



# XCache Packaging

Brian Lin  
OSG Software  
University of Wisconsin–Madison





# OSG Software and XCache

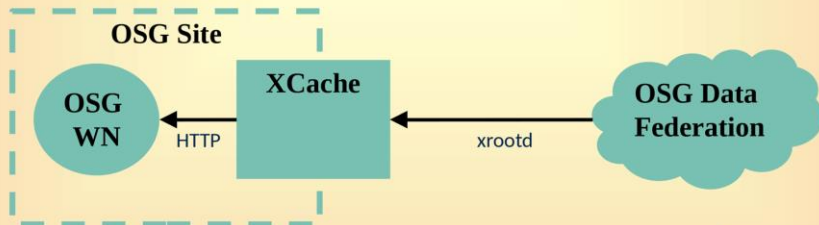
- The OSG Software Team provides RPM packaging and repositories for grid software deployed at OSG sites
- Most software distributed by the OSG have their own upstream developers, so the OSG Software Team focuses on the curation and integration of the software stack via helper tools, packaging, occasional patches, etc.
- Through IRIS-HEP (<https://iris-hep.org/>), the OSG and U.S. LHC have decided to combine efforts for development and packaging of a common XCache configuration
- Additionally, we have been working on improving the XRootD release pipeline:
  - XRootD release candidates are distributed via OSG Yum repositories to reach a wider audience of users
  - Releasing XCache as a container
    - <https://hub.docker.com/r/opensciencegrid/atlas-xcache/>
    - <https://hub.docker.com/r/opensciencegrid/stash-cache/>
    - CMS XCache coming soon...

# XCache Deployment Architectures

## Frontends, backends, and more.

At a high level, the difference between deployments is the frontend and backends.

StashCache:



US ATLAS Facility Meeting

Dec 3-5 18

14



- Stash Cache origins (based on XRootD) serve as the sources for the OSG Data Federation
- The only deployment where the caches use HTTP/S
- GSI-authenticated Stash Caches are served via a separate XRootD process
- Stash cache and origins are currently available in the OSG Yum repositories as part of the XCache 1.0.5 source RPM

Image source:

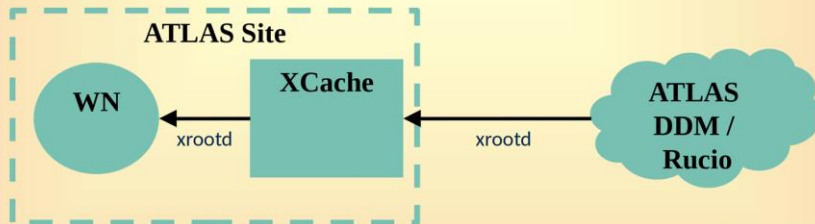
<https://indico.cern.ch/event/766802/contributions/3223469/>

# XCache Deployment Architectures

## Frontends, backends, and more.

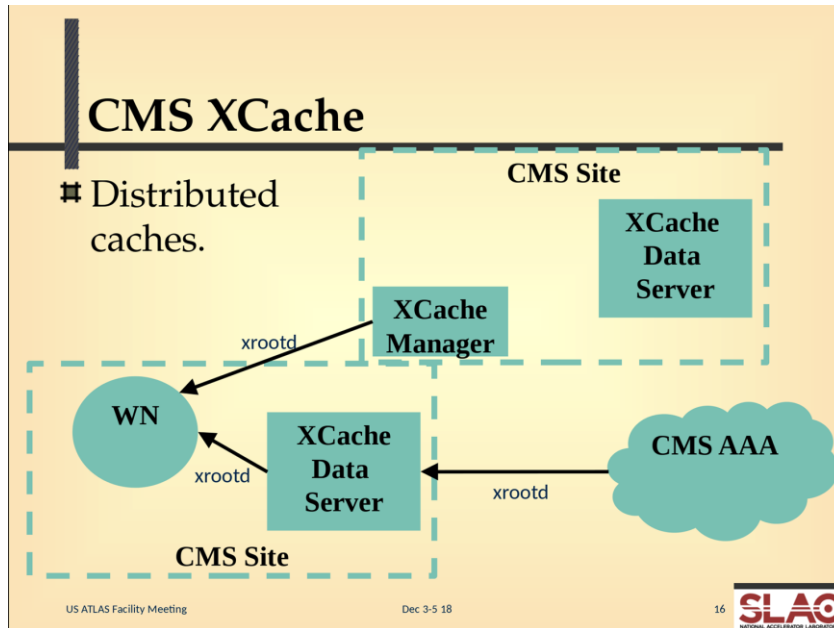
‡ There are also differences in authentication, monitoring infrastructure, and authorization.

