Day-to-day HTCondor Operations at RAL

European HTCondor Workshop
RAL Sept 2018

John Kelly

Who am I?
What do I do?

John Kelly

European HTCondor Workshop 2018

Production Team
Doing day to day tasks

Including fixing worker nodes.

History

Original batch system was PBS

Assessed other systems.

Decided to use HTCondor

Details can be found at

https://indico.cern.ch/event/272785/contributions/1612799/attachments/490393/677896/PREGDB_HTCondor_RAL_v3.pdf

'Original' condor setup

- HA Central managers.

- Four Schedds running on Arc-ce.

- About 600 worker nodes running SL6.

- Using a 'conventional' setup with jobs running directly on the WNs and using cgroups to manage resources.

- Some experimenting with bursting into the RAL cloud.

- All set up by A.L.

Science & Technology
Facilities Council

UK Research
and Innovation

The big changes

Enter SL7, Docker, Echo/Ceph.

- All WNs running SL7

- Used JOB_TRANSFORM to convert (almost) all jobs to docker jobs.

- Used wrapper script on WNs to start start the Docker image with custom settings.

- Jobs can have SL6 or SL7 containers.

- Docker images to provide access to Echo/Ceph.

JOB_TRANSFORM

```
# Convert job to Docker universe
JOB_TRANSFORM_NAMES = $(JOB_TRANSFORM_NAMES), DefaultDocker
JOB_TRANSFORM_DefaultDocker @=end
[
  Requirements = JobUniverse == 5 && DockerImage =?= undefined && Owner =!= "nagios";
  set_WantDocker = true;
  eval_set_DockerImage = ifThenElse(NordugridQueue =?= "EL7", "stfc/grid-workernode-c7:2018-
07-09.2", "stfc/grid-workernode-c6:2018-07-09.2");
  set_Requirements = ( TARGET.HasDocker ) && ( TARGET.Disk >= RequestDisk ) && (
TARGET.Memory >= RequestMemory ) && ( TARGET.Cpus >= RequestCpus ) && (
TARGET.HasFileTransfer ) && ( x509UserProxyVOName =?= "atlas" && NumJobStarts == 0 ||
x509UserProxyVOName =!= "atlas");
  copy_TransferInput = "OriginalTransferInput";
  eval_set_TransferInput = strcat(OriginalTransferInput, ",", Cmd);
]
@end
```

Worker node with Docker
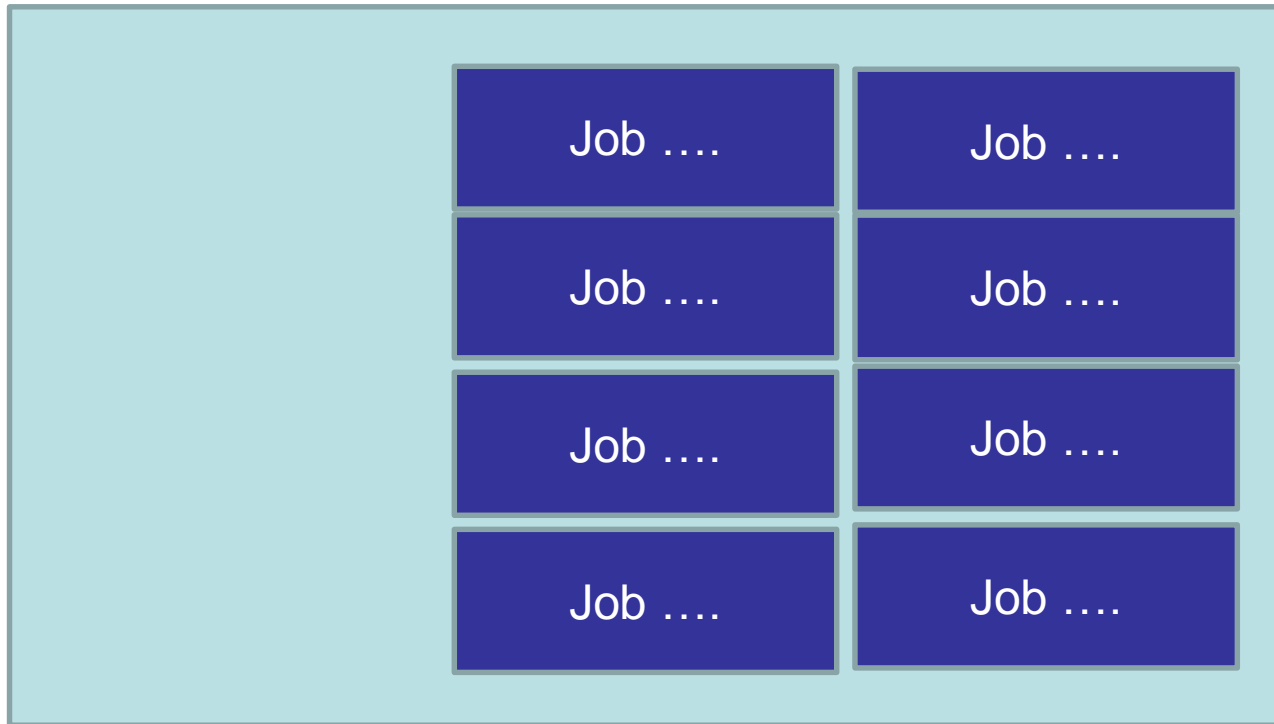
Using xrood docker containers to talk to ECHO

