
CERN

CERN HTCondor Migration

Ben Jones

Batch Service at CERN

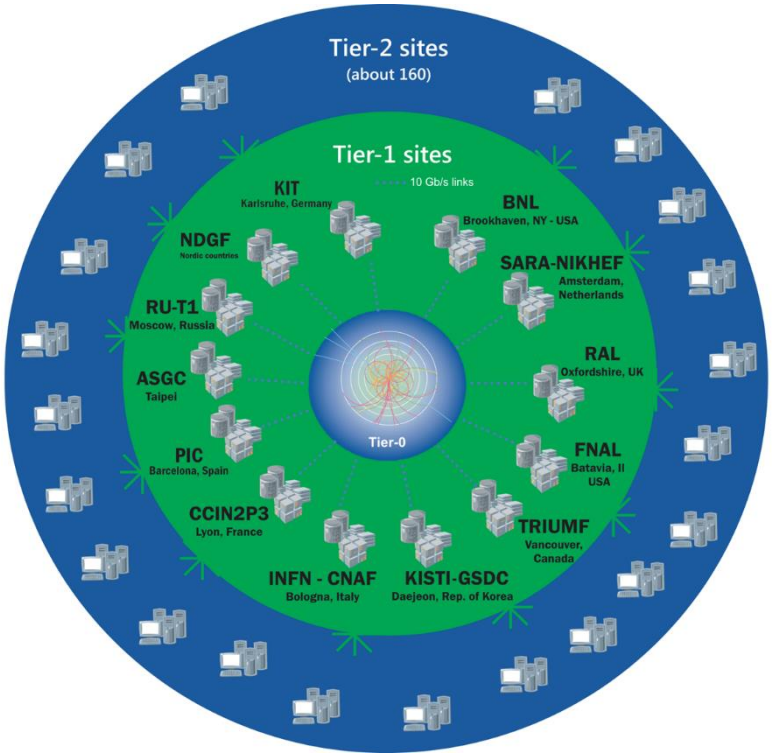
- CERN Batch system to process CPU intensive workload ensuring fairshare among various user groups
- Maximize utilization, throughput, efficiency
- Split of Grid or “local” submissions
- ~~110k~~ 156k cores
  - Mostly VM
  - HTCondor: 8 core or 10 core VMs
  - 1.3 million jobs per day

Worldwide LHC Computing Grid

TIER-0 (CERN):
data recording, reconstruction and distribution

TIER-1:
permanent storage, re-processing, analysis

TIER-2:
Simulation, end-user analysis



nearly 170 sites, 40 countries

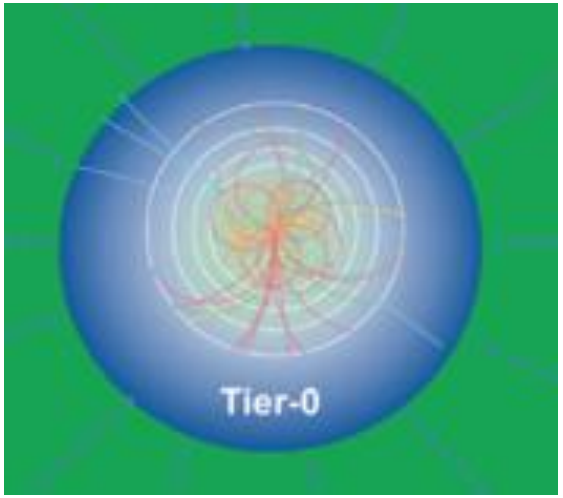
~350’000 cores

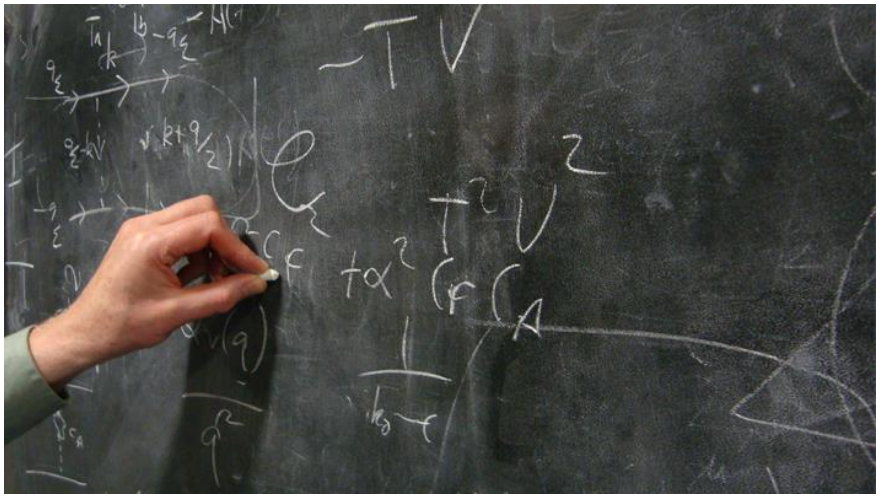
500 PB of storage

> 2 million jobs/day

10-100 Gb links

Not just the grid





LSF to HTCondor

- Proprietary vs Open
- Scale
  - LSF has 5K host limit
  - Can scale but only by splitting up instances
  - Central master for queries
  - Some divergence of feature set from “high throughput computing”
- HTCondor community
  - Great support from both HTCondor core team and others in WLCG
  - So far for us, CMS global pool pushing scale

