UNIVERSITY OF BIRMINGHAM

XCache at Birmingham

GridPP Collaboration Meeting, 29th August, 2019

Mark Slater, Birmingham University

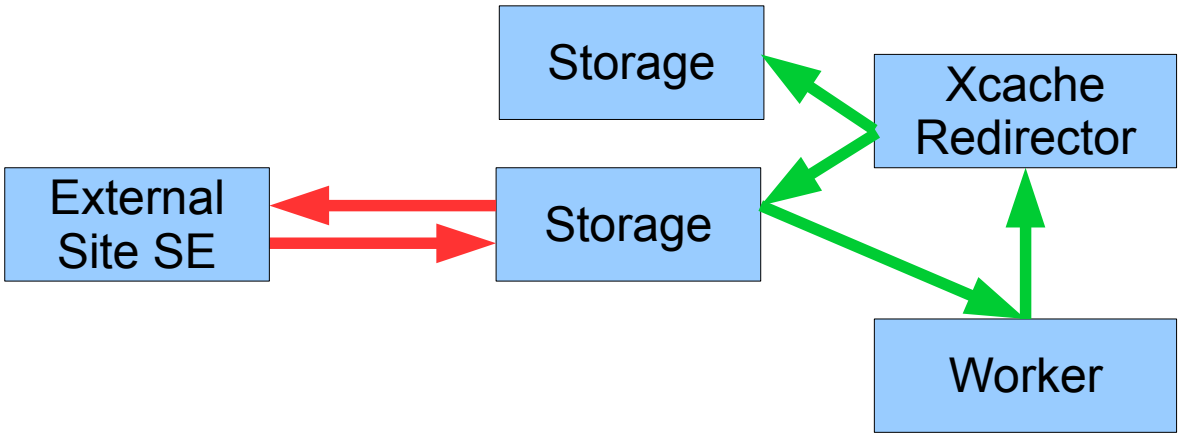
UNIVERSITY OF BIRMINGHAM

What is Xcache?

Xcache is a general squid-like caching mechanism based on XRootD and is designed to cache data files from external sources at a location

It is being used by a number of experiments though I will be talking specifically about using it for Atlas in place of a full-blown SE endpoint

To allow integration with RUCIO, there is an Atlas specific plugin that provides some namespace mapping and (I believe!) can send cache information to RUCIO for data tracking



root://xcache.bham.ac.uk/root://dpm.manc.ac.uk/..

UNIVERSITY OF BIRMINGHAM

Why Switch to XCache

After making the decision to switch off DPM and just run with EOS, I've been in discussions with Atlas wrt how best to use the storage at Bham

After initially looking at a separate ATLAS instance of EOS, I was then pointed towards Xcache instead

There are several possible benefits for both the site and Atlas:

- Easy to setup
- Lower maintenance/administrative overhead
- Small storage is still useful
- Smaller network load on the larger sites
- Fewer and less complicated services to manage
- Fewer storage endpoints for Atlas to maintain

As Bham would be a good case study for the wider adoption of Xcache in Atlas, I thought it would good to give it try!

Hardware

The current hardware dedicated to Xcache is:

- Basic server (16C, 24GB) to act as redirector
- Two 40TB pool nodes
- ~400 cores for Atlas → 50 simultaneous 8-core jobs
- Pool nodes are connected at 10Gb/s, workers at 1Gb/s.

In theory, I have more storage available though I'm not certain if it's worth dedicating to Xcache yet

UNIVERSITY OF BIRMINGHAM

Xcache Installation

There are 2(+?) options for setting it up:

Using SLATE:

This means allowing DDM to run Docker containers on the cluster. Pros being you don't have to setup it up or manage it yourself, Cons that you have to allow an outside source root access to machines

Manual install and configuration:

All setup and config is done by sys-admin but full control is maintained

I chose the latter and got the install instructions from the singularity definition here:

https://raw.githubusercontent.com/wyang007/rucioN2N-for-Xcache/master/xcache.singularity.def

This came down to:

- Enable the WLCG Repo
- Install some basic packages (xrootd server/client, xrootd-rucioN2N-for-Xcache, etc.)
- Add an xrootd user
- Create a startup script to launch the xrd and cmsd daemons
- Add appropriate firewall holes

