Protocol Plumbing: Presenting iRODS as WebDAV, FUSE, REST, NFS, SFTP, K8s CSI, and S3

Kory Draughn
Chief Technologist
iRODS Consortium

June 5-9, 2023
TNC23
Tirana, Albania

iRODS CONSORTIUM

renci

RESEARCH \ ENGAGEMENT \ INNOVATION

THE UNIVERSITY of NORTH CAROLINA at CHAPEL HILL

Our Membership

Our Mission

- Continuous Improvement
- Grow the Community
- Standardization
- Show value to our Membership

Our Business Model

Start with proof of concept

- Use Case Driven
- Hands on
- Service and Support Contracts

Consortium Membership

- Four Levels - $11k to $165k
- 10 to 300 hours of support
- Participation in Software roadmap
- Discounted hourly rate

Tier 3 Support

- Systems Integrators
- Compute Vendors
- Storage Vendors

History

- 1995 - SRB started (grid storage)
- 2004 - iRODS started (added rule engine / policy)
- 2013 - Consortium founded by RENCI, DICE, and DDN
- 2014 - Consortium accepted the code base
- 44 releases of iRODS to date

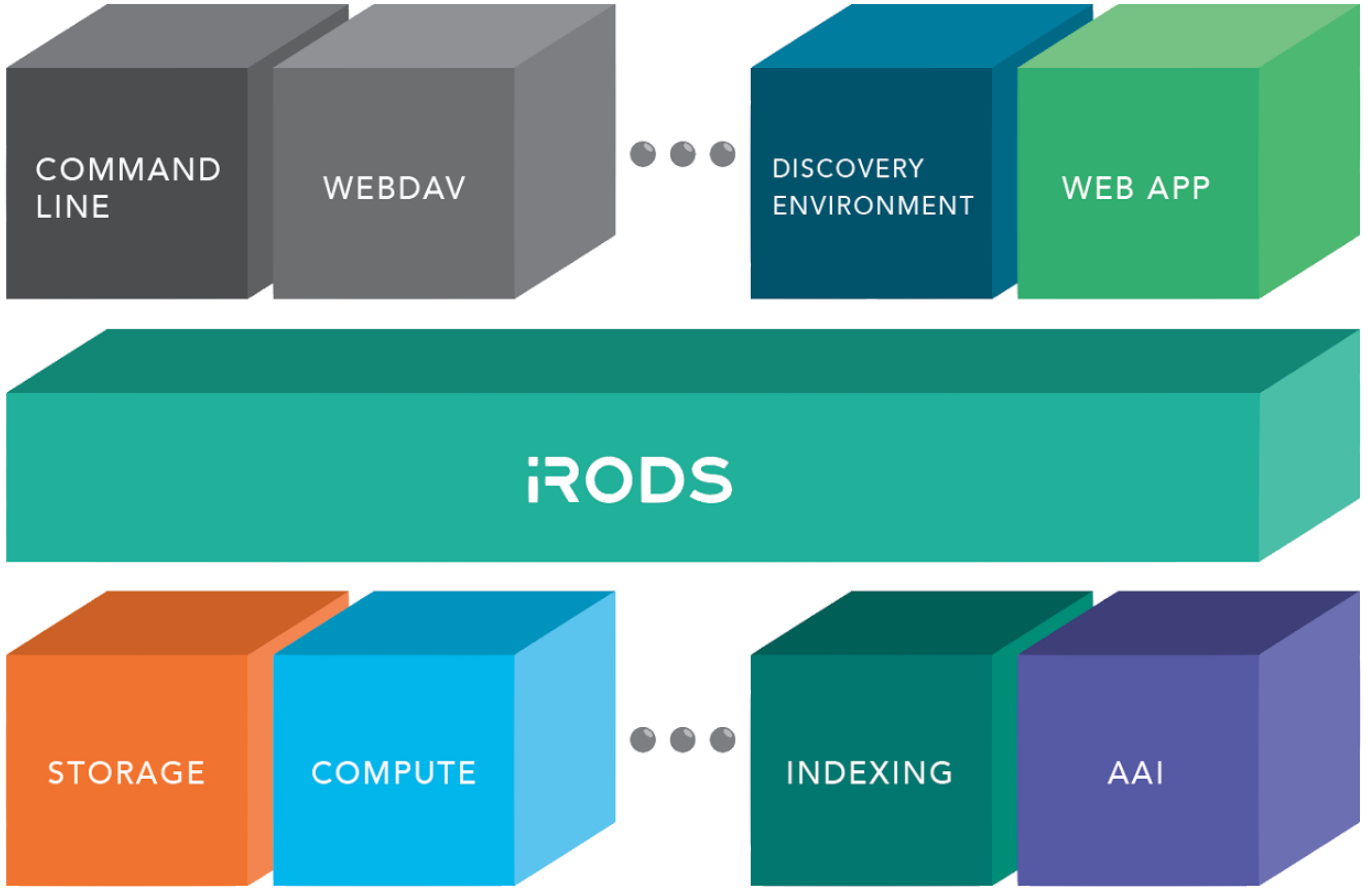
Why use iRODS?

