



SAM4Users Tutorial

Pengfei Ding

FIFE workshop in 2017

June 22nd, 2017

What is SAM For Users?

- Utilities to assist **individual users** to make use of the SAM catalogue for their **own data**
- Advantages of using SAM for Users toolkit:
 - users' own data will be just like production data,
 - **submitting grid jobs using SAM project;**
 - making use of **existing tools and monitoring for SAM jobs;**
 - **moving files** between different storage locations are made simple.

List of available tools in SAM for Users toolkit

- Dataset commands:
 - `sam_add_dataset`
 - `sam_revert_names`
 - `sam_modify_dataset_metadata`
 - `sam_validate_dataset`
- Dataset copy and move:
 - `sam_clone_dataset`
 - `sam_move_dataset`
 - `sam_move2archive_dataset`
 - `sam_copy2scratch_dataset`
 - `sam_move2persistent_dataset`
- Delete datasets:
 - `sam_unclone_dataset`
 - `sam_remove_location_dataset`
 - `sam_retire_dataset`
- Miscellaneous commands:
 - `sam_archive_dataset`
 - `sam_archive_directory_image`
 - `sam_restore_directory_image`
 - `sam_prestage_dataset`
 - `sam_audit_dataset`
 - `sam_condense_dataset`
 - `sam_pin_dataset`

* Examples can be found in this tutorial

Hands-on session

- Required setups;
- Access files in scratch dCache:
 - Write, read and delete files;
- Using sam4users tool to:
 - Declare a dataset with files in scratch area;
 - Store files to persistent or tape-backed area;
 - Remove replicas of the dataset in the scratch area;
 - Validate dataset and what to do when a file is missing;
 - Retire a dataset.
- Commands in this session can be found at:
- http://home.fnal.gov/~dingpf/sam4users_tutorial_commands.txt

Setups

```
# On GPVM (e.g. dunegpvm01.fnal.gov)
```

```
# setup UPS etc.
```

```
source /cvmfs/dune.opensciencegrid.org/products/dune/setup_dune.sh
```

```
# Getting a valid certificate and VOMS proxy
```

```
kx509
```

```
voms-proxy-init -noregen -rfc -voms dune:/dune/Role=Analysis
```

```
# Setup fife_utils, current version is v3_1_0
```

```
setup fife_utils
```

```
# set experiment name
```

```
export EXPERIMENT=dune
```

Access file in dCache (I) – copy files to scratch

Create a directory in scratch area for this tutorial

```
export SCRATCH_DIR=/pnfs/dune/scratch/users/${USER}/tutorial
ifdh mkdir_p ${SCRATCH_DIR}
```

Write files to scratch dCache (best to have files written in local
disk or BlueArc first and then **copy** to the scratch area with ifdh
or xrootd)

create four 5MB dummy files, these files will be used for
demonstration of data handling. You do not need to create the dummy
files. You can use files of your own.

```
for i in `seq 0 3`; do \  
head -c 5242880 /dev/urandom > ~/dummy_${USER}_${i}.bin; \  
done
```

copy files into scratch dCache with “ifdh cp”.

```
ifdh cp -D ~/dummy_${USER}_[0-3].bin ${SCRATCH_DIR}
```

To explore other options available with “ifdh cp”, just type “ifdh”.

Access file in dCache (II) – delete files in scratch

```
# delete files with "ifdh rm"
```

```
ifdh rm ${SCRATCH_DIR}/dummy_${USER}_0.bin  
for i in seq `1 3`; do\  
ifdh rm ${SCRATCH_DIR}/dummy_${USER}_${i}.bin;\  
done
```

```
# Copy files to scratch dCache using xrootd
```

```
xrdcp ~/dummy_${USER}_{0-3}.bin ${SCRATCH_DIR}
```

```
# or
```

```
xrdcp ~/dummy_${USER}*.bin \  
root://fndca1.fnal.gov:1094//pnfs/fnal.gov/usr/dune\  
/scratch/users/${USER}/tutorial
```

```
# note that one should convert the path to scratch dCache to URI
```

```
# recognized by xrootd:
```

```
# e.g. from: /pnfs/dune/scratch/users/${USER}/dummy_${USER}_1.bin
```

```
#           to: root://fndca1.fnal.gov:1094//pnfs/fnal.gov/usr/dune\  
#           /scratch/users/${USER}/dummy_${USER}_1.bin
```