- The world talks to ECHO via 'gateways' , xrootd.echo.stfc.ac.uk

  gridftp.echo.stfc.ac.uk

- Worker nodes talk to ECHO via local docker gateways.

  – In the hosts file in a docker container there is :

  – 172.28.1.1   xrootd.echo.stfc.ac.uk

  – So WNs talk to their local docker gateway, while the rest of the world talk to the dedicated gateways.

Science & Technology
Facilities Council

UK Research
and Innovation

Worker node with xrootd
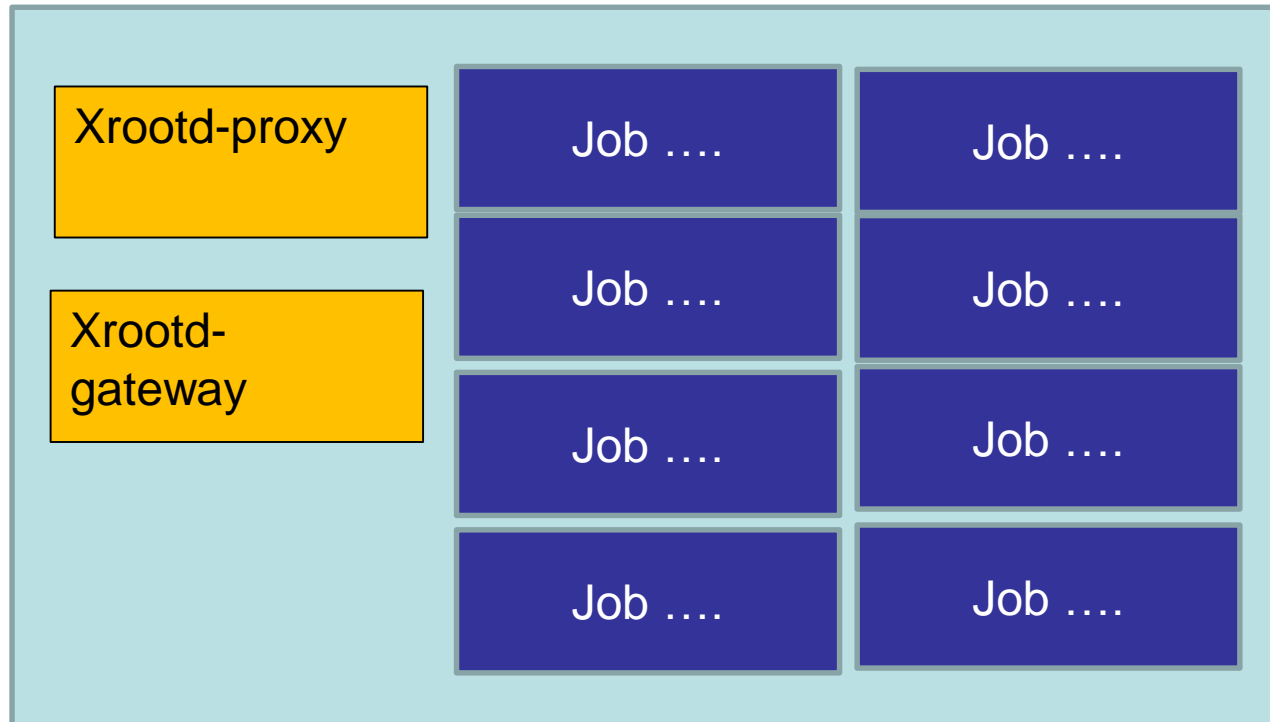
Recent management changes

