OSN User Guide

The Open Storage Network (OSN) is a distributed data sharing and transfer service intended to facilitate exchanges of active scientific data sets between research organizations, communities and projects, providing easy access and high bandwidth delivery of large data sets to researchers.

The OSN serves two principal purposes: (1) enable the smooth flow of large data sets between resources such as instruments, campus data centers, national supercomputing centers, and cloud providers; and (2) facilitate access to long tail data sets by the scientific community. Examples of data currently available on the OSN include synthetic data from ocean models; the widely used Extracted Features Set from the Hathi Trust Digital Library; open access earth sciences data from Pangeo; and Geophysical Data from BCO-DMO. These data sets are being used by researchers to machine learning models, validate simulations, and perform statistical analysis of live data.

System Overview

OSN data is housed in storage *pods* interconnected by national, high-performance networks creating well-connected, cloud-like storage that is easily accessible at high data transfer rates comparable to or exceeding the public cloud storage providers, where users can temporarily park data, for retrieval by a collaborator or create a repository of active research data .

This user guide is designed for the following categories of OSN user:

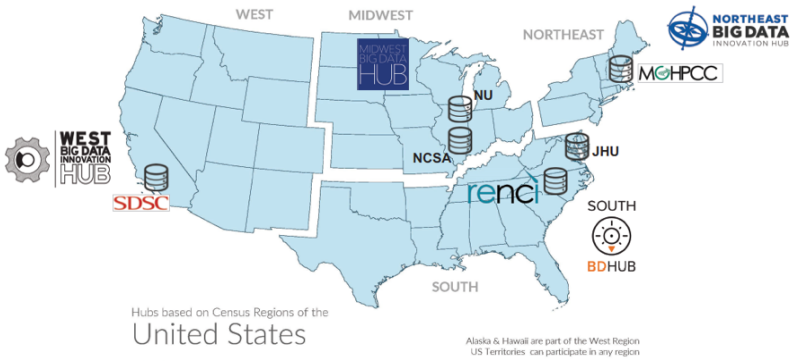
- End Users who wish to view metadata and retrieve data.
- Data Curators who maintain data sets
- Data Managers who grant access to data sets for Curators and End Users

System Configuration

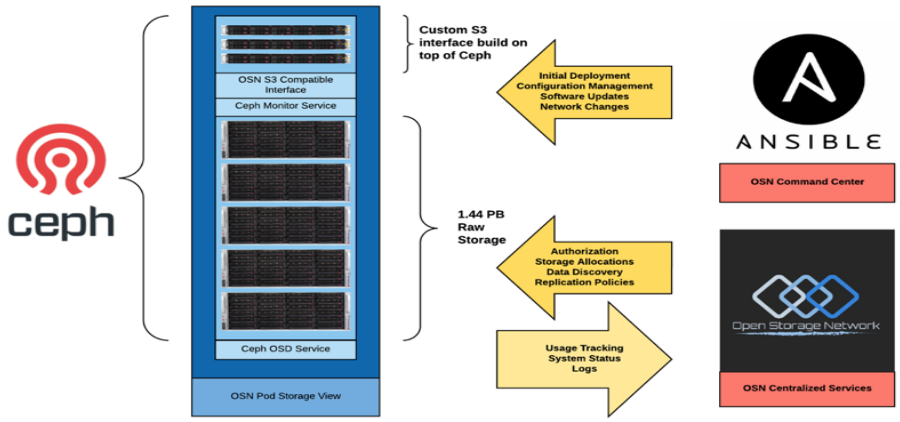
Key characteristics of OSN storage are:

- Ability to access data from anywhere via a RESTful interface that follows S3 conventions.
- Federated identity management, allowing access to protected information with existing identity via InCommon or commercial services.
- High speed access and transfer via national research and education networks
- Security and data integrity

OSN storage pods are located in science DMZs at Big Data Hub sites, interconnected by national, high-performance networks. 5 petabytes of storage are currently available for allocation.



OSN Pod Deployment at six sites as of January,2021



OSN Storage Pod

File Systems

OSN Storage is disk based and primarily intended to house active data sets. OSN storage is allocated from the pod(s) closest to the requestor with capacity to fulfill the request. Allocations of a minimum 10 terabytes and a maximum of 300 terabytes can be

requested through the XRAS process. If your project needs more than 300 terabytes, please contact the OSN team directly to discuss before you submit your request.

The OSN supports two types of data sets:

1) Open Access Data Sets that are readable by anyone and writable by Curators and Data Managers.
2) Protected Access Data Sets that are readable by invitation from a data manager and writable by Curators and Data Managers.

Every data set is a collection of objects that are individually and uniquely accessible from anywhere. For Open Access data sets, an S3 RESTful interface allows users to manipulate storage objects simply by issuing commands in the form of Uniform Resource Identifiers. For example issuing the following URI via your web browser will download a copy of this document: https://www.openstoragenetwork.org/wp-content/uploads/2021/04/OSN-UserGuide.pdf

For Protected Access Data Sets, the user first obtains an access key which is then embedded into the access command. Examples of each are provided below.

Coming soon! Consistent with FAIR principles, every OSN data set will have a landing page that makes it easy to "visit" a data set from a browser, search engine, or data catalog. The landing page contains metadata that describes the data set, along with the links to preconfigured, downloadable tools for accessing the data.

