

# FIFE and FI Status Report

T. Levshina

# Status Update

- Disclaimer: the proposed phased approach that has been presented at the last meeting is NOT a complete plan. It is just a few steps we need to do in order to start testing some of the component. The comprehensive plan is maintained by Julie Marsh.
- The PO for CILogon Token Service has been paid!!!
- Mine started the discussion with Jim Basney. Expect to have a test Token Issuer instance by early August.
- One of the first steps is to get a token and transfer the file.
- dCache is ready to handle the request. It will completely rely on capabilities provided in the token. It supports the following capabilities:
  - storage.read
  - storage.create
  - storage.modify
  - storage.stage

```
"scope":  
  "storage.read:/disk/users/joedoe  
  storage.create:/store/mc/datasetA", ...
```

# DUNE Storage

- Multiple sites provide storage elements:

- dCache
- Castor
- DPM
- EOS

Examples:

- Fermi DCache/ENstore  
root://fndca1.fnal.gov:1094/pnfs/fnal.gov/usr/dune/tape\_backed/dunepro/protodune-sp/raw/2020/detector/cosmics/None/00/01/11/45/np04\_raw\_run011145\_0012\_dl10.root
- CASTOR  
srm://srm-public.cern.ch:8443/srm/managerv2?SFN=/castor/cern.ch/neutplatform/protodune/rawdata/np04/protodune-sp/raw/2020/detector/cosmics/None/00/01/11/45/np04\_raw\_run011145\_0012\_dl10.root
- Sample of disk path  
root://fal-pygrid-30.lancs.ac.uk/dpm/lancs.ac.uk/home/dune/protodune-sp/71/bf/np04\_raw\_run005184\_0003\_dl9.root

- Storage path “root path” (/pnfs/fnal.gov/usr) + experiment (dune/)+**WHATEVER\_YOU\_WANT** (tape\_backed, persistent, resilient, scratch...)+dataset+file\_name
- DUNE user will use Fermilab Token Issuer to obtain a token to manage files
- In order to manage a file Token Issuer has to produce capability that have information about a storage element(s) and access capabilities.
- This information should be populated by FERRY
- In order to be able to maintain this information **WHATEVER\_YOU\_WANT** should be either the same for every storage or at least do not overlap.

# dCache directory layout for DUNE

A user <uname> can read all files under dune/

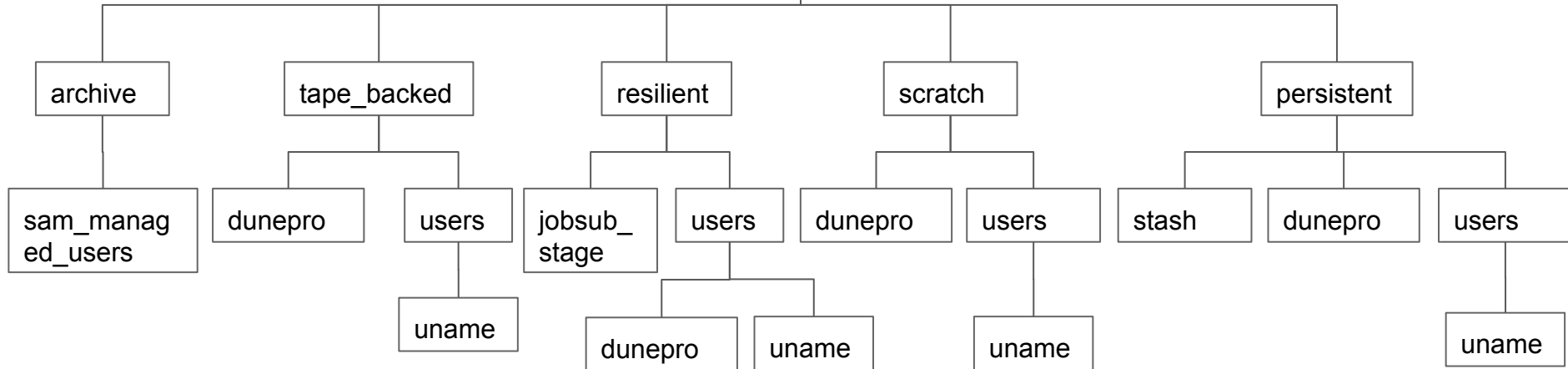
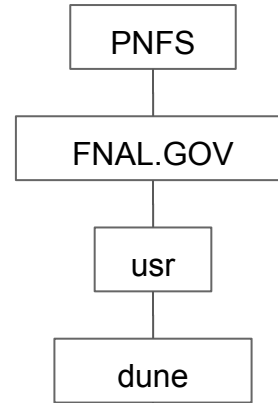
A user can <uname> write:

- dune/scratch/users/<uname>
- dune/resilient/users/<uname>
- dune/resilient/jobsub\_stage/
- dune/tape\_backed/users/<uname>
- dune/persistent/users/<uname>
- dune/persistent/stash/ContinuousIntegration

<dunepro> user can read all files under dune/

<dunepro> can write:

- dune/tape\_backed/dunepro/
- dune/scratch/dunepro/
- /usr/dune/resilient/users/dunepro/
- /usr/dune/persistent/dunepro



# Next Steps

- ProtoDune has already written 10PB of unique data, it is replicated all over the world. The collaboration has to decide how they want to structure the data across storage elements.
- CMS has done it, so it might be wise to see what directory layout they are using. The path that cmsprod is using in dcache looks like `/dcache/uscmsdisk/store/`
- We cannot rely on FERRY to be aware of directory layouts in various storages.
- The good news for FIFE experiments that they are only using dCache.
- The bad news for FIFE experiments that the currently used layouts are convoluted. It will be hard to maintain access controls if they remained unchanged.

# Other Activities

- Refining the token migration plans for FIFE services
- Working with Bonnie King and Pat Riehecky to incorporate DAQ into FERRY
  - Ingested examples of password,group and k5login files
  - Will need to work on SNOW forms and workflows design