What's new in HTCondor? What's coming?

HEPiX Fall 2016

LBNL -- Oct 19, 2016

Todd Tannenbaum
Center for High Throughput Computing
Department of Computer Sciences
University of Wisconsin-Madison

CENTER FOR HIGH THROUGHPUT COMPUTING

HTCondor

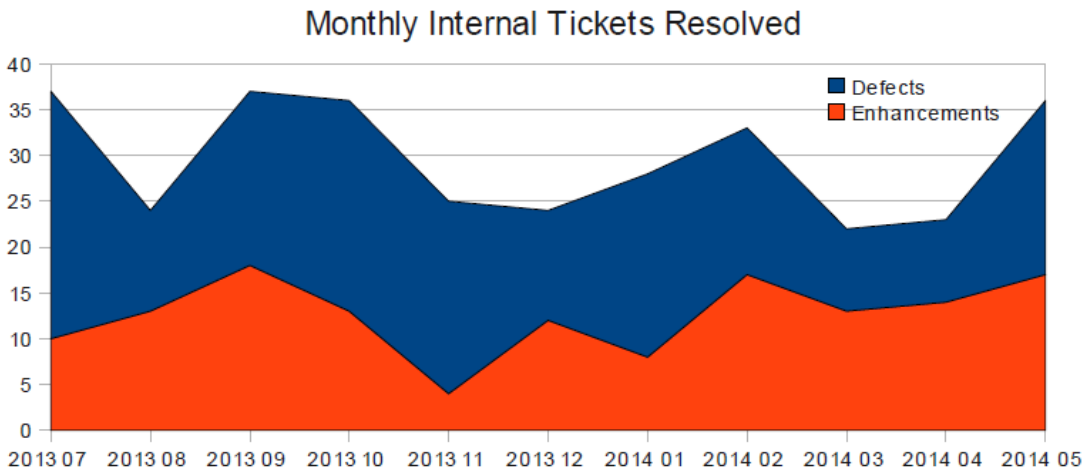
University of Wisconsin Center for High Throughput Computing

HTCondor

- Open source distributed high throughput computing
- Management of resources, jobs, and workflows
- Primary objective: assist the scientific community with their high throughput computing needs
- Mature technology…

Mature… but actively developed

- Since Fall HEPiX 2015 : 18 releases, 2157 commits by 20 contributors
- Open source development model
- Evolve to meet the needs of the science community in a ever-changing computing landscape