Store files to persistent/tape-backed area (I)

- declare a SAM dataset with files in scratch area

choose a dataset name, better to be user, purpose and time specific

```
export TUTORIAL_DATASET=${USER}_tutorial_`date +%y%m%d%H%M`_01
```

Add a SAM dataset for files in dCache scratch area

```
sam_add_dataset -n ${TUTORIAL_DATASET} -d ${SCRATCH_DIR}
```

Instead of the “-d” option, it can take “-f” option followed by a

text file containing a list of paths to files

NOTE: sam_add_dataset will change the filename with UUID prefix.

```
ls ${SCRATCH_DIR}
```

List files in the dataset

```
samweb list-definition-files ${TUTORIAL_DATASET}
```

Store files to persistent/tape-backed area (II)

- clone the dataset to persistent/tape-backed area

```
# If the files under scratch area worth being kept for longer time,  
# they can be added to SAM first with sam_add_dataset, followed by  
# copying to the persistent or tape-backed area.
```

```
# create a destination directory in the persistent area first  
export PERSISTENT_DIR=/pnfs/dune/persistent/users/${USER}/tutorial  
mkdir -p ${PERSISTENT_DIR}
```

```
# Copy the dataset to persistent area with sam_clone_dataset  
sam_clone_dataset -n ${TUTORIAL_DATASET} -d ${PERSISTENT_DIR}
```

```
# Advanced tips for cloning large dataset:  
# “sam_clone_dataset” has “--njobs” option to launch multiple jobs to do  
# the cloning. “launch_clone_jobs” can launch grid jobs to do the cloning.
```

Store files to persistent/tape-backed area (III)

- remove replicas in the scratch area

check file locations, you will see two locations.

```
DUMMY_01=`samweb list-definition-files ${TUTORIAL_DATASET}|head -n 1`  
samweb locate-file ${DUMMY_01}
```

Remove replicas of the dataset files in the scratch area

```
sam_unclone_dataset -n ${TUTORIAL_DATASET} -d ${SCRATCH_DIR}
```

List \${SCRATCH_DIR} to check if files are still there.

```
ls ${SCRATCH_DIR}
```

check the file locations again, you will see only one location left

```
samweb locate-file ${DUMMY_01}
```

Store files to persistent/tape-backed area (IV)

- validate dataset and dealing with missing files

Validate dataset, that is to check if each files in a dataset exists
in the storage volume

```
sam_validate_dataset -n ${TUTORIAL_DATASET}
```

Let's move one file in the dataset and run "sam_validate_dataset"

```
FPATH=`samweb locate-file ${DUMMY_01}|cut -d ':' -f 2`
```

```
ifdh mv ${FPATH}/${DUMMY_01} \
```

```
sam_validate_dataset -n ${TUTORIAL_DATASET}
```

When there is a file missing, one can either replace the file with
a backup copy; or use "--prune" option to remove the file from the
dataset; otherwise there will be errors when using SAM record for
file access.

```
sam_validate_dataset -n ${TUTORIAL_DATASET} --prune
```

Let's list the files in the dataset again

```
samweb list-definition-files ${TUTORIAL_DATASET}
```

Store files to persistent/tape-backed area (V)

- retire dataset

```
# This will delete the dataset definition in SAM, retire all files  
# contained in the dataset and delete them from disk. To be safe, use  
# this command with “-j” (“--just_say”) option first to see what will  
# be done before letting it take real action.
```

```
sam_retire_dataset -n ${TUTORIAL_DATASET} -j
```

```
# You can use “--keep_files” option if you don’t want to delete the  
# files.
```

```
sam_retire_dataset -n ${TUTORIAL_DATASET} --keep_files
```

```
# Once the dataset being retired, you can revert the file names for the  
# last copy of files with sam_revert_names
```

```
sam_revert_names -d ${PERSISTENT_DIR}
```