An active research data set can remain in OSN storage up to five years and usage must comply with the OSN Acceptable Use Policy.

Allocations

Storage on the OSN is allocated in standalone buckets independent of HPC allocations. There is a one-to-one mapping between buckets and allocations. This User Guide uses "Allocation" when referring to outward-facing operations such as Allocation requests, and "Bucket" when referring to inward-facing operations such as Bucket creation.

OSN storage is allocated from the resources at the location(s) closest to the requestor with capacity to fulfill the request. Allocations of a minimum 10 terabytes and max of 50 terabytes supporting up to 1.6 million files can be requested through the XRAS process. Larger allocations can be accommodated with additional review. If your project needs more than more than 50 terabytes or more than 1.6 million files, please contact the OSN team directly to discuss before you submit your request.

An active research dataset can remain in OSN storage up to five years.

Accessing Datasets

OSN supports a RESTful API that is compatible with the basic data access model of the Amazon S3 API. Any software that complies with that API can access data stored on the OSN.

There are three common methods for connecting to and using OSN resources:

- OSN portal built-in web tools,
- Third party desktop applications
- Third party data management server applications.

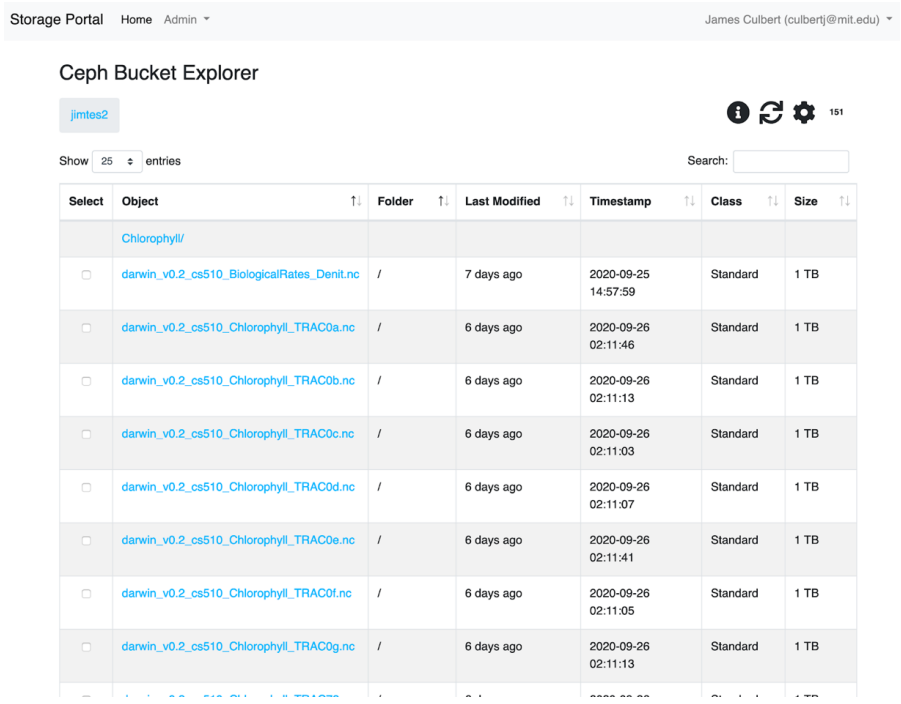
OSN Portal Built-in web tools

The OSN portal (portal.osn.xsede.org) supports a simple UI that allows end users to browse allocations and to upload and download objects via the browser. This mode of access is most appropriate for browsing a dataset and uploading/downloading smaller files (typically <100G).

To use the built-in browser, a user logs onto the OSN portal and clicks on one of the allocations that they have been granted access to. This brings the user to a searchable/sortable table listing of the allocation and its subdirectories. Clicking on any of the objects shown initiates a download of the object to the local disk.

To upload a file, the user locates the file on their local filesystem and drags the file to the browser window. This initiates an upload to the bucket location that the user is currently browsing.