‡ ATLAS XCache:



- Rucio serves as the source for the ATLAS data federation
- So ATLAS XCache utilizes the XRootD-Rucio plugin
- ATLAS uses its own tools to parse .cinfo files and report monitoring

# XCache Deployment Architectures



- Each CMS XCache (based off the UCSD instance) runs both XRootD and cmsd

Image source:  
<https://indico.cern.ch/event/766802/contributions/3223469/>



# Common XCache Configuration

- Common configuration between deployments is quite small
- AuthN, authZ, and monitoring are clearly deployment-specific
- But there appear to be some easily overridable tuning variables that we can add to the common configuration, e.g. `pfc.writequeue`, `xrd.network`, various `pss` options, etc.
- `all.export` directives for the three deployment do not conflict with each other
- And then there's `oss.space`, utilized by both ATLAS and CMS, but it's not clear if this is recommended

```
# site specified
oss.localroot $(rootdir)
# VO specified
pss.origin $(originhost)

# see next slide
xrd.port 1094

ofs.osslib libXrdPss.so
pss.cachelib libXrdFileCache.so

# with minimum default values
pfc.blocksize $(blocksize)
pfc.diskusage $(min_du) $(max_du)
pfc.ram $(ram)
pfc.prefetch $(prefetch)

all.adminpath /var/spool/xrootd
all.pidpath /run/xrootd

# minimal logging
ofs.trace delay
pfc.trace info
cms.trace defer redirect stage
```



# Configuration Headaches

- Some configuration variables cannot be overridden, i.e. setting the same variable twice takes the first value. Known examples are `http.port` and `pss.origin`.
- A `continue` statement causes the rest of the file to be ignored (and isn't documented). Since Stash Cache/Origin and CMS XRootD/cmsd share some common configuration, allowing multiple continue statements would be helpful:

```
# File: xrootd-cms-xcache-cmsd.cfg  
  
continue /etc/xrootd/cms-common.cfg  
  
all.export /store stage r/o  
cms.trace defer redirect stage
```

```
# File: xrootd-cms-xcache.cfg  
  
continue /etc/xrootd/cms-common.cfg  
continue /etc/xrootd/xcache.d/
```



# XCache Containers

- Containers are built off of the latest CentOS container with freshly built XRootD and XCache RPMs (including release candidates) [1]
- Each container runs multiple processes (xrootd processes, fetch-crl, helper scripts) via Supervisor
- ‘Stable’ images for Stash Cache and Stash Origin as well as ‘fresh’ images for ATLAS XCache are currently available. CMS XCache is under review [2].
- Stash Cache and ATLAS XCache containers are currently deployed in production at Amsterdam and AGLT2, respectively

[1] <https://opensciencegrid.org/technology/policy/container-release/>

[2] <https://github.com/opensciencegrid/docker-xcache/pull/19/>





# Future Work

- Release XCache 1.1, including packages for ATLAS and CMS XCaches, along with fixes for authenticated Stash Origin [IN PROGRESS]
- Add integration tests for container validation [IN PROGRESS]
- Document ATLAS and CMS XCache installations
- Document XCache container installation and operation
- Deliver an XCache package that can serve all three data federations (??)
- Kubernetes and/or Helm packaging (??)