

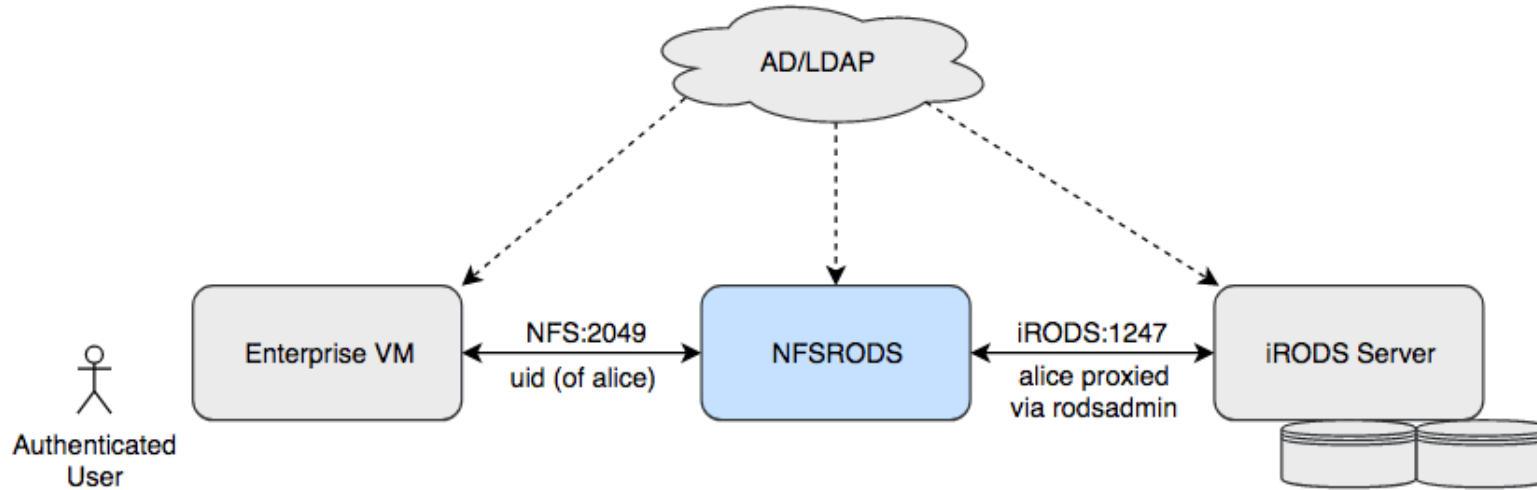
# iRODS<sup>®</sup>

## NFSRODS 2.0

Kory Draughn  
Software Developer  
iRODS Consortium

June 8-11, 2021  
iRODS User Group Meeting 2021  
Virtual Event

- Brief description of NFSRODS
- What's changed since the last release
- The list operation (i.e. `/bin/l`s)
- Future Work



A standalone NFSv4.1 server that enables presenting iRODS as a mountable filesystem.

- Configurable
- Users do not have to learn iCommands
- Users do not have to install any additional packages
- Supports many common \*nix commands and software

13 issues have been resolved.

- NFSRODS now uses Jargon Connection Caching
- The mtime rule engine plugin requirement has been relaxed
- The rename operation (i.e. mv) now supports overwriting data objects
- The portmap service message warning has been resolved

All resolved issues can be found at the following URL:

- [https://github.com/irods/irods\\_client\\_nfsrods/releases/tag/2.0.0](https://github.com/irods/irods_client_nfsrods/releases/tag/2.0.0)

Prior implementations had issues.

- Didn't handle large collections
  - Results were truncated
  - The implementation was incorrect
- Poor performance against Oracle databases
- Slow
  - Extra work
  - No caching



## The list operation (cont.)

These issues have been resolved.

- NFSRODS now caches many, if not all, results obtained through Jargon
- Added options to Jargon that enable support of Oracle. See the troubleshooting section of the NFSRODS README for details
- We learned that using a log level that results in a lot of messages degrades performance
- We learned that applying color to the output triggers more stat operations (which means more network traffic)

# The list operation (cont.)

Performance comparison between 1.0.0 and 2.0.0.

Test setup:

- iRODS provider (backed by Postgres) running on Ubuntu 16.04 (32 cores)
- NFSRODS container running on the same machine hosting the provider
- Set NFSRODS log level to INFO
- Mount command: **sudo mount -o port=2050 localhost:/ /mnt/nfsrods**
- Timing command: **time /bin/ls <collection> | wc -l**

Collection Size	1.0.0	2.0.0 (no cache)	2.0.0 (cached)
500	1m7.211s	0m0.857s	0m0.046s
1,000	3m49.246s	0m0.708s	0m0.041s
3,000	31m45.147s	0m2.345s	0m0.045s
6,000	86m56.771s (results truncated)	0m5.274s	0m0.052s
10,000	87m18.675s (results truncated)	0m9.302s	0m0.058s

- Add parallel I/O support for large file transfers
- Investigate support of metadata via extended file attributes
- Improve test coverage
- Return better error messages
- SMBRODS



Thank you!

Questions?