoming release.

3.6.1 Group management

Metalnx provides complete group management abilities within its interface. That is, it is possible to add, edit, and remove a group as either a system administrator or as a group administrator (Conway et al. 2018). However, while HydroShare does not provide complete group management abilities, it is possible to add and remove a group, and it is possible to invite a user to a group (Henderson et al. 2017). However, it is not possible for a group administrator to add a user to the group they administrate.

3.6.2 User management

Within HydroShare, it is possible to create a new user from the “Register” page, for the user or an administrator to modify that user, and for the user or an administrator to remove that user (Henderson et al. 2017). However, it is not possible for a group administrator to create new users. Instead, group administrators must invite existing users to join the group. The ability of a group administrator to create users is a feature of the underlying iRODS software, but which has not been extended to the HydroShare GUI. Metalnx has this feature fully implemented.

3.6.3 Metadata templates

HydroShare allows for effective collaboration, but lacks a way to uniformly ensure that metadata and data is shared within the community. Metalnx is currently working on a feature that supports a similar functionality – metadata templates (Worth et al. 2018). The interface is fully implemented; however, this feature is marked “Partial” because there is an underlying functionality within iRODS that is not yet complete. Once the underlying feature is complete, this functionality will work as advertised within Metalnx.

4 Recommendations and Conclusions

This report has examined the reasoning behind adding a GUI to iRODS, the benefits of a web-enabled interface for iRODS, and why HydroShare and Metalnx were chosen to be examined. It was shown that both projects have many of the same features, but a few important features were identified in each project that do not exist in the other. With HydroShare, its uniqueness lies mostly in its social features and its architecture design, which allows for easy application integration with the data provided through the system. With Metalnx, its uniqueness is in its management features and its close coupling with iRODS. Taking consideration for the features within HydroShare and Metalnx and the impact of these features on projects utilizing the iRODS installation, it is recommended that Metalnx be used as the iRODS GUI. If additional HydroShare features are found necessary, it is recommended to create a separate, public HydroShare zone within iRODS and point to that zone from a group in the publicly accessible HydroShare cloud. This would enable public sharing of unclassified data for the benefit of the greater hydraulic community while retaining the privacy of non-shared data.

References

- Bandaragoda, C., and M. Leon. 2018. Using Groups to Collaborate. (Accessed 27 December 2018). <https://github.com/hydroshare/hydroshare/wiki/Using-Groups-to-Collaborate>.
- The Consortium of Universities for the Advancement of Hydrologic Science, Inc. (CUAHSI) 2016a. About HydroShare. (Accessed 27 December 2018) <https://help.hydroshare.org/participation-and-collaboration/>.
- Conway, M., S. Worth and M. Y. Ng. 2018. Metalnx Web. iRODS Consortium, 25 October 2018. (Accessed 27 December 2018). <https://github.com/irods-contrib/metalnx-web>.
- . 2016b. Formally Publishing Data. (Accessed 3 January 2019). <https://help.hydroshare.org/introduction-to-hydroshare/frequently-asked-questions/>.
- . 2016c. Frequently Asked Question. (Accessed 27 December 2018). <https://help.hydroshare.org/creating-and-managing-resources/formally-publishing-data/>.
- . 2016d. Getting Started With HydroShare. (Accessed 27 December 2018). <https://help.hydroshare.org/introduction-to-hydroshare/getting-started/>.
- . 2016e. National Water Model Forecast Viewer App. (Accessed 28 December 2018). <https://help.hydroshare.org/apps/nwm-forecast-viewer/>.
- . 2016f. Participation and Collaboration. (Accessed 26 December 2018). <https://help.hydroshare.org/about-hydroshare/>.
- . 2019. HydroShare: Find, Analyze and Share Water Data. (Accessed 2 January 2019). <https://www.hydroshare.org/>.
- Henderson, M. R., M. J. Stealey, P. Dash and D. Tarboton. HydroShare. 2017. The Consortium of Universities for the Advancement of Hydrologic Science, Inc. (CUAHSI). (Accessed 27 December 2018). <https://github.com/hydroshare/hydroshare>.
- Russell, T. 2018. iRODS Development Update: January 2018. iRODS Consortium, 31 January 2018. (Accessed 28 December 2018). Guerra, A. J. 2017. LDAP: CUAHSI. (Accessed 3 January 2019). <https://github.com/irods-contrib/metalnx-web/wiki/LDAP>.
- Worth, S., T. Russell, and M. Conway. 2018. Metalnx 2.0 - The Future of Metalnx," in iRODS User Group Meeting, Durham, 2018.

Acronyms and Abbreviations

Acronym	Meaning
ACL	Access Control Lists
API	Application Programming Interface
AVU	Attribute-Value-Units
CAC	Common Access Card
CLI	Command Line Interface
CUAHSI	Consortium of Universities for the Advancement of Hydrologic Science, Inc.
DoD	Department of Defense
DOI	Data Object Identifier
ERDC	Engineer Research and Development Center
ERS	Engineered Resilient Systems
GUI	Graphic User Interface
IT	Information Technology
iRODS	Integrated Rule-Oriented Data System
LDAP	Lightweight Directory Access Protocol
PAM	Pluggable Authentication Modules
REST	Representational State Transfer
USACE	U.S. Army Corps of Engineers
WAR	Web Application Resource

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing this collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. **PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.**

1. REPORT DATE (DD-MM-YYYY) August 2020		2. REPORT TYPE Final report		3. DATES COVERED (From - To)	
4. TITLE AND SUBTITLE Web-Enabled Interface for iRODS: Comparing Hydroshare and Metalnx				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S) Kevin D. Winters, Mark A. Cowan, Glover E. George, Megan E. Gonzalez, Brian Priest, Omar Morris, and Jonathan Landrum				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER 92L5D8	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) U.S. Army Engineer Research and Development Center Information Technology Laboratory 3909 Halls Ferry Road, Vicksburg, MS 39180-6199				8. PERFORMING ORGANIZATION REPORT NUMBER ERDC/ITL SR-20-9	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) Headquarters, U.S. Army Corps of Engineers Washington, DC 20314-1000				10. SPONSOR/MONITOR'S ACRONYM(S)	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution unlimited.					
13. SUPPLEMENTARY NOTES					
14. ABSTRACT The Integrated Rule-Oriented Data System (iRODS) software provides ample resources for managing data and collections thereof, but there are occasions where utilizing its command line interface (CLI) is impractical or not desirable. One such example is when it is required that the user authenticate using a common access card (CAC), which is more easily accomplished through a graphical user interface (GUI) than through a CLI. Furthermore, restricting the system to only offering a CLI can alienate users who would normally be averse to using a system in such a way, and there are users who are not averse to utilizing a CLI, but who would still benefit from a GUI until they are able to familiarize themselves with the iCommands provided by iRODS. Thus, it becomes imperative to either implement or utilize an existing GUI for the system.					
15. SUBJECT TERMS iRODS (Computer system) High performance computing Computer architecture Computer science Web-based user interfaces					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			19b. TELEPHONE NUMBER (include area code)
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	SAR	24	