People need a solution for:

- Managing large amounts of data across various storage technologies
- Controlling access to data
- Searching their data quickly and efficiently
- Automation

The larger the organization, the more they need software like iRODS.

iRODS as the Integration Layer



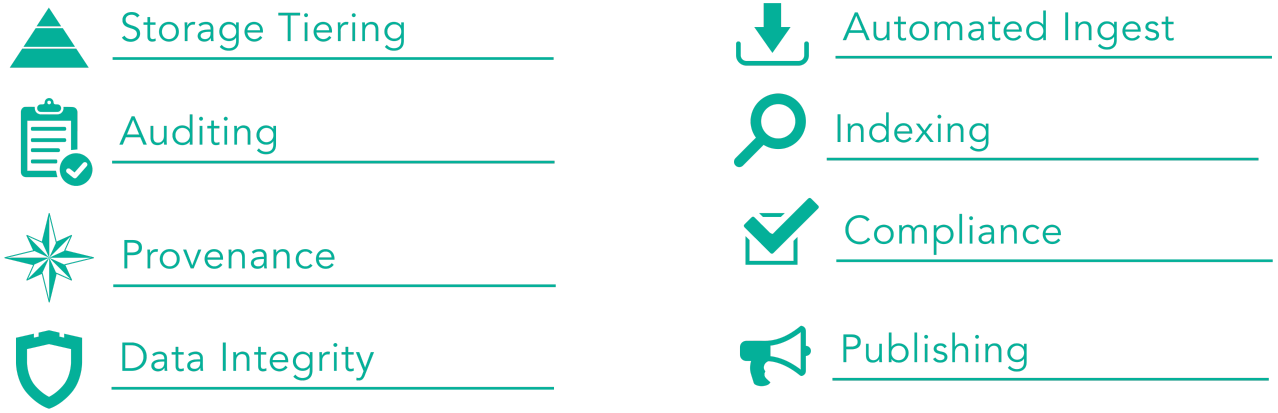
iRODS Clients

iRODS provides a layer of abstraction which integrates with your pre-existing infrastructure.

This flexibility allows your infrastructure to continue to change over time.

Existing Infrastructure

- Packaged and supported solutions
- Require configuration not code
- Derived from the majority of use cases observed in the user community