OSN Basic Bucket Browser

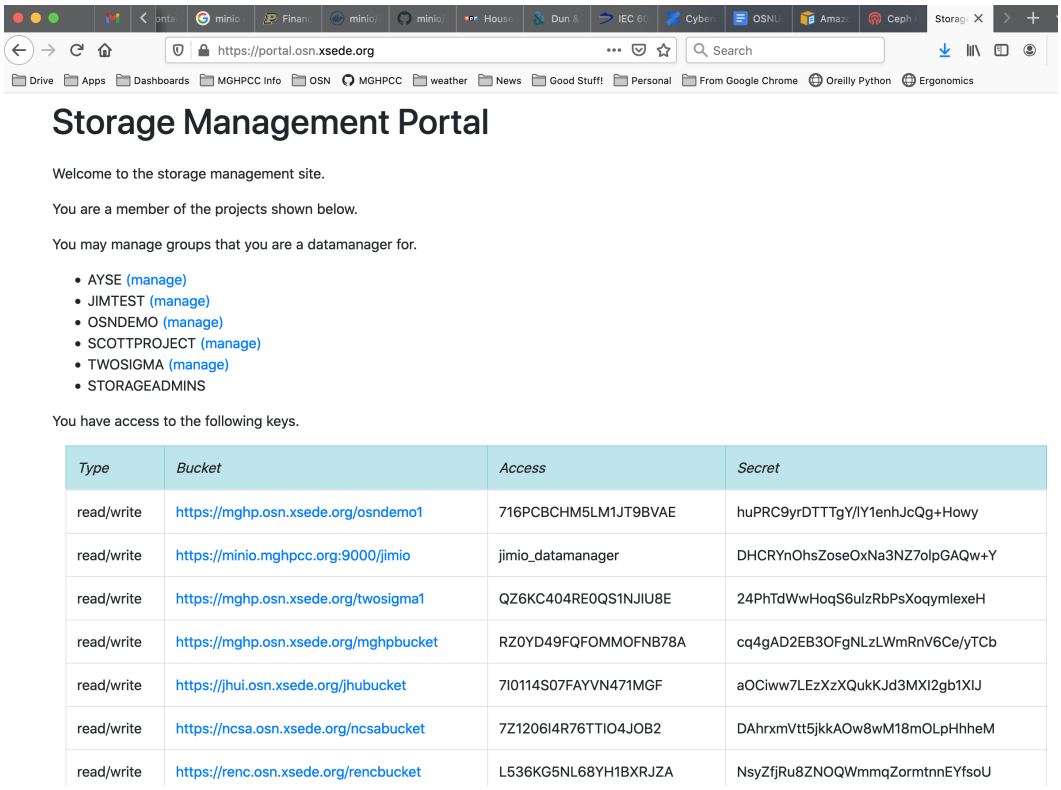


OSN Basic Bucket Explorer

Third Party Desktop Applications

There are numerous commercial and open source software tools for moving files to and from S3 buckets. These tools provide more sophisticated capabilities than the built-in browser tool including transfer management, multi-upload management and provide configuration options that can help optimize data transfer for a given computer/network environment.

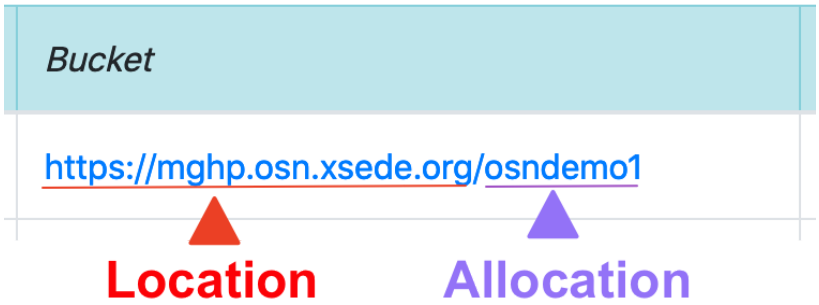
To use these tools, you will need to retrieve a pair of keys that are used to access the buckets stored on OSN. To retrieve these keys, you can contact your data manager and she will either give you keys or create an account for you on the OSN portal where you can retrieve these keys. If your data manager creates a portal account for you and gives you access to the keys you can visit https://portal.osn.xsede.org to retrieve them; the allocations you have access to and their associated keys will be listed on your home page.



Type	*Bucket*	*Access*	*Secret*
read/write	https://mghp.osn.xsede.org/osndemo1	716PCBCHM5LM1JT9BVAE	huPRC9yrDTTTgY/lY1enhJcQg+Howy
read/write	https://minio.mghpcc.org:9000/jimio	jimio_datamanager	DHCRYnOhsZoseOxNa3NZ7olpGAQw+Y
read/write	https://mghp.osn.xsede.org/twosigma1	QZ6KC404RE0QS1NJIU8E	24PhTdWwHoqS6uIzRbPsXoqymlexeH
read/write	https://mghp.osn.xsede.org/mghpbucket	RZ0YD49FQFOMMOFNB78A	cq4gAD2EB3OFgNLzLWmRnV6Ce/yTCb
read/write	https://jhui.osn.xsede.org/jhubucket	7I0114S07FAYVN471MGF	aOCiww7LEzXzXQukKJd3MXI2gb1XIJ
read/write	https://ncsa.osn.xsede.org/ncsabucket	7Z1206I4R76TTIO4JOB2	DAhrxmVtt5jkkAOw8wM18mOLpHhheM
read/write	https://renc.osn.xsede.org/rencbucket	L536KG5NL68YH1BXRJZA	NsyZfjRu8ZNOQWmmqZormtnnEYfsoU

OSN Portal User Home page

Note that the “Bucket” information displayed in the portal has two components (this will be important when you configure third party tools). The bucket information contains the OSN site/pod location and the specific allocation on that pod.



Cyberduck

Cyberduck is a popular file transfer tool that supports the S3 api. The following describes how to configure Cyberduck to connect to an OSN resource.

Cyberduck is a "cloud storage browser" for Mac and Windows that supports multiple storage providers/protocols. Cyberduck is known to work with:

- FTP
- SFTP
- WebDAV
- Amazon S3
- Ceph S3
- OpenStack Swift
- Backblaze B2
- Microsoft Azure & OneDrive
- Google Drive
- Dropbox

The software may be downloaded at:

https://cyberduck.io/download/

Using Cyberduck with OSN is straightforward.