Condor has been pretty much unloved since AL left.

James Adams has recently been appointed batch farm manager
We should see a more focused approach

Recent Problems

- Some nodes still running stock kernel–Fixed — Nodes re-installed

- Typo in mainline kernel script–Fixed

- Some file-systems created with incorrect flags–Fixed — Nodes re-installed

- Some nodes running older Docker version–Fixed — Upgraded

- Network configuration fighting authconfig over NIS–Fixed — Removed NIS

- Attempting to configure incorrect/non-existent network interfaces–Fixed — Configuration now renames interfaces (update reality to match config)

- Efficient reboot script cron being called without args–Fixed

- Docker using 172.17.0.0/16 — routed on campus–Mitigated

- Telegraf version too old for mainline kernel–Fixed

- Firmware bugs in half of 2015 nodes causing instability/hourly reboots–Ongoing, fixed code must be deployed manually

James Adams
gridpp41

Science & Technology
Facilities Council

UK Research
and Innovation

Condor issues

There are surprisingly few

- Lots of custom scripts.
- Lots of stuff we don't understand
- Lots of STFC specials.

healhcheck_wn_condor

It runs as a STARTD_CRON_JOBLIST

And originally returned:

- NODE_IS_HEALTHY = True
- NODE_STATUS = "All_OK"

And now it also returns:

- ECHO_XROOTD_GATEWAY_STATUS = "Healthy"
- ECHO_XROOTD_PROXY_STATUS = "Healthy"

The script output is part of the start expression.

It also sends a packet to nagios for a healthcheck test

It needs a review and maybe a rewrite.

Science & Technology
Facilities Council

UK Research
and Innovation

Docker.py

Wrapper to start docker containers.

Special conditionals to start singularity containers within docker containers and to set things up eg. Networking and permissions.

It may be possible to move this functionality into Condor itself.

efficientdefrag.py

Used to defrag worker nodes so as to allow multicore jobs to run.

I am not sure it works as advertised.

It needs rewriting or the functionality can be moved into Condor itself

Also there is condor_efficient_reboot.sh

Condor issues

- Lots of custom scripts.
- Lots of stuff we don't understand – this work shop should help
- Lots of STFC specials and maybe redundant configs.

Future Plans

- Period of stability
  - If nothing else, prove that everything is working
- Full automation of routine operations
  - Upgrades and rolling-reboots
- Cloud bursting
  - We are working on this again
- Generally get more comfortable with Condor
  - This workshop will help

Thank you.
??