LSF v HTCondor

- Until very recently, we've had LSF at max size, and grown HTCondor
- HTCondor pool passed LSF in size recently
- LSF two instances, "share" and ATLAS T0
  - Share: ~51k cores
  - ATLAS T0: ~17k non-HT cores
- HTCondor one (slightly partitioned) pool:
  - ~87k cores
- Big "local" customers (Theory, Beams) now 90/10 HTCondor / LSF
- Grid will move as soon as we build some more CEs

Grid

- Happy users of HTCondor-CE
  - Simplify middleware
  - Job router has helped manage opportunistic resources
- CEs: m2.xlarge with io1 spool
  - 8 cpus
  - ~15gb RAM
  - io1 CEPH volume (500 IOPS)
- We currently allow 10k running jobs per schedd + 4k idle

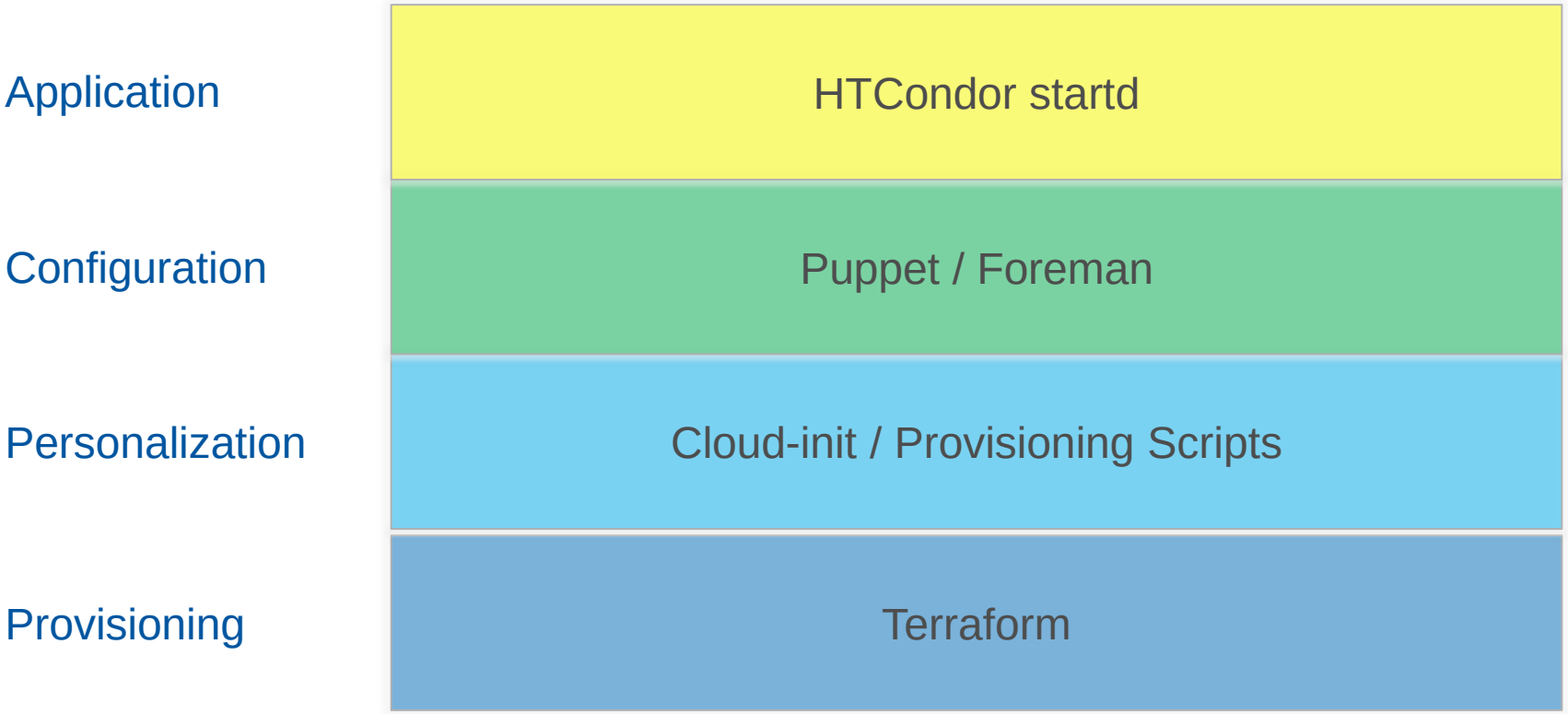
xBatch

- Seamless extension of resources available to HTCondor pool (nobody say hybrid)
- Certificates deployed whilst building cloud resources; execute nodes GSI authenticated
- Route so jobs could elect to run on cloud
- CCB for exec nodes to call back

```
[
  MaxJobs = 0;
  MaxIdleJobs = 0;
  TargetUniverse = 5;
  name = "External_Cloud";
  set_Requirements = (XBatch =?= True);
  set_WantExternalCloud = True;
  Requirements = (TARGET.WantExternalCloud =?= True) || (TARGET.queue =?= "externalcloud");
]
```

xBatch Stack

Job of each layer is just to bootstrap the next



BEER