1. Visit the osn portal and retrieve your allocation keys or retrieve them from the data manager for your project.
2. Open Cyberduck and select the bookmarks icon (see image 1)
3. Click the add icon at the bottom left of the screen to create the bookmark (image 1)
4. Edit the new bookmark to point at the desired OSN pod using the allocation key pair retrieved in step #1

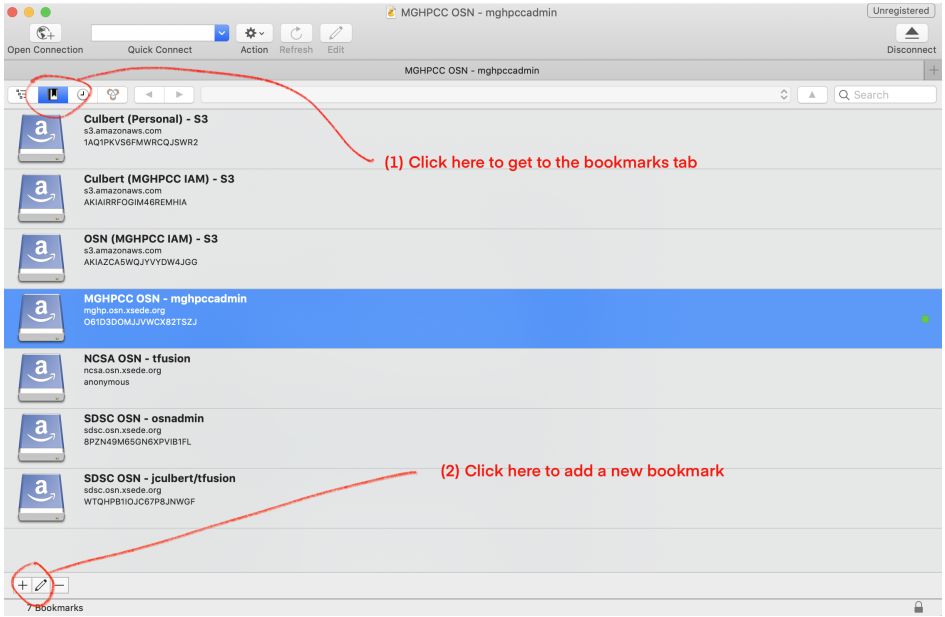


Image 1: Selecting the bookmarks page and adding new bookmark

When specifying the server, use the hostname portion of the location (i.e. if the location is https://mghp.osn.xsede.org the hostname is mghp.osn.xsede.org).

When specifying “Port”, use 443 if the location starts with “https://”; use 80 if the location starts with “http”//”.