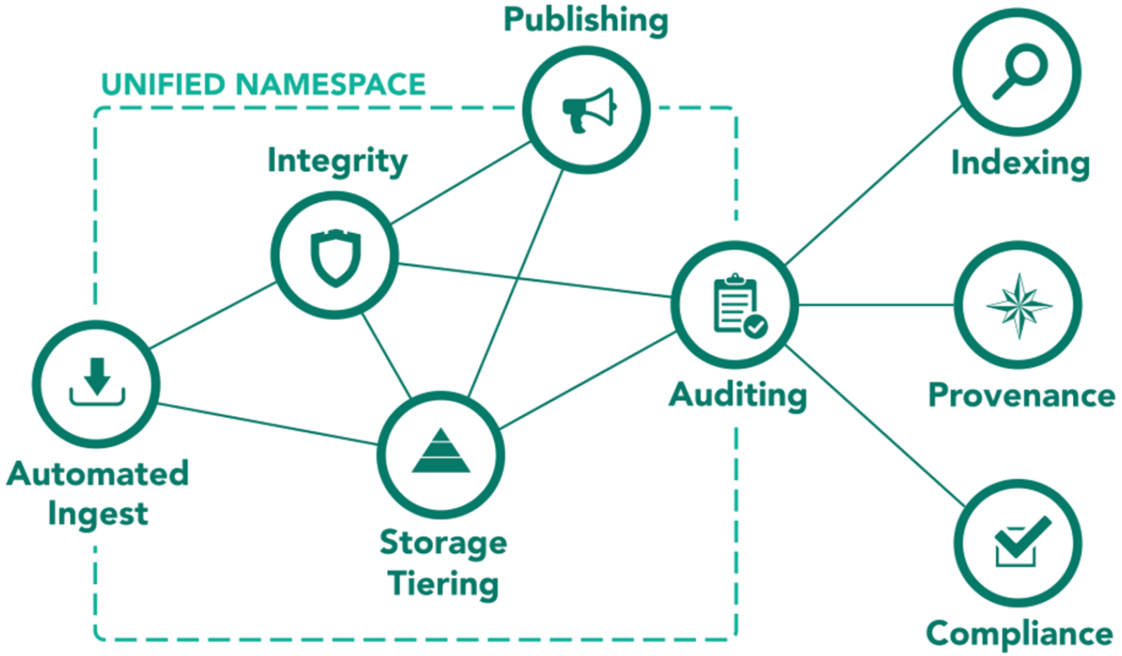
The Data Management Model

iRODS provides eight packaged capabilities, each of which can be selectively deployed and configured.

These capabilities represent the most common use cases as identified by community participation and reporting.

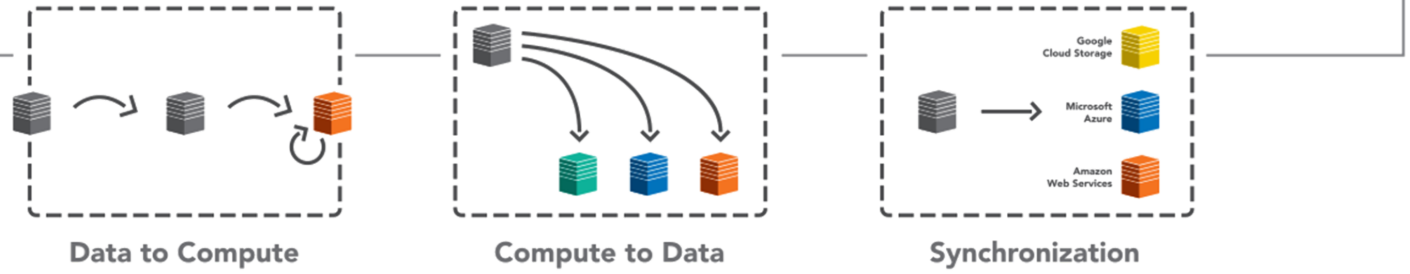
The flexibility provided by this model allows an organization to address its immediate use cases.

Additional capabilities may be deployed as any new requirements arise.



A pattern represents a combination of iRODS capabilities and data management policy consistent across multiple organizations.

Three common patterns of iRODS deployment have been observed within the community:



Over the last few years, the ecosystem around the iRODS server has continued to expand.

Integration with other types of systems is a valuable way to increase accessibility without teaching existing tools about the iRODS protocol or introducing new tools to users.

With some plumbing, existing tools get the benefit of visibility into an iRODS deployment.

Protocol Plumbing

- WebDAV
- FUSE
- REST
- NFS
- SFTP
- K8s CSI
- S3

Protocol Plumbing - WebDAV

"Davrods provides access to iRODS servers using the WebDAV protocol. It is a bridge between the WebDAV protocol and the iRODS API, implemented as an Apache HTTPD module."

Designed, developed, and maintained by

Utrecht University

Universiteit Utrecht

Open Source:

https://github.com/UtrechtUniversity/davrods

"FUSE implementation of iRODS Client written in Golang"

(2nd generation)

Designed, developed, and maintained by

CyVerse at the University of Arizona

Open Source:

https://github.com/cyverse/irodsfs

Protocol Plumbing - REST

"This REST API is designed to be deployed in front of an iRODS Server to provide an HTTP REST interface into the iRODS protocol."

(2nd generation)

Designed, developed, and maintained by

the iRODS Consortium

Open Source:

https://github.com/irods/irods_client_rest_cpp

Protocol Plumbing - NFS

"A standalone NFSv4.1 server (via nfs4j) with a Virtual File System implementation supporting the iRODS Data Management Platform."

Designed, developed, and maintained by

the iRODS Consortium

Open Source:

https://github.com/irods/irods_client_nfsrods

Protocol Plumbing - SFTP

"Support for serving local filesystem, encrypted local filesystem, S3 Compatible Object Storage, Google Cloud Storage, Azure Blob Storage, iRODS Storage or other SFTP accounts over SFTP/SCP/FTP/WebDAV."

Designed, developed, and maintained by

CyVerse at the University of Arizona

Open Source:

https://github.com/cyverse/sftpgo

"iRODS Container Storage Interface (CSI) Driver implements the CSI Specification to provide container orchestration engines (like Kubernetes) iRODS access."

