

Rucio and Xcache for User Analysis.

Wei Yang

Make it easier for users to access “distributed data”

Why we care about this ?

US ATLAS funds two User Analysis Facilities (aka. Shared Tier 3s)

- Hosted @ BNL and @SLAC
 - Because for some tasks, conventional login and batch at the Analysis Facilities are more efficient for users than the Grid
 - Plus there are supports on unix platform and on ATLAS software environment
- Our (@BNL & @SLAC) tasks are to provide resources (HW and SW) to facilitate user analysis activities.
- One of these tasks is to help them access the official ATLAS data products.
 - They are distributed at many sites.
 - They are managed by Rucio

How do users access ATLAS data products

Grid model:

- User [R2D2](#) to move data to ATLAS_XXX DISK at BNL or SLAC
 - Unimportant requests (e.g. user requests) will suffer long tails in completion - be patient !
- Then (excerpt from [US ATLAS analysis tutorial](#)):
 - `lsetup rucio`
 - `voms-proxy-init -voms atlas`
 - `rucio list-file-replicas --rse SLACXRD_LOCALGROUPDISK --protocols root $dset # find data location`
 - `inputFile=root://griddev03.slac.stanford.edu//xrootd/atlas/atlaslocalgroupdisk/rucio/data16_13TeV/f9/bd/DAOD_SUSY15.11525262._000003.pool.root.1`
 - `../bin/Exam_JetsPlot $inputFile`

How can Xcache help?

Use remote access,

- **Eliminate the R2D2 step** ← this save time
- Instead, find data location (remote URL)

Use Xcache to hide the access latency

- [Study showed that Xcache can help hiding long network latency during remote access](#)
 - My guess: probably due to aggressive readahead
- Xcache allows using cascaded URL format
 - `root://xcache.bnl//root://cern.ch//eos/atlas/rucio/data16_13TeV/ab/dc/xAOD.root`

Maintained
work by users

Users need to keep track of data location change

global Logical File Name

gLFN is data location agnostic

- A concept developed by the FAX project
 - `root://global-redirector//atlas/rucio/scope:file` ← this is Rucio DID
- The FAX redirector network would route the user to the actual data location
- FAX is decommissioned. But Rucio provides the same info for redirection !

So with Rucio and Xcache, we want to

- Free user from R2D2 and long wait
- Free user from finding and keeping trace of data location
- In the meantime, hide remote data access latency
- A single command gives all gLFNs of a dataset
 - `rucio list-files --csv $dset | gawk -F, '{printf("root://myXcache_changeme//atlas/rucio/%s\n", $1)}'`

A Rucio N2N plugin for Xcache

[A Xcache / Xrootd N2N plugin](#) to achieve those goals

- Identify cascade URL or gLFN (this is easy)
 - For gLFN, query Rucio for all data locations, listed in a “metalink”
 - Cache metalinks for 24 hours -- reduce load on Rucio
 - For cascade URL, generate a metalink (to use the same backend mechanism)
 - Overwrite every time
 - Xrootd client treats metalink as a virtual redirector
 - Xrootd team spent significant effort to enhance handling of site failure
- Identify Rucio managed files and non-Rucio managed files
 - Optional, not easy, has to depend on keyword (such as /rucio/) in path
 - But worth doing so: Rucio managed file at all locations share a single cache entry
- Cluster-able
 - It is how Xrootd scales up !

This makes
the plugin
invisible

Examples of using Xcache

```

root-c++.ipynb x pyroot.ipynb x uproot.ipynb x truth.C x
Python 3

[1]: import uproot
      prefix / scope : file

[2]: file = uproot.open("http://atlfax:8080/atlas/rucio/data16_13TeV:DAOD_SUSY15.11525262._000003.pool.root.1")

[3]: file.keys()
      Xcahce @ SLAC, also available at BNL
      • root://atlfax// (double // after hostname) or
      • http://atlfax:8080/... (use http:// - no xrootd binding in python3 yet).

[3]: [b'##Shapes;1',
      b'##Links;1',
      b'##Params;1',
      b'CollectionTree;1',
      b'POOLContainerForm;1',
      b'POOLContainer;1',
      b'POOLCollectionTree;1',
      b'MetaData;1',
      b'MetaDataHdrForm;1',
      b'MetaDataHdr;1']
      Xcache path /atlas/rucio/scope:file is location independent
      • You need to know what files are needed for input.
      • But you don't need to keep trace of the files' physical locations
      • Accessed file will be cached to speed up future access
      • So far we don't require authentication with X509
[ ]:
      • uproot seems does not allow X509 authentication (no https://)
      • So uproot has to access remote files via Xcache
  
```

Summary

Rucio and Xcache work together to easy user access to their data

- The plugin is available as RPM as WLCG and OSG
- Not limited to ATLAS
- It is transparent to other use cases

Work by FAX, AAA, WLCG DOMA TPC WG and ACCESS WG expand the remote access door coverage to almost all sites

This is only one of the Xcache use cases in working with Rucio