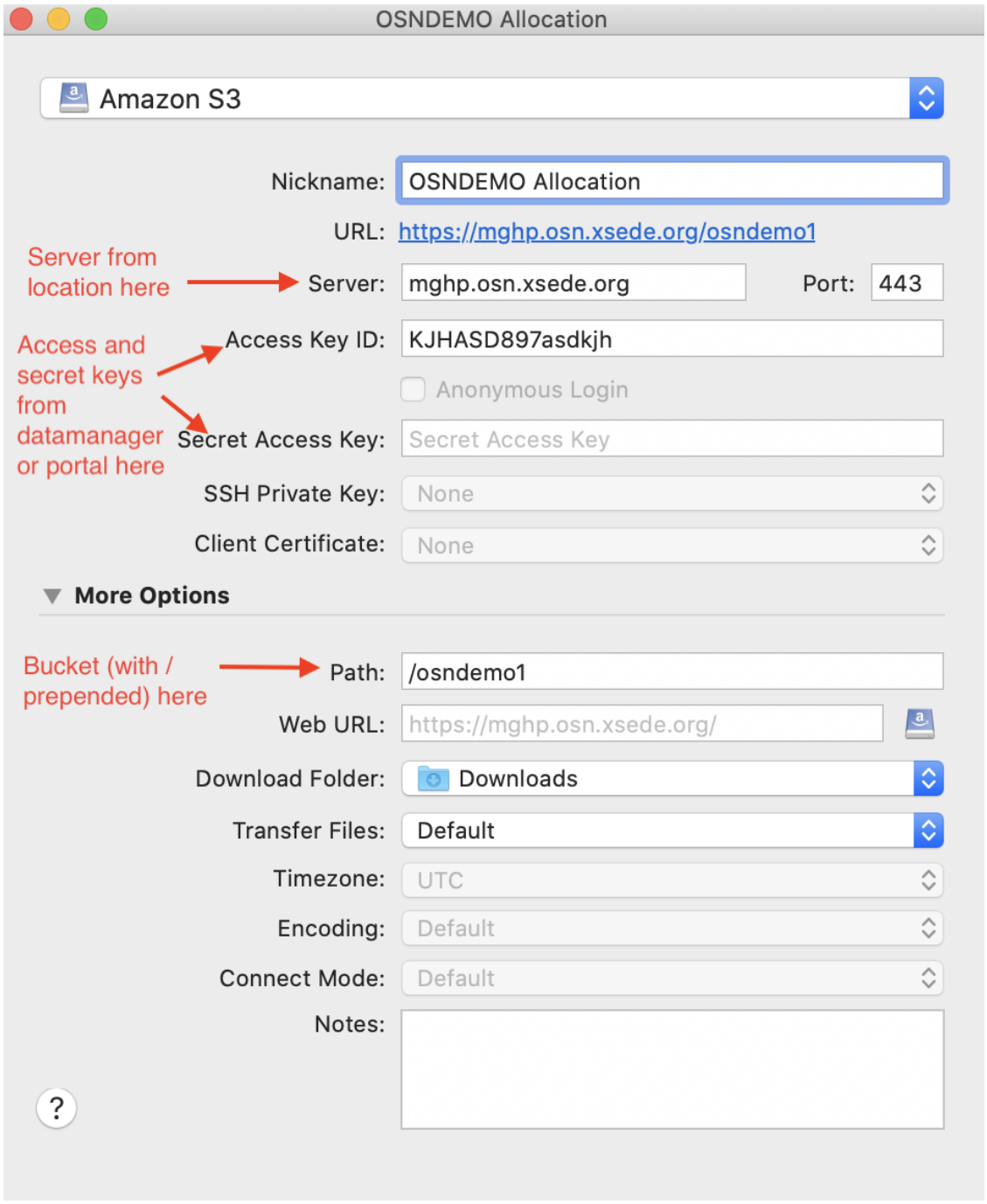


Image 2: Adding OSN pod and user information to bookmark

Anonymous Access Data Sets

Some datasets provide anonymous read access; if you are accessing buckets anonymously, type "anonymous" into the Access ID portion and Cyberduck will then select the grayed out anonymous access box in the window.

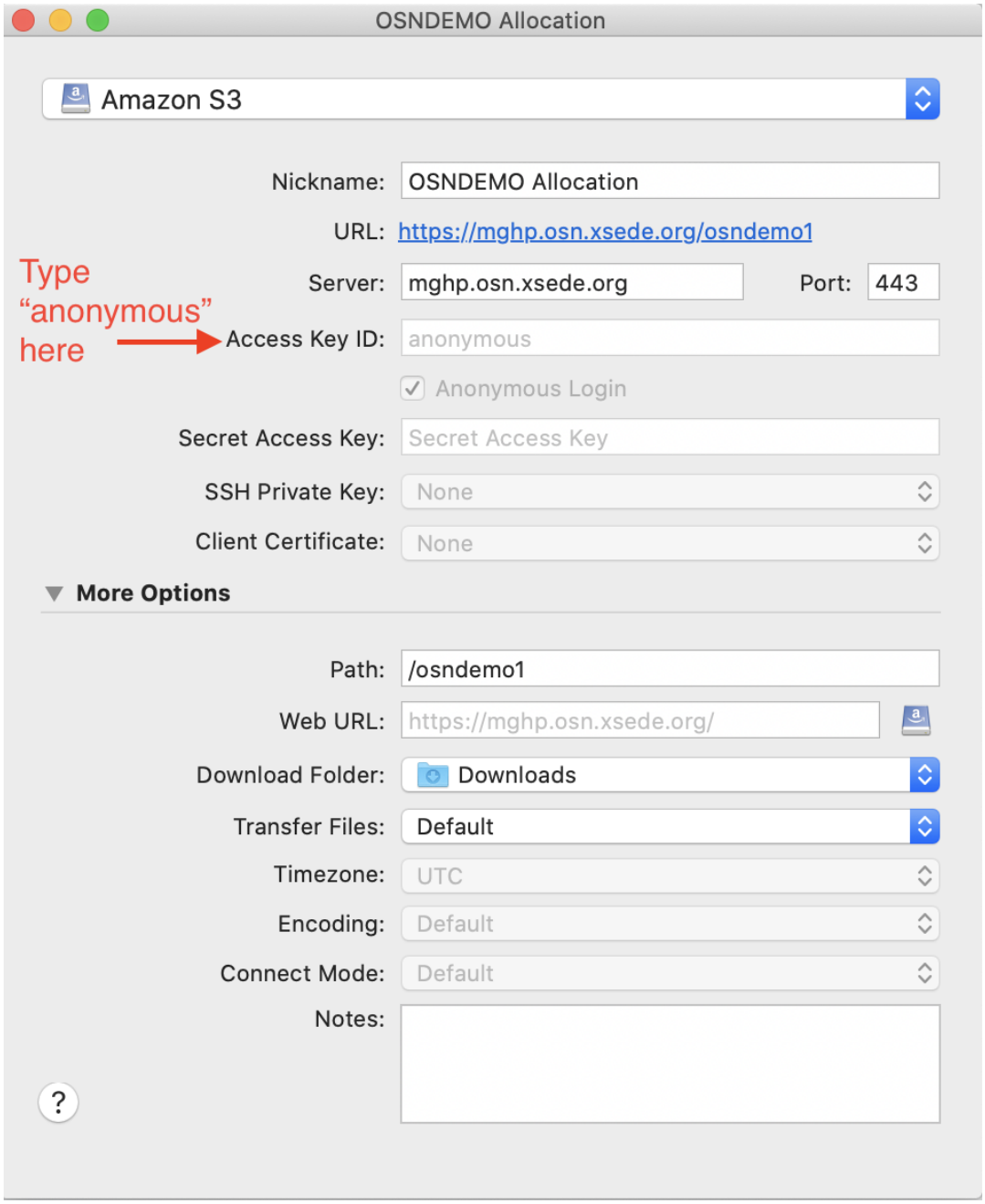


Image 2.1: Using anonymous access as your user.

Exit the window for the bookmark to save.

Browsing, Uploading and Downloading

Once a bookmark is created, you can use it to access data by double-clicking the bookmark. This logs your user in and lists the contents of the dataset.

Note: If your buckets have large object counts, you will need to increase the Timeout settings for connections.

Go to Preference>Connection and change the box next to *Timeout for opening connections (seconds)* and change the setting to 90 seconds.