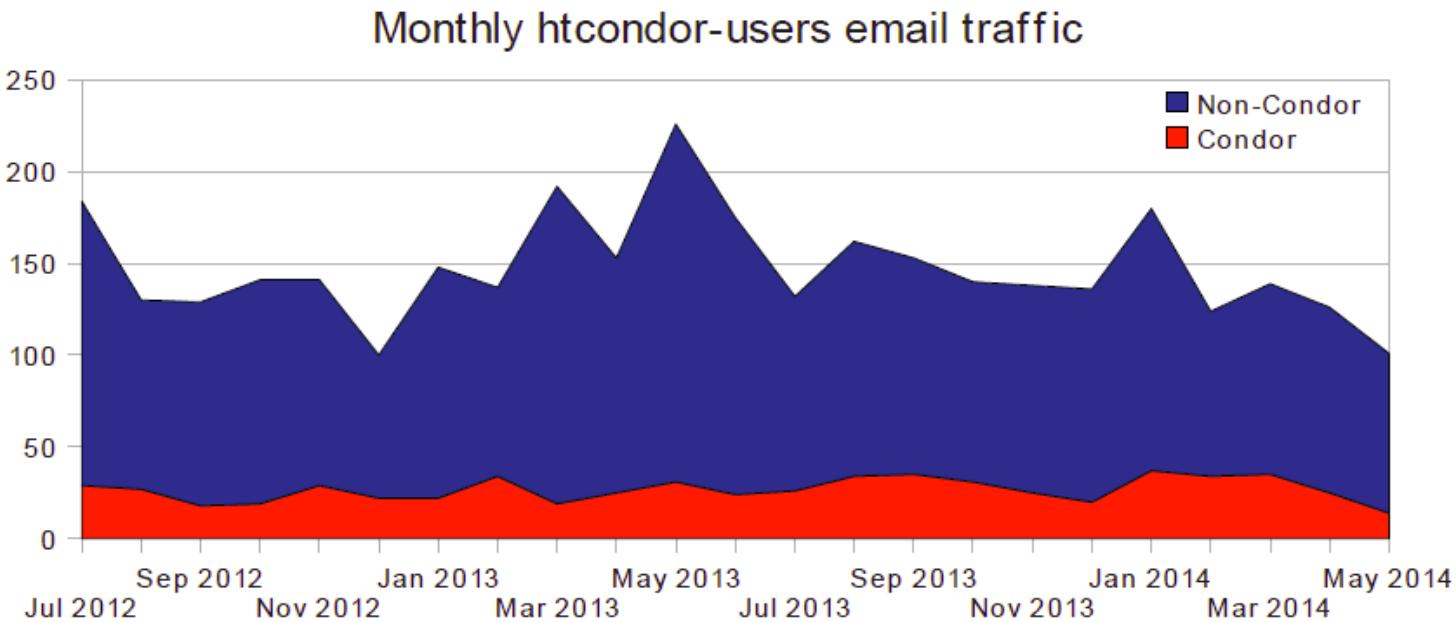


Why am I here?

- Desire to work together with the HEP community to leverage our collective experience / effort / know-how to offer an open source solution that meets the growing need of HEP high throughput computing in a challenging budget environment

Current Channels

- Documentation
- Community support email list (htcondor-users)
- Ticket-tracked developer support
- Bi-weekly/monthly phone conferences
  - Identify and track current problems
  - Communicate and plan future goals
  - Identify and collaborate on challenges, f2f
- Fully open development model
- Commercial options for 24/7



Meet w/ CMS, LIGO, IceCube, LSST, FNAL, OSG, …

HTCondor Week