- Batch on EOS (Evaluation of Resources?)
- Disk servers have lots of unused CPUs
  - Possible they don't have "unused" memory
- Can we use the CPUs?
- Condor Service in Cgroup
- To minimise configuration of "host", run jobs in Docker
  - Job Router transforms to Docker jobs
  - Host requires HTCondor, Docker, CVMFS and not much else

BEER isn't for everyone

- Use requirements on both the job and the startd to match jobs

```
BeerMachine = True
START = (StartJobs =?= True)  && (BeerJob =?= True) &&
(SendCredential =?= undefined || SendCredential =?= False)
```

```
[
 MaxJobs = 100;
 MaxIdleJobs = 10;
 TargetUniverse = 5;
 name = "Beer";
 set_Requirements = (BeerMachine =?= True);
 set_BeerJob = True;
 Requirements = (TARGET.WantBeer =?= True) || (TARGET.queue =?= "beer");
]
```

Volunteer Computing

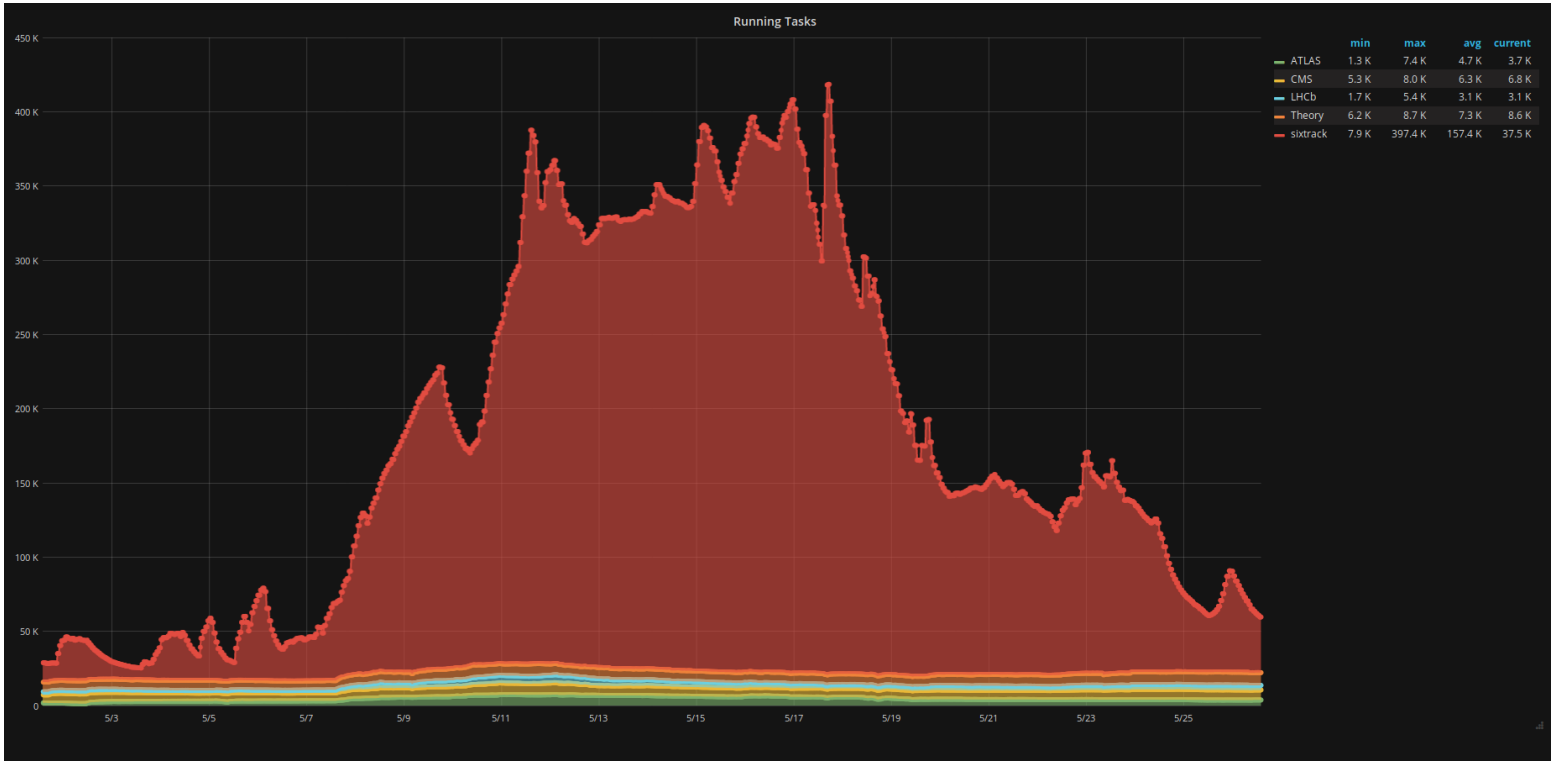
- A type of distributed computing
  - Computer owners donate computing capacity
    - Spare cycles on desktops but also idle machines in data centers
- Berkeley Open Infrastructure for Network Computing
  - Provides the middleware for volunteer computing
- Motivation
  - Free* resources
    - 100K + jobs slots for large established projects
  - Community engagement
    - Outreach channel
  - Participation
    - Offering people a chance to contribute