Science & Technology
Facilities Council

UK Research
and Innovation

Docker command line

- /usr/bin/**docker** run -eSINGULARITY_BINDPATH=/etc/hosts --cap-add=SYS_ADMIN --cap-add=DAC_OVERRIDE --cap-add=SETUID --cap-add=SETGID --cap-add=SYS_CHROOT --cpu-shares=800 --memory=26000m --memory-reservation=13000m --network=ralworker --add-host=xrootd.echo.stfc.ac.uk:172.28.1.1 --label=xrootd-local-gateway=true --cap-drop=all --hostname patls036-27044010.0-lcg2171.gridpp.rl.ac.uk --name HTCJob27044010_0_slot1_3_PID39564 -e PATH=/sbin:/usr/sbin:/bin:/usr/bin -e LD_LIBRARY_PATH=/opt/xrootd/lib -e PREVLEVEL=N -e runlevel=3 -e _CONDOR_SLOT=slot1_3 -e LANG=en_GB -e OMP_NUM_THREADS=8 -e RUNTIME_ENABLE_MULTICORE_SCRATCH=1 -e ARC_LOCATION=/usr -e previous=N -e _CONDOR_CHIRP_CONFIG=/pool/condor/dir_39564/.chirp.config -e TMPDIR=/pool/condor/dir_39564 -e GLOBUS_LOCATION=/usr -e _CONDOR_SCRATCH_DIR=/pool/condor/dir_39564 -e UPSTART_EVENTS=runlevel -e UPSTART_INSTANCE= -e TEMP=/pool/condor/dir_39564 -e _CONDOR_JOB_IWD=/pool/condor/dir_39564 -e _CHIRP_DELAYED_UPDATE_PREFIX=Chirp* -e RUNTIME_CONFIG_DIR=/etc/arc/runtime -e GLOBUS_TCP_PORT_RANGE=50000,52000 -e X509_CERT_DIR=/etc/grid-security/certificates -e OPENSSL_ALLOW_PROXY_CERTS=1 -e _CONDOR_JOB_PIDS= -e ATLAS_CGROUPS_BASE=/cgroup/memory -e PWD=/var/spool/arc/grid08/EKAMDm3liGtnCIXDjqiBL5XqABFKDmABFKDm4nESDmABFKDmdeu3en -e _=/usr/bin/condor_submit -e LANGSH_SOURCED=1 -e X509_USER_PROXY=/pool/condor/dir_39564/job.EKAMDm3liGtnCIXDjqiBL5XqABFKDmABFKDm4nESDmABFKDmdeu3en.proxy -e CONDOR_CONFIG=/etc/condor/condor_config -e TMP=/pool/condor/dir_39564 -e BATCH_SYSTEM=HTCondor -e VO_ATLAS_SW_DIR=/cvmfs/atlas.cern.ch/repo/sw -e _CONDOR_JOB_AD=/pool/condor/dir_39564/.job.ad -e GRIDMAP=/etc/grid-security/local-grid-mapfile-ral -e X509_USER_KEY=fake -e UPSTART_JOB=rc -e RUNLEVEL=3 -e SHLVL=3 -e CONSOLETYPE=vt -e CONDOR_BIN_PATH=/usr/bin -e TERM=linux -e _CONDOR_MACHINE_AD=/pool/condor/dir_39564/.machine.ad -e OLDPWD=/ -e X509_USER_CERT=fake --volume /pool/condor/dir_39564:/pool/condor/dir_39564 --volume /etc/grid-security:/etc/grid-security:ro --volume /etc/machinefeatures:/etc/machinefeatures:ro --volume /etc/profile.d/grid-env.sh:/etc/profile.d/grid-env.sh:ro --volume /etc/glexec.conf:/etc/glexec.conf:ro --volume /etc/lcmaps:/etc/lcmaps:ro --volume /etc/lcas:/etc/lcas:ro --volume /etc/passwd:/etc/passwd:ro --volume /etc/group:/etc/group:ro --volume /cvmfs:/cvmfs:shared --volume /etc/cvmfs:/etc/cvmfs:ro --volume /sys/fs/cgroup:/sys/fs/cgroup:ro --volume /pool/atlas/recovery:/pool/atlas/recovery --volume /etc/atlas:/etc/atlas:ro --volume /etc/arc:/etc/arc:ro --workdir /pool/condor/dir_39564 --user 36897:24331 stfc/grid-workernode-c6:2018-07-09.2 ./condor_exec.exe

Recent Problems 1

Standard RH kernel does not work well with Docker containers.

Fragmented memory issues.

Fixed by going to the mainline kernel.

docker: Error response from daemon: oci runtime error: container_linux.go:247: starting container process caused "process_linux.go:245: running exec setns process for init caused \"exit status 29\"".

Recent Problems 2

- CMS are running jobs in singularity containers.

- These jobs didn't use the xrootd-gateway / xrootd-proxy docker images.

- Passed parameters to Docker so that the singularity container was started with correct environment.

- See GGUS https://ggus.eu/?mode=ticket_info&ticket_id=135822

Recent Problems 3

- Docker needs an XFS filesystem with d_type set.

- Docker 17.09 sort of works without this, but fails in mysterious ways.

- docker 18.03 does not work.

- So many nodes had to be re-installed.