- Annually each May in Madison, WI
- May 2-5, 2017

European Union HTCondor Workshops

- Previous workshops:
  - Fall 2015 @ CERN, Spring 2016 @ PIC
- "Mini-Workshop" @ CNAF next week!
  - See https://is.gd/B4jvnh
- Plan for 2017
  - Likely When: June 6-9, 2017
  - Where: Germany @ DESY in Hamburg, Germany

Release Timeline

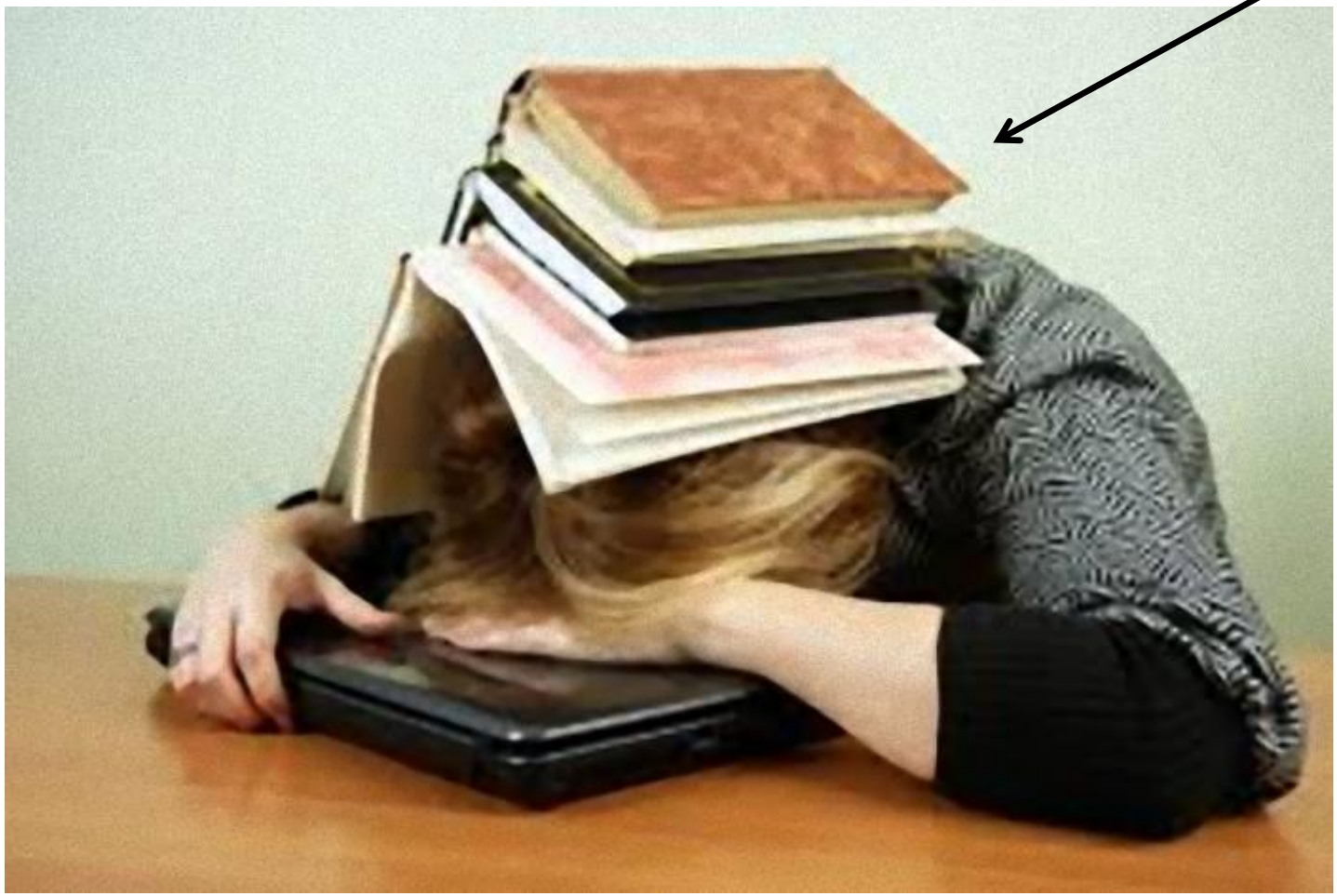
- Stable Series
  - HTCondor v8.4.x - introduced Sept 2015
  - Currently at v8.4.9
- "Development" Series v8.5.x (*should be 'new features' series')*
  - Currently at v8.5.7.
  - HTCondor v8.5.8 is the last release planned for the 8.5.x series; scheduled for code freeze on 2016-10-21, release on 2016-11-14.
- HTCondor v8.6.0 scheduled for public release on 2016-12-05.