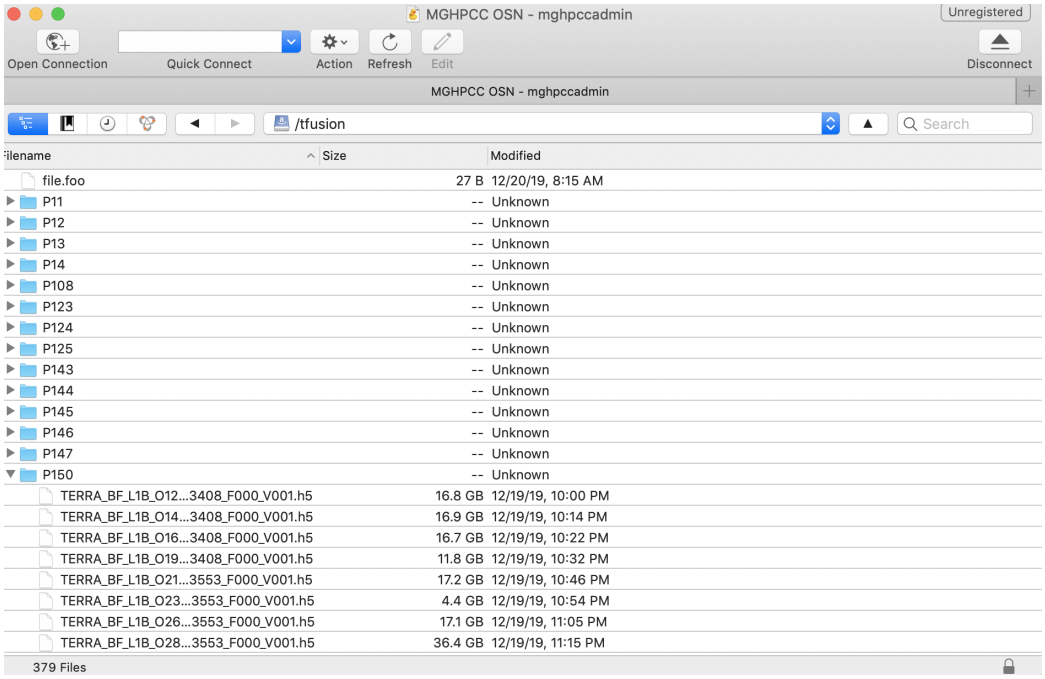


Image 4: Directory listing within bucket

Cyberduck client is a full-fledged transfer client so desktop up/downloads can be easily performed for data sets.

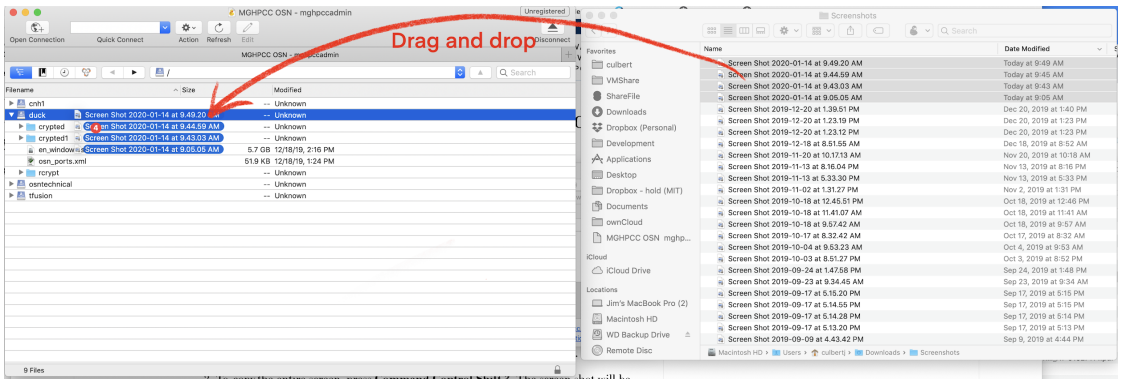


Image 7: Drag-and-drop file copy from desktop to OSN bucket

The tool supports multiple upload/download streams, chunking, pausing and restarting.

Rclone

Rclone is an open source command line utility that functions similarly to rsync and is able to communicate with numerous cloud-based storage providers. The application and documentation may be found at https://rclone.org. Download and install the application per the instructions at the rclone website.

Rclone Configuration

The most straightforward way to configure Rclone for OSN is to edit the rclone configuration file. This file may be found by typing the command “rclone config file”. The command will return the path to the rclone config file. Open this file with a text editor and add the following stanza to the end of the file:

```
[<alias here>]
type = s3
provider = Ceph
access_key_id = <access key here>
secret_access_key =<secret key here>
endpoint = <location here>
```

Where:

- <alias here> should be replaced with a nickname of your choice for the allocation
- <access key here> should be replaced with the access key from the data manager or from the portal
- <secret key here> should be replaced with the secret key from the data manager of the portal
- <location here> should be replaced with the location information provided by the data manager or portal

An example of a configuration stanza might look like:

```
[ocean-data]
type = s3
provider = Ceph
access_key_id = ASasd8KJHDAKH**&asd
secret_access_key =asd(*&Adskj*(*(&868778
endpoint = https://mghp.osn.xsede.org
```

Rclone commands are of the form:

```
rclone command alias:/bucket
```

So, using the example config file entry described above and assuming a bucket named “phytoplankton” one would list the content of the bucket using the following command:

```
rclone ls ocean-data:/phytoplankton
```

You could copy a local file to the bucket with the command

```
rclone cp my-local-file.dat ocean-data:/phytoplankton
```

Rclone offers a wide range of commands for performing typical unix file operations (ls, cp, rm, rsync, etc.) Details on these commands can be found here.

