SuperMUC CVMFS usage

- Rod Walker, LMU, 29th Jan 2018

Motivation and disclaimer

- Goal is to run ATLAS MC production opportunistically on Munich HPC
  - with all the very hard constraints
- I am not a computer scientist
  - using technologies I hear about in HEP world
  - not necessarily the best or most performant solution

SuperMUC

- HPC at LRZ in Munich
  - installed Phase I 2012 150k cores
  - with Phase II 240k cores, 6.8PFlop/s
    - was #4 in top500
  - pure cpu based, Intel, Linux, 2GB RAM/core
- No local disk, shared GPFS
- No outside network, in or out
  - ssh into headnode
  - outbound from headnode
- SLES11 - not SL6
- Tuned for HPC
  - no customization of nodes

ATLAS SW access

- Traditional cvmfs on WN ruled out
  - no IP, no disk, admins refuse to install anything
  - many workarounds ruled out too
- Fallback solution
  - rsync of selected /cvmfs directories to shared GPFS: stable for long sim campaign
  - relocation hacks: /cvmfs to /gpfs/… in many scripts, envs, PFC
    - meanwhile relocation formalized in ALRB
  - Problem was performance and scalability on sharedFS
    - big reason we moved cvmfs on grid

SharedFS performance

- Setup and athena initialization
  - touches lots of files and loading many libraries
  - 4mins increased to 20mins
    - bad but bearable for 2-4hrs job
- Event loop slowdown: 35s to 115s per event
  - Running with strace see file access times are responsible
    - Geant4 loads detector data files lazily
    - each of 24 AthenaMP processes reads(and caches) a few hundred files
  - GPFS client caches 1000 inodes (maxFilesToCache 1000)
    - recommendation is 4000 but LRZ refuse to change
    - exact same problem on MPI HPC (Hydra) fixed with this config change
- Factor 3 slowdown! Need a new SW access method

Problems and solutions

- GPFS metadata is the problem, not the io
  - CVMFS client caches metadata
- No changes to nodes, no fuse
  - User-level cvmfs possible using ccools parrot
    - uses ptrace to intercept FS operations
- No outbound IP to proxy
  - pre-loaded ALIEN cache on GPFS
    - everything is cached so avoids need for connectivity

One last complication

- SL6 ATLAS SW not fully-compatible with SLES11
  - Openssl lib links and some other tricks make most things work (eg. on LRZ T2)
  - still get segfault on SuperMUC
- could run in chroot environment
  - no chroot! User-level equivalent is proot - also a ptrace application.
- Parrot can change root dir too

A working command line

exec cctools/bin/parrot_run -m pmounts.lis -l proot/slc6/lib64/ld-linux-x86-64.so.2 $0

- Parrot with
  - chroot for SL6
  - cvmfs lib
    - Pre-filled ALIEN cache

```
/ /home/hpc/pr58be/ri32bz2/software/proot/slc6
/home/hpc/pr58be/ri32bz2/software/proot/slc6/lib64
/home/hpc/pr58be/ri32bz2/software/proot/slc6/lib64
/dev /dev
/misc /misc
/net /net
/proc /proc
/sys /sys
/var /var
/tmp /tmp
/selinux /selinux
/etc/hosts /etc/hosts
/etc/resolv.conf /etc/resolv.conf
/etc/group /etc/group
/home /home
/cvmfs /cvmfs
/gss /gss
/gpfs /gpfs
```

Does it help?

- Startup 17min cf. 20min for GPFS SW
- Event loop timing back to ‘good’ GPFS value
  - ptrace overhead offset by metadata caching of cvmfs
    - first time anything ran faster inside parrot!
  - contrived way to fix bad gpfs client config
- Running ATLAS production like this
  - 300 whole-nodes with 16 cores each
    - bigger than LMU T2, and free (~10% FTE)
  - preemptable to let proper HPC workloads run
    - event level checkpointing means we lose very little work (EventService)
  - limit is LRZ policy not scalability(yet)

Can we do better?

- Heard from CSCS work about cvmfs tiered cache
  - using some RAM as cache despite no local disk
- Looks nice in pure cmvfs but ….
  - pre-filled ALIEN cache, parrot_run, sl6 chroot, SLES11
- Corner case finds bugs and features
  - ALIEN cache, workaround : cp .../preload/cvmfschecksum.atlas.cern.ch .
  - Cvmfs version built into parrot too old
    - only got working version last week, so not much experience so far
- Initial test shows no difference (during meeting!)

Open issues

- ALIEN pre-loaded cache
  - Cvmfs_preload with dirtab sometimes scrambles cache
    - leads to missing directories
  - no way to delete from the pre-loaded cache to save disk space
    - seems difficult, so can live with creating a new slim cache
- Still uses many small files on GPFS
  - SquashFS of subset of /cvmfs, loopback mount
  - GPFS serves single big file. Application sees FS
  - Needs new parrot functionality, if possible at all.
- Start athena from checkpoint image, made after initialization
  - Proof of principle done - long road to production
- Maybe other and better solutions.

Future

- Next generation SuperMUC
  - contracts signed for 26PFlop/s, all Intel cpu, Lenovo machine.
  - planned for October 2018, but expect long burn in.
- Should be more options(IF we get access)
  - Singularity support
    - no need for chroot to get SL6
    - FAT image with SW inside
  - maybe we can get outbound http via proxy

Acknowledgments

Impressive this corner-case is covered at all, by cvmfs and parrot.

Very good and patient support for poorly reported bugs and features.

Thanks to Jakob, Doug Thain & cvmfs/cctools teams