Designed, developed, and maintained by

CyVerse at the University of Arizona

Open Source:

https://github.com/cyverse/irods-csi-driver

"C++ S3 API for iRODS"

Designed, developed, and maintained by

the iRODS Consortium

Open Source:

https://github.com/irods/irods_client_s3_cpp

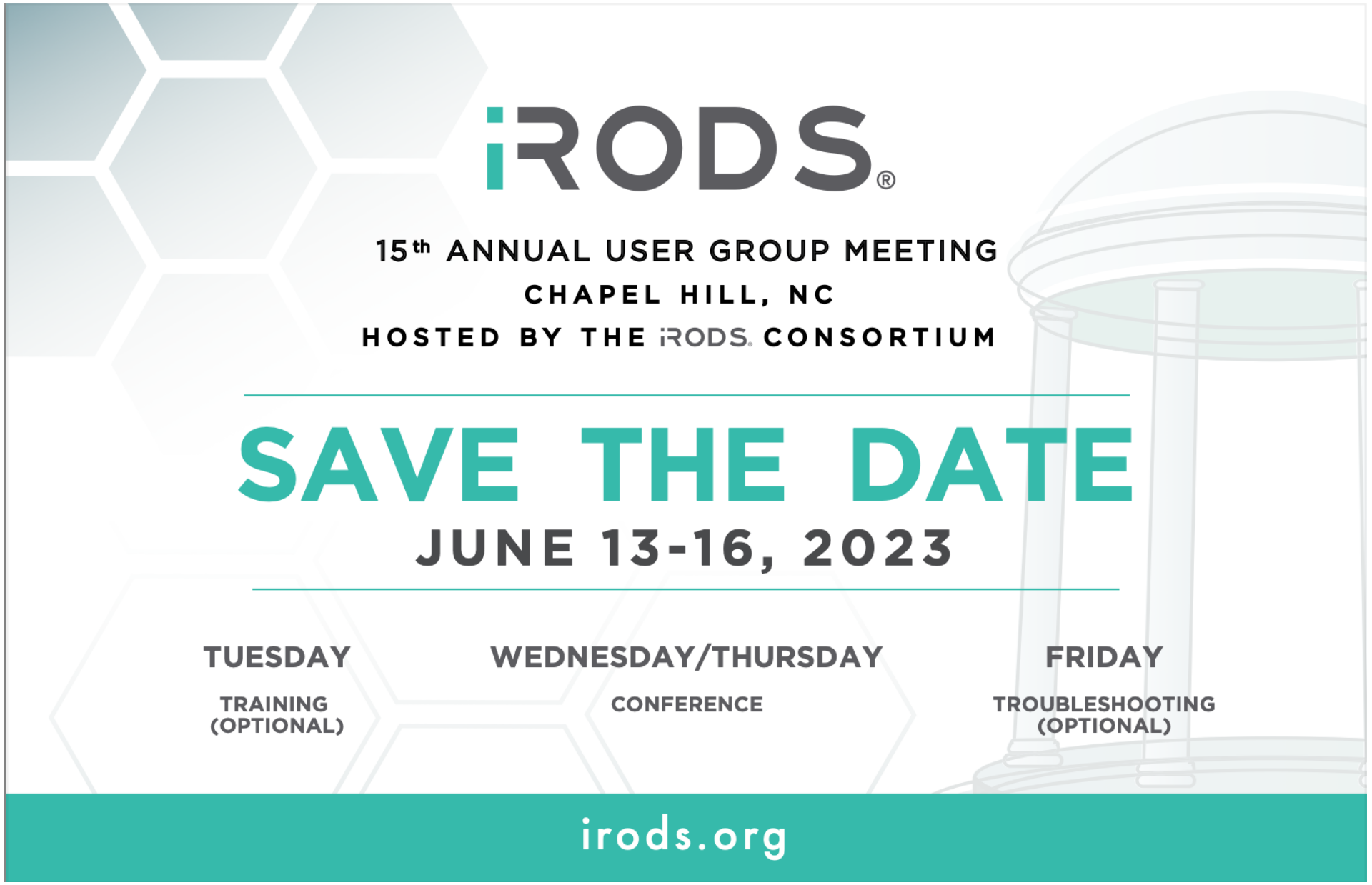
Lessons Learned

- shims are never perfect - always some impedance mismatch
- they are still very valuable and worth the effort
- maintenance burden can be minimal

Future Work

- continued identification of useful candidate protocols
- ongoing effort to stabilize existing implementations
- consideration of the iRODS server offering REST natively

Save the Date - iRODS UGM2023



Questions?

Thank you.

Kory Draughn

iRODS Consortium