Third Party Data Management Applications

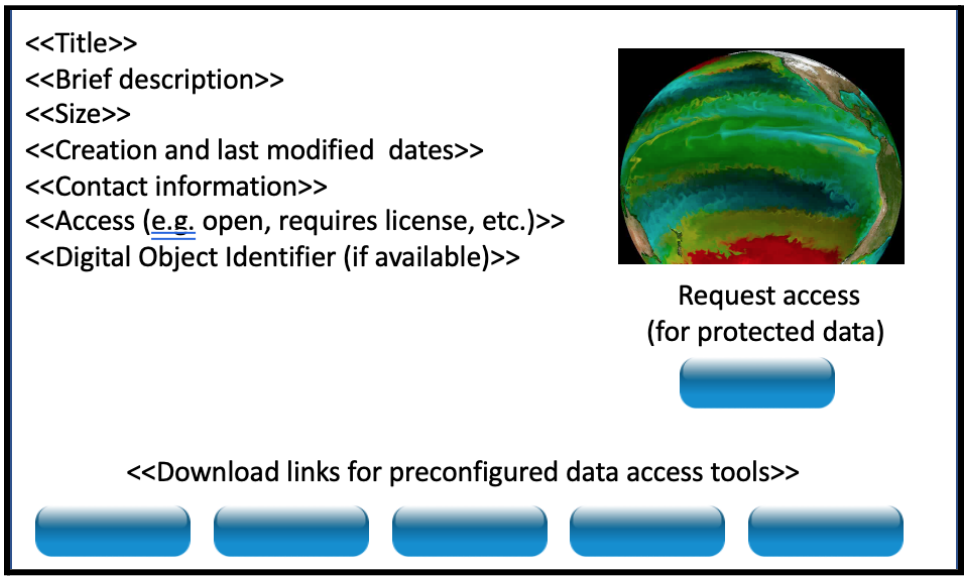
OSN users may also choose to layer more sophisticated data management applications on top of the S3 API services that OSN provides. Two applications that have been used with OSN

include Globus (using the Globus S3) connector and iRods. Both packages have detailed descriptions on how to connect the service with a S3 storage provider.

Landing Pages - Coming in 1H2021!

The data set owner may also create a landing page that follows DOI landing page conventions, making it easy to visit from a browser or data catalog. The landing page contains metadata that describes the data set and links to preconfigured, downloadable tools for accessing the data set.

OSN provides tools to create a Basic landing page that may be overridden by more sophisticated landing pages depending upon the needs of the End User community. The landing page has the following generic template.



Sample landing page

A completed template example is shown below.

Open access and Protected Data Sets

OSN datasets can be either open access or protected. In the former case, keys are only needed to write new objects to the dataset otherwise, read access can be accomplished anonymously (e.g. as shown earlier for the anonymous cyberduck configuration).

Protected datasets require keys for both reading and writing.

When an open access dataset is created only one pair of keys are created which are accessible to the data manager for a project and are used to upload data to the allocation. No keys are needed by users to download/read the data stored in the allocation.

Protected datasets have two sets of keys. One set allows writing to the dataset and is identical in function to the on previously described for open access datasets. The second set of keys provides read access to the dataset. Anonymous access is not allowed on protected datasets.

Keys are shared with users in two ways. Data managers can choose to share keys with other users “out of band” by simply sending other users keys that they are interested in via whatever secure mechanisms the project uses to store and communicate project-specific secrets (e.g. username/passwords, certificates, PKI material, access keys, etc.)

Keys can also be managed using the OSN portal.