See link "Forthcoming release plans" on http://htcondor.org

Enhancements in HTCondor v8.4 discussed last year

- Scalability and stability
  - Goal: 200k slots in one pool, 10 schedds managing 400k jobs
- Introduced Docker Job Universe
- IPv6 support
- Tool improvements, esp condor_submit
- Encrypted Job Execute Directory
- Periodic application-layer checkpoint support in Vanilla Universe
- Submit requirements
- New packaging

Page 790

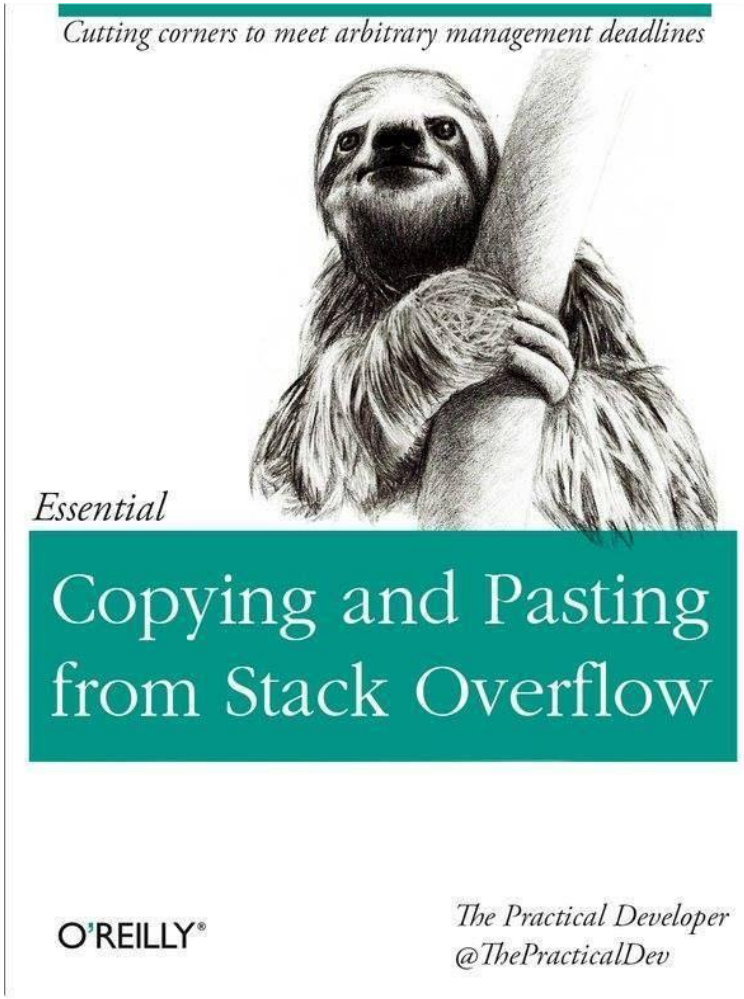


Enabled by default and/or easier to configure

- Enabled by default: IPv6 and IPv6 auto detection, shared port, cgroups
  - Have both IPv4 and v6? Prefer IPv4 for now
- Configured by default: Kernel tuning
- Easier to configure: More "meta-knobs", e.g.
  - `use policy: preempt_if_cpus_exceeded`
  - `use policy: hold_if_cpus_exceeded`

Seeking ideas to help users and admins learn

- Move HOWTO recipes on wiki to stackoverflow?
- Sub-reddit or forum instead of email list? Stack instead of IRC?
- YouTube videos?



New condor_q default output

- Only show jobs owned by the user
  - disable with `-allusers`
- Batched output (`-batch`, `-nobatch`)
- Proposed new default output of condor_q will show summary of current user's jobs.

```
-- Submitter: adam      Schedd: submit-3.chtc.wisc.edu
OWNER       IDLE RUNNING   HELD  SUBMITTED  BATCHNAME    JOBIDs
adam           -       1      -  3/22 07:20 DAG: 221546  230864.0
fred           -       -      1  3/23 08:57 AtlasAnlysis 263203.0
alice          -       1      -  3/27 09:37 matlab.exe   307333.0
mary         133      21      -  3/27 11:46 DAG: 311986  312342.0 ... 313304.0
```

New condor_status default output

- Only show one line of output per machine
- Can try now in v8.5.4+ with "-compact" option
- The "-compact" option will become the new default once we are happy with it

```
Machine       Platform      Slots Cpus Gpus  TotalGb FreCpu  FreeGb  CpuLoad ST

gpu-1         x64/SL6           8    8    2    15.57      0    0.44     1.90 Cb
gpu-2         x64/SL6           8    8    2    15.57      0    0.57     1.87 Cb
gpu-3         x64/SL6           8    8    4    47.13      0   16.13     0.85 Cb
matlab-build  x64/SL6           1   12         23.45     11   23.33     0.00 **
mem1          x64/SL6          32   80       1009.67      0  160.17     1.00 Cb
```

HTCondor and Kerberos

- HTCondor currently allows you to authenticate users and daemons using Kerberos
- However, it does NOT currently provide any mechanism to provide a Kerberos credential for the actual job to use on the execute slot

HTCondor and Kerberos/AFS

- So we are adding support to launch jobs with Kerberos tickets / AFS tokens
- Details
  - HTCondor 8.5.X to allows an opaque security credential to be obtained by condor_submit and stored securely alongside the queued job (in the condor_credd daemon)
  - This credential is then moved with the job to the execute machine
  - Before the job begins executing, the condor_starter invokes a call-out to do optional transformations on the credential