Monitoring

I am currently monitoring Xcache using my own scripts that pull info from the logs and the OS. These check:

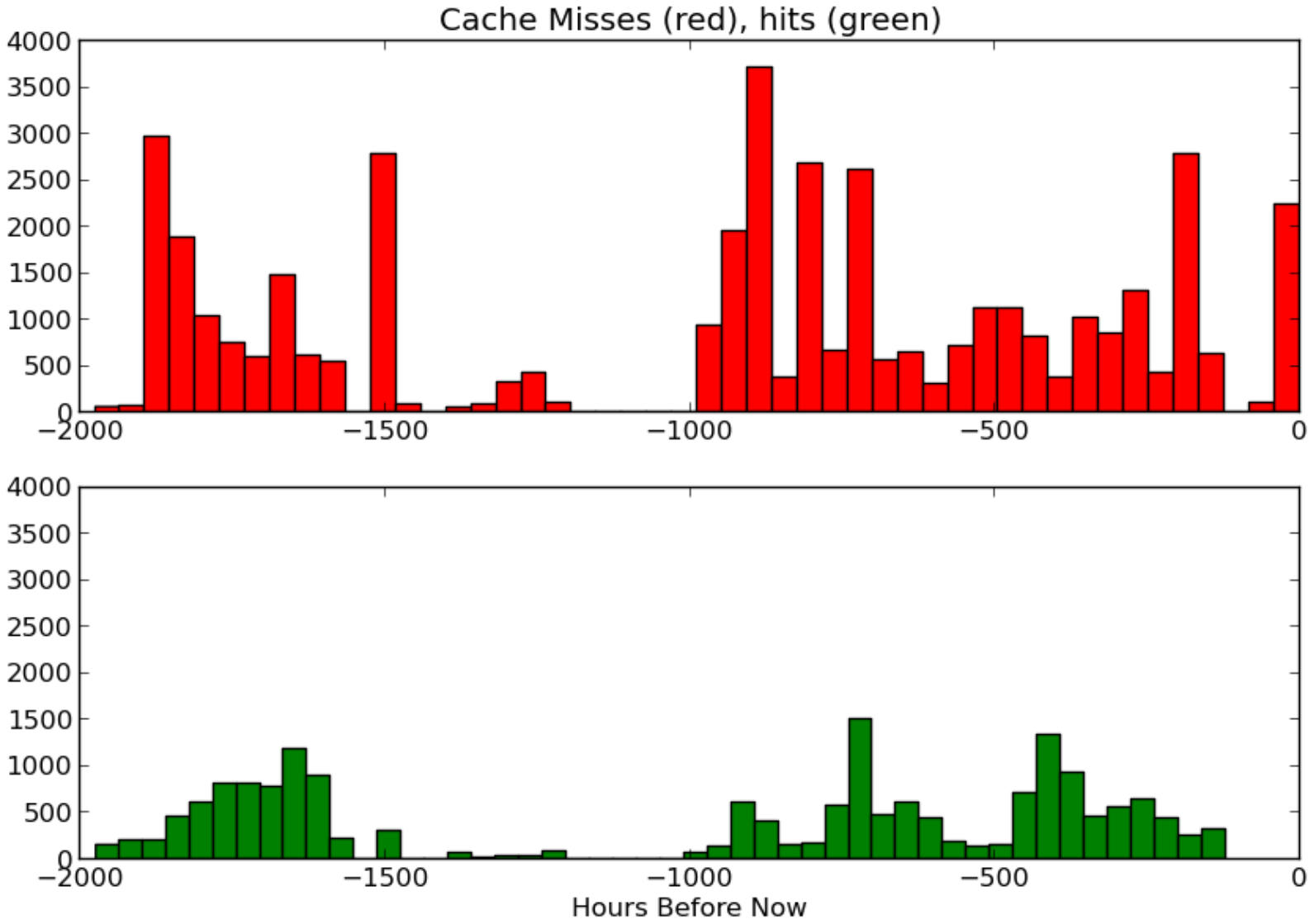
- Cache Hit/Miss Rates
- Cache usage
- Network rates

These scripts feed my Graphite/Grafana monitoring and can also be run offline to allow studies after the fact

Note that Teng has got some a monitoring container available that would fill an ELK stack at RAL. Unfortunately, I haven't had chance to implement this yet but I do plan to!