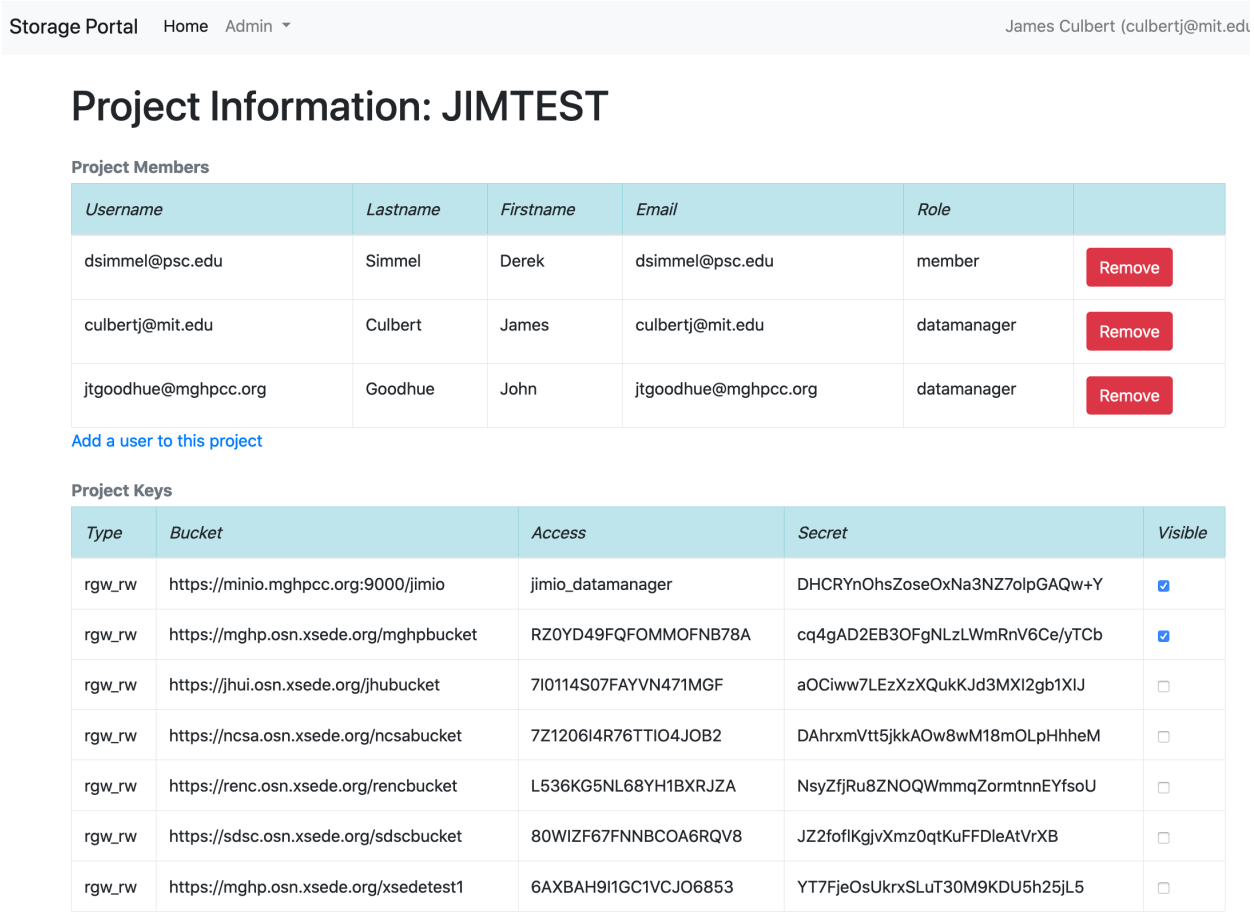
A project in the OSN portal may have multiple allocations associated with it. Data managers for a project have access to all keys for a project’s allocations. A data manager may share all project keys with another portal user by:

1) adding the portal user to the data manager’s group
2) Assigning that group member the data manager role

Once another portal user has been added to a group and given the data manager role for that group, she will have access to all the keys (and have the same privileges in the portal) as the original data manager for the group.

Data managers may also add users to a group without assigning the user the data manager role. When this happens the user will only have access to keys that the data manager has made “visible”. Visible keys are available to all group members whereas non-visible keys are only available to group members in the data manager role.

In the image below, culbertj@mit.edu and jtgoodhue@mghpcc.org have access to all the keys in the project because they are both data managers for the project “JIMTEST”, dsimmel@psc.edu will only have access to the two keys shown with the “visible” checkbox checked.



Managing Files and Data

There are four roles associated with an OSN allocation:

1. Principal Investigator
   Responsible for the allocation and serves as either the Data Manager or the Alternate Data Manager for the allocation.

2. Data Manager
   - Adds/removes data curators and data managers
   - Adds/removes end users for protected data
   - Maintain Data Set Landing Page Information
   - Monitors capacity vs utilization and requests allocation changes when needed

     The OSN Portal is used by PIs/Data managers to manage their allocations. The Portal uses CiLogon for authentication, and provides bucket administration tools to the PI/Data Manager who requested the allocation. When requesting an allocation, the PI provides an identity that is recognized by CILogon. After the bucket is created, the PI can log in to the OSN Portal and administer access to the bucket.

3. Data Curator
   - Maintains the data set

4. End User
    - Has read access to all of the data in the bucket. Public-access buckets allow access to anyone who has the name of the pod and bucket. Authenticated access buckets allow access to anyone who has the READ key.
    - Registers via any identity service that is trusted by the data manager (InCommon, ORCID, Github, Google, Amazon, etc.
    - Logs in after receiving an invitation from a Data Manager or OSN Operations

Transferring Data to the OSN

OSN data sets are comprised of Ceph Objects accessible from anywhere, via a RESTful protocol that follows S3 conventions. All end user access is via S3 put and get requests, mediated by Bucket Policies.

All put requests to OSN buckets must include an Authorization String.

There are two modes of get request:

- Protected Dataset buckets -- All requests must include an Authorization String.
- Open access buckets -- Get requests do not include an Authorization String, making the bucket accessible to anyone who has the name of the bucket and pod. The Ceph Object Gateway documentation refers to requests of this type as coming from an anonymous user.

Unlike many cloud object stores, OSN Ceph Object stores contain no information about individual users. Access to an allocation is mediated by the keys that are assigned to authorized end users. Some of the implications of this approach include:

- Object-level access control is not supported
- There is no audit trail that identifies the originator of any request
- Keys are unique to a bucket, and buckets are unique to a pod
- For example, if a data set has been replicated across two pods, each instance has a different set of keys and separately maintained access control
- Since there is one READ key per bucket, the origin of a Get request may only be distinguished by source IP address

Help

Submit an XSEDE help ticket.

Policies

Usage must comply with the OSN Acceptable Use Policy.