* Exploitation results in operations and maintenance costs

HTCondor Specifics

- The instant Glidein
  - HTCondor as a short-term lease on a resource
- Authentication using the Volunteer CA
  - GSI but a Low Level of Assurance
- HTCondor on residential networks
  - Satellite ISPs with > 1s latency
- HTCondor in the wild
  - Anyone, any system, any where, any state
- Suspend/Resume
  - Time Skip is normal and needs to be handed
- Accounting
  - Wall time is not equal to run time
    - Efficiency calculations
- HTCondor Submission to BOINC
  - For the classic case where pre- and post-processing requires the CERN batch service

8th BOINC Pentathlon 2017



”Local” users

- Many different use cases for local users of batch system
  - Shared filesystem and the submit file my grandfather gave me
  - Dedicated T0 resources with backfill from the grid
  - Complex workflows with dependencies (ie Theory)
  - Long running multicore (ie Beams)
  - Dedicated resources run as distinct partition (AMS)
  - Local T3 of LHC experiments

Mapping users to schedds

- Local schedds plan on ~15k running jobs\
- Need to map users to schedds

```
LOCAL_CONFIG_FILE = /usr/bin/cernbatchsubmit|
[bejones@aiadm26 ~]$ cernbatchsubmit
SCHEDD_HOST = bigbird05.cern.ch
CREDD_HOST = $(SCHEDD_HOST)
```

- Zookeeper with mapping to /htcondor/users/$username/current
- Plan for /htcondor/users/$username/previous for remapping

Job Flavours

- We want users to give us an idea of how long the job is
- MaxRuntime Ad set explicitly or via “flavour”
- Why?
  - Prioritise shorter jobs. Start expressions of some classes of machine, eg:
    - '(MaxRuntime < 432000)’
  - Machine draining (more later)
- Job Flavour mapped via Transform

Transforms

- We really like Job Transforms
- For Job Flavour:

```
CLASSAD_USER_MAPFILE_JobFlavours = \
/etc/condor/maps/jobflavours.mapdata
[…]
EVALSET MaxRuntime (userMap("JobFlavours", JobFlavour,
undefined) =!= undefined) ? int((userMap("JobFlavours",
JobFlavour, undefined))) : (MaxRuntime =?= undefined) ? 1200 :
MaxRuntime
```

- Other transforms to set defaults, or to route to “dedicated” resources based on Accounting Group, or to ensure we give out 2gb / cpu

Draining

- startd needs to be drained to upgrade
  - To be fair, just one of many reasons we drain
- We populate /etc/shutdowntime with timestamp on when node needs to be drained
- If present, sets startd Ad “InStagedDrain” to True and “ShutDownTime”
- START = (NodeIsHealthy =?= True) && (StartJobs =?= True) && ((InStagedDrain =?= True && (time() + MaxRuntime < ShutdownTime)) || InStagedDrain =?= False)

Other tools

- “Fifemon” from Fermi Lab adopted
  - Grafana / Graphite / HTCondor python bindings
  - https://batch-carbon.cern.ch/grafana/
- “Group Quota” from BNL
  - https://github.com/fubarwrangler/group-quota
  - Accounting Group management and delegation
  - Backronymed HAGGIS at CERN
  - REST service to automate dumping accounting groups and user maps to CM / schedds

Questions?

http://cern.ch/lhcathome