Summary (I)

- We have just gone through a full lifecycle of dataset files in the hands-on session;
- Please follow these practices in your own data management tasks, and keep the following things in mind:
 - Avoid using BlueArc area for grid jobs;
 - Avoid using “rsync” on any dCache volumes;
 - Store files into dCache scratch area first;
 - Always have files under persistent or tape-backed area bookkept by SAM;
 - Access files in dCache volumes via NFS is not as reliable as using “ifdh” or “xrootd”.

Summary (II)

- With SAM for Users toolkit, one can:
 - Add own files to SAM
 - Copy/move dataset files between different storage locations
 - No accidents of deleting files
 - Most importantly: various tools for using production data are now available to users' own data.
- Additional links
 - Understanding storage volumes
https://cdcv.s.fnal.gov/redmine/projects/fife/wiki/Understanding_storage_volumes
 - SAM4Users wiki
https://cdcv.s.fnal.gov/redmine/projects/sam/wiki/SAMLite_Guide
 - SAM wiki
https://cdcv.s.fnal.gov/redmine/projects/sam/wiki/User_Guide_for_SAM

Backup

Modify file metadata (I)

- File metadata:
 - `samweb get-metadata 43ccc572-d856-4413-8f41-535fd66755bf-nearnet_r00011382_s15_nuexsec.root`

```
File Name: 43ccc572-d856-4413-8f41-535fd66755bf-nearnet_r00011382_s15_nuexsec.root
File Id: 212505656
Create Date: 2016-06-21T16:12:22+00:00
User: dingpf
File Type: unknown
File Format: unknown
File Size: 6082761
Checksum: (none)
Content Status: good
Dataset.Tag: dingpf_test_20160621
```

Suggestion for experiments' SAM admins:

- add metadata parameters for users' own data;
- ask users to only modify metadata for those parameters.

```
novagpvm11: ~ $ samweb add-parameter Users.AnalysisGroup string
```

Modify file metadata (II)

- Modify file metadata for a single file:
 - `samweb modify-metadata ${FILE_NAME} ${METADATA_JSON_FILE}`

```
novagpvm11: ~ $ cat ~/file.json
{
  "Users.AnalysisGroup": "NDXsec",
  "file_format": "root"
}
novagpvm11: ~ $
```

```
novagpvm11: ~ $ samweb get-metadata 3f2346bb-a886-415a-9515-74fd3b463c57-neardet_r00011361_s08_nuexsec.root
File Name: 3f2346bb-a886-415a-9515-74fd3b463c57-neardet_r00011361_s08_nuexsec.root
File Id: 212503465
Create Date: 2016-06-21T16:12:23+00:00
User: dingpf
Update Date: 2016-06-21T16:33:18+00:00
Update User: dingpf
File Type: unknown
File Format: root
File Size: 6424415
Checksum: (none)
Content Status: good
Dataset.Tag: dingpf_test_20160621
Users.AnalysisGroup: NDXsec
```

Modify file metadata (II)

- Modify file metadata for all files in a dataset:
 - `sam_modify_dataset_metadata -n {DATASET_NAME} -m ${META_DATA_STRING_JSON}`
- Or use SAM python API

```
import samweb_client as swc
sam = swc.SAMWebClient("nova")
mfmetadata = {"Users.AnalysisGroup": "NDXsec"}
f="3f2346bb-a886-415a-9515-74fd3b463c57-nearnet_r00011361_s08_nuexsec.root"
sam.modifyFileMetadata(f,mfmetadata)
```

```
novagpvm11: ~ $ samweb list-files "Users.AnalysisGroup NDXsec"
3f2346bb-a886-415a-9515-74fd3b463c57-nearnet_r00011361_s08_nuexsec.root
```