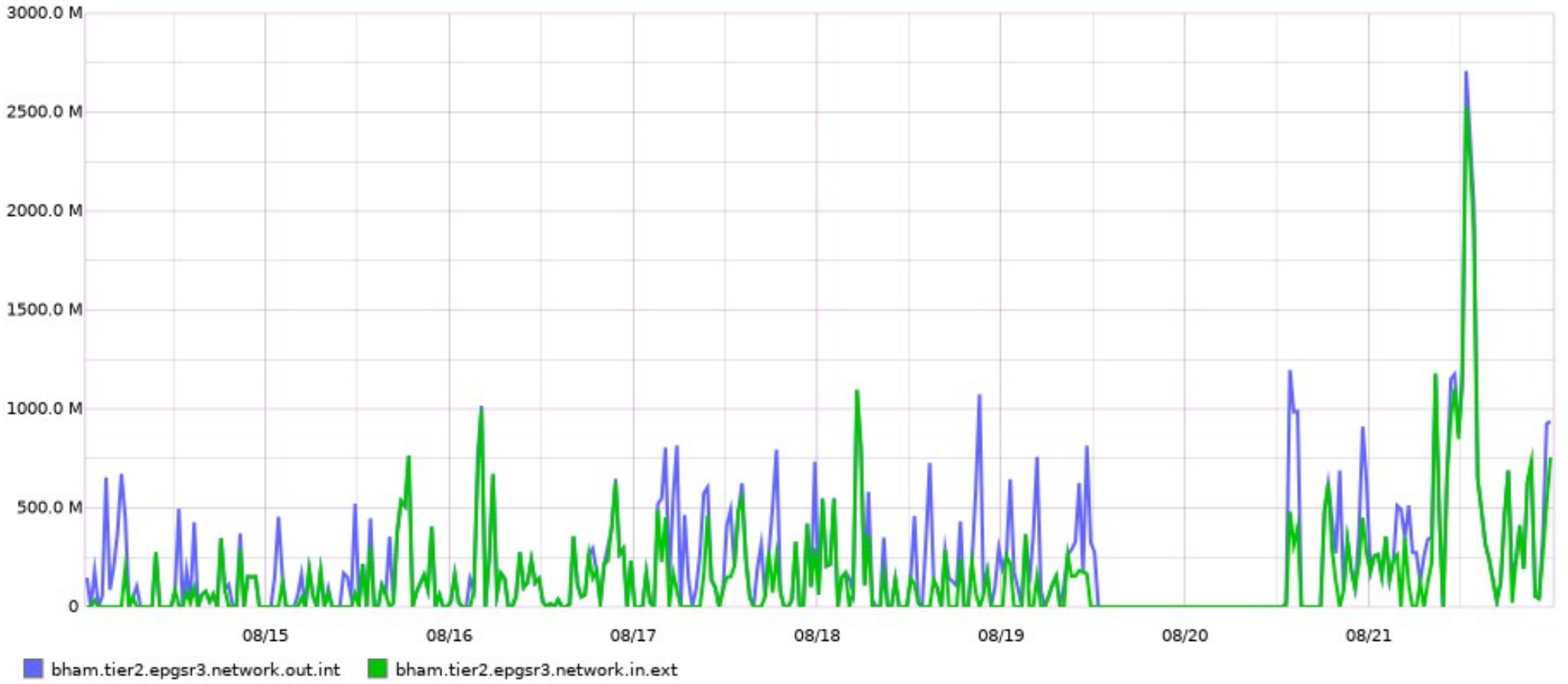
UNIVERSITY OF BIRMINGHAM

Cache File Access



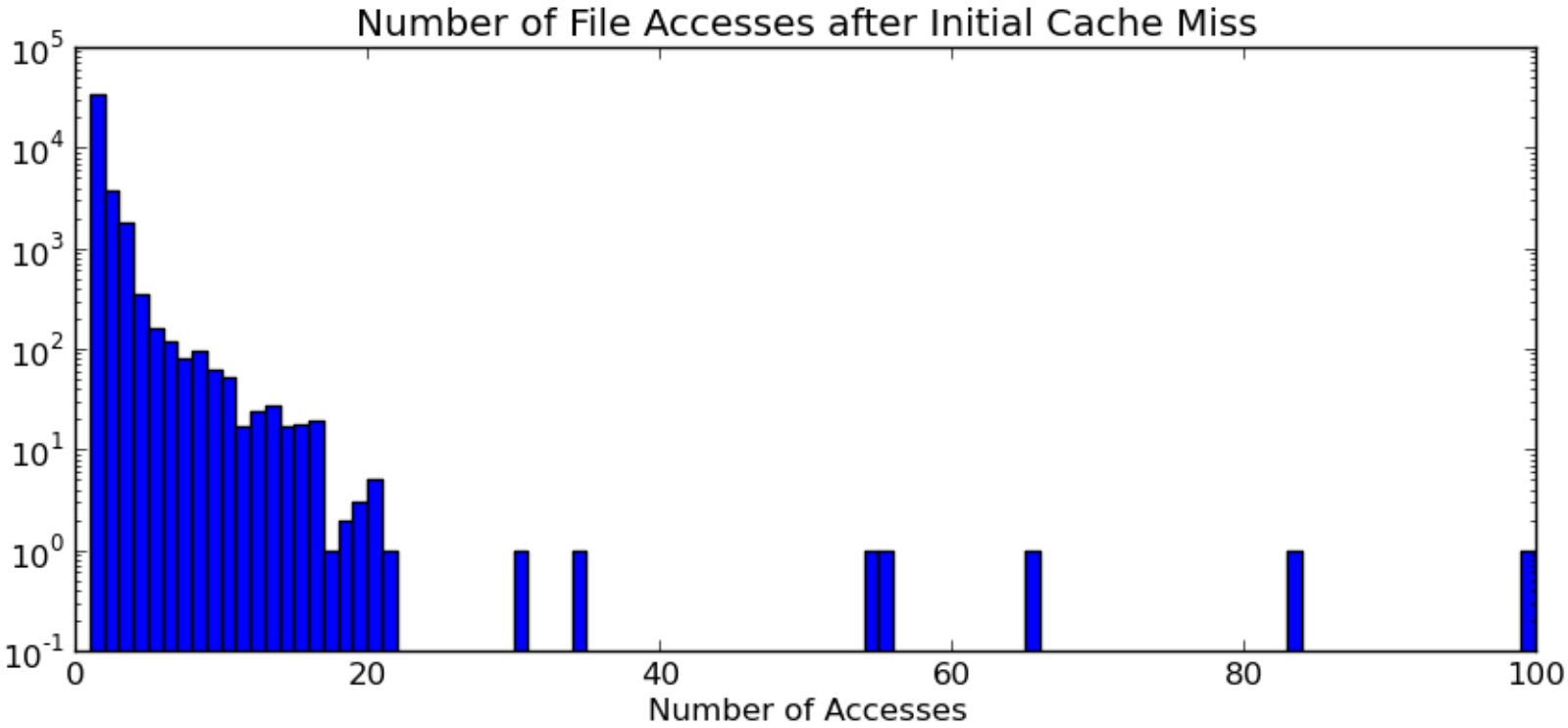
UNIVERSITY OF BIRMINGHAM

Network Access



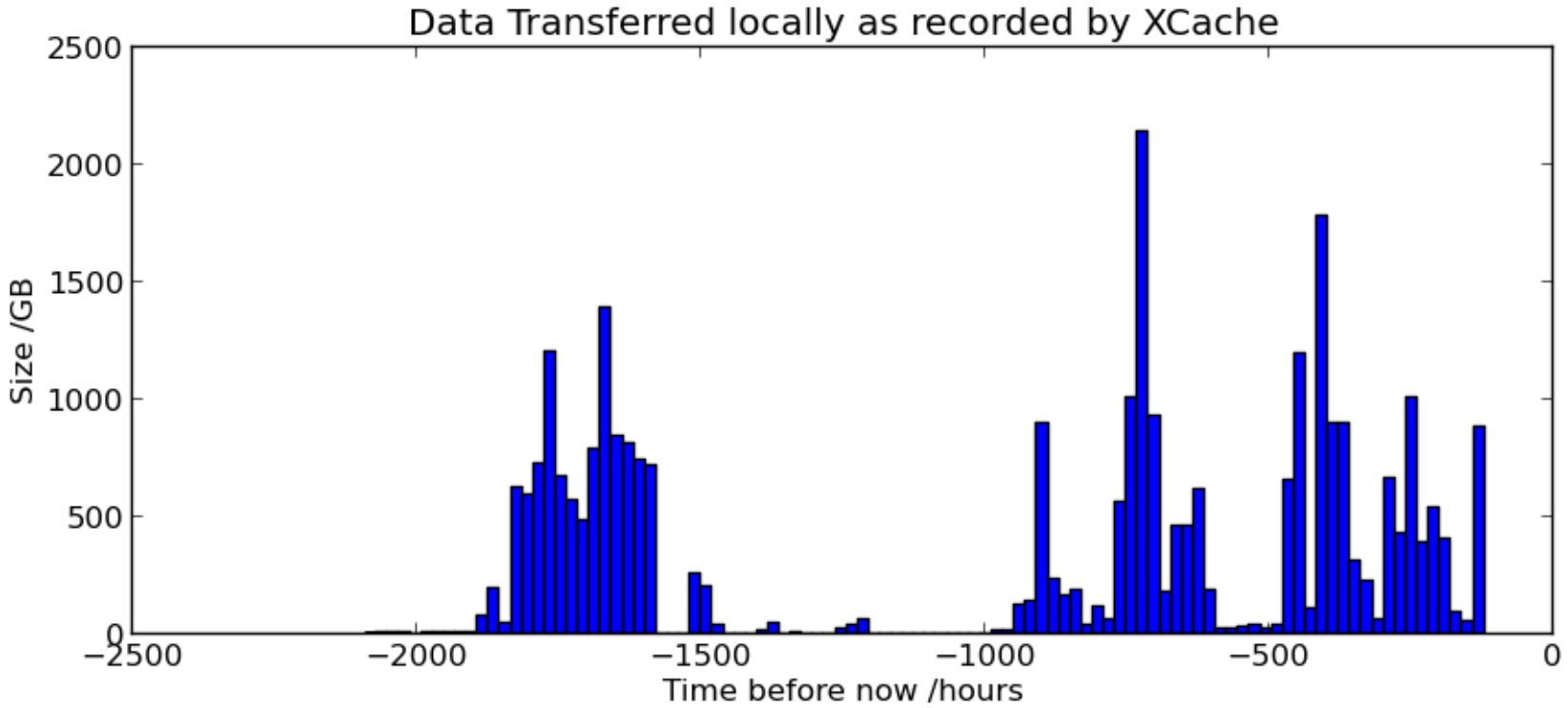
Cache Metrics

Number of file accesses (over ALL time) after initial cache miss



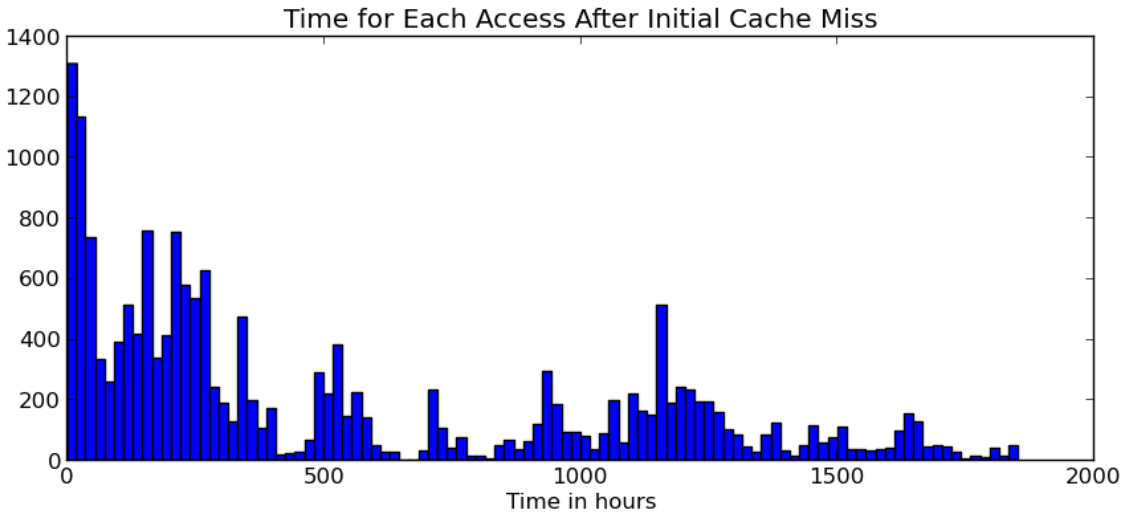
Cache Metrics

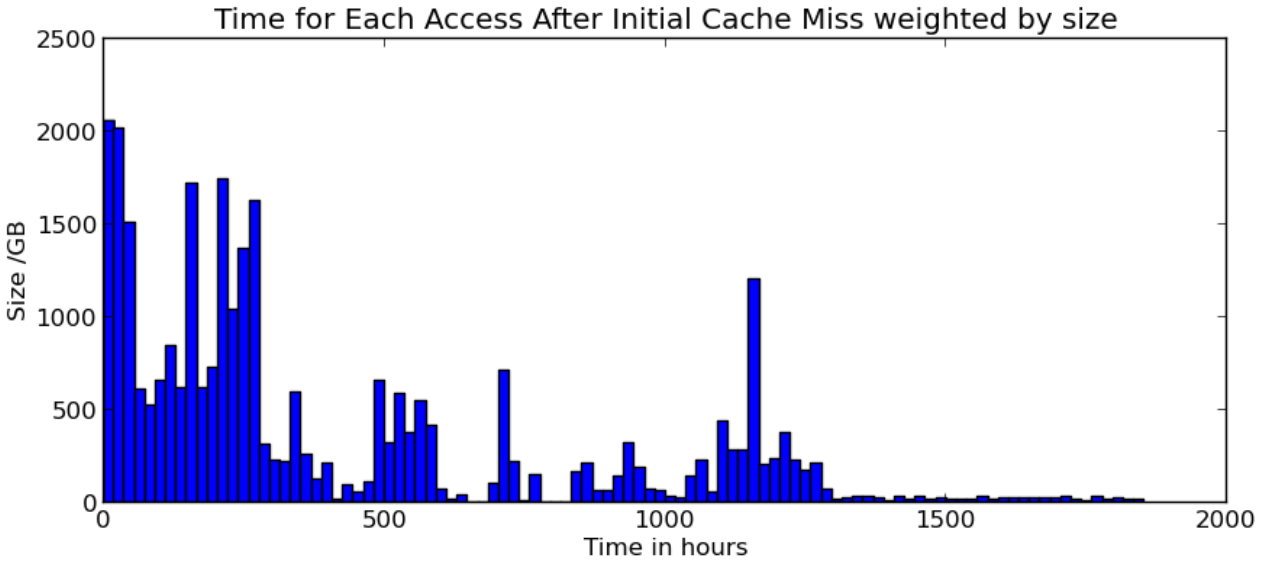
Data transferred from the local nodes as recorded by XCache



Cache Metrics

Time for each access after initial cache miss





UNIVERSITY OF BIRMINGHAM

Problems and Thoughts

Overall, my experience with Xcache has been very positive:

- The caching mechanism seems to work as intended
- New files are cached for arriving jobs then access ~3-5 times soon after
- During production, disks were filled at ~ 2.5TB/day
- From network graphs, it seems there is a some (significant?) saving on external traffic

This system has been pretty easy to setup and run from my point of view – I gather there have been issues on the Panda/AGIS side of things though…

There have only been two significant issues I can think of that weren't associated with my own expertise shortfall:

- Recent update to xrootd-rucioN2N-for-Xcache doesn't work with xrootd from UMD
- I need to use my own certificate/proxy to download the data files from RUCIO

Overall though, it is a very easy way to fill a small amount of storage with useful files and therefore allow (almost) diskless sites to run without overloading the larger ones.

Plans for the Future

Going forward, I have a few things planned:

- Get Teng's monitoring working
- Add more storage if needed
- If required, do some more analysis of the hits rates, file use, etc.

I also believe Atlas have more plans as well, principally taking advantage of RUCIO being 'Xcache-aware' and 'warming up' a cache in preparation for jobs