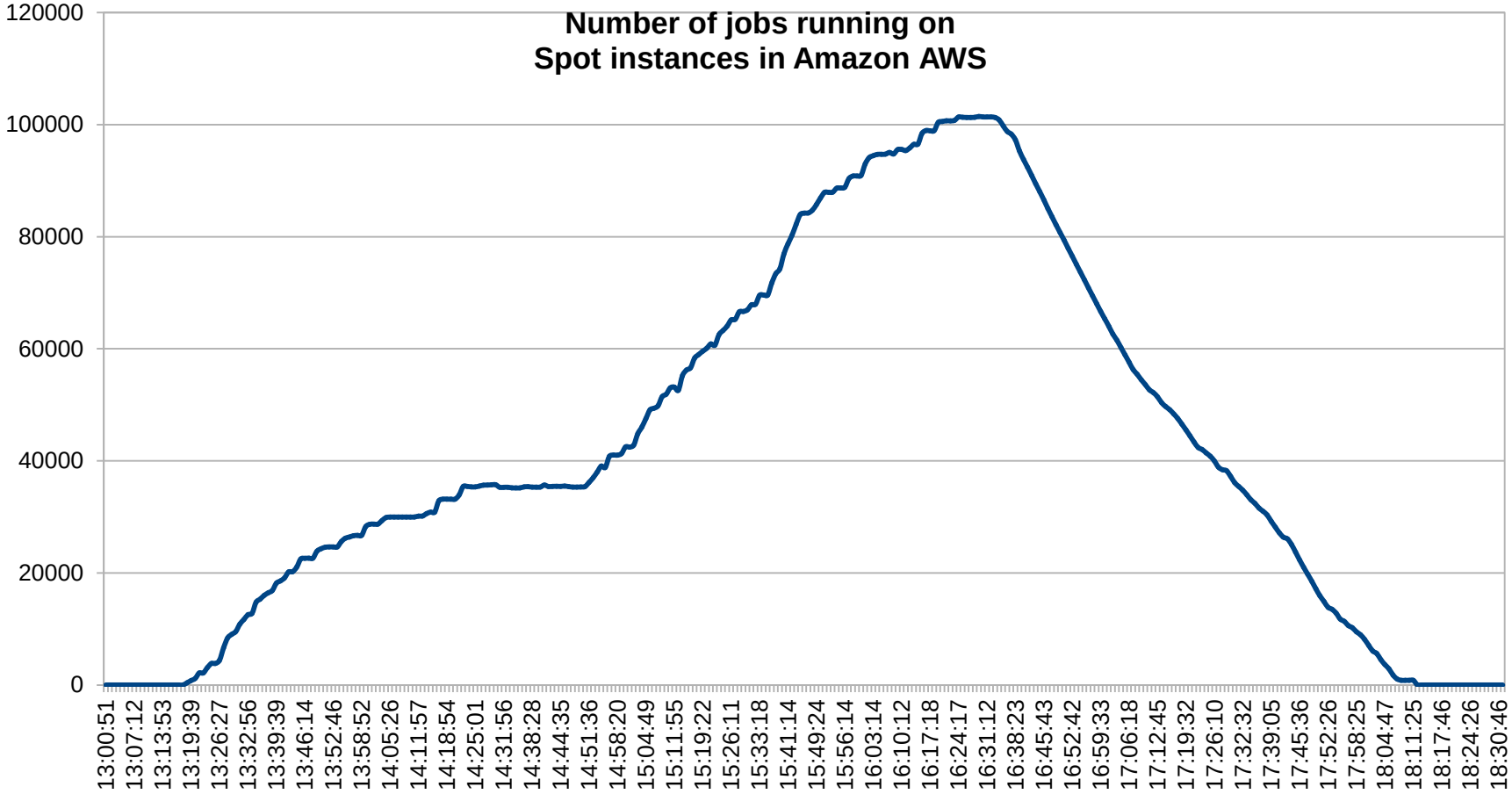
Grid Universe

- Reliable, durable submission of a job to a remote scheduler
- Popular way to send pilot jobs, key component of HTCondor-CE
- Supports many “back end” types:
  - HTCondor
  - PBS
  - LSF
  - Grid Engine
  - Google Compute Engine
  - Amazon EC2
  - OpenStack
  - Cream
  - NorduGrid ARC
  - BOINC
  - Globus: GT2, GT5
  - UNICORE

Add Grid Universe support for SLURM, OpenStack, Cobalt

- Speak native SLURM protocol
  - No need to install PBS compatibility package
- Speak OpenStack's NOVA protocol
  - No need for EC2 compatibility layer
- Speak to Cobalt Scheduler
  - Argonne Leadership Computing Facilities

Improved Scalability of Amazon EC2 grid jobs



Elastically grow your pool into the Cloud: *condor_annex*

- Start virtual machines as HTCondor execute nodes in public clouds that join your pool
- Leverage efficient AWS APIs such as Auto Scaling Groups and Spot Fleets
- Secure mechanism for cloud instances to join the HTCondor pool at home institution

Without `condor_annex`

+ Decide which type(s) of instances to use.

+ Pick a machine image, install HTCondor.

+ Configure HTCondor:

- to securely join the pool. (Coordinate with pool admin.)
- to shut down instance when not running a job (because of the long tail or a problem somewhere)

+ Decide on a bid for each instance type, according to its location (or pay more).

+ Configure the network and firewall at Amazon.

+ Implement a fail-safe in the form of a lease to make sure the pool does eventually shut itself off.

+ Automate response to being out-bid.

with condor_annex

› Goal: Simplified to a single command:

```
condor_annex --annex-id 'TheNeeds-MooreLab' \
     --expiry '2015-12-18 23:59' \
     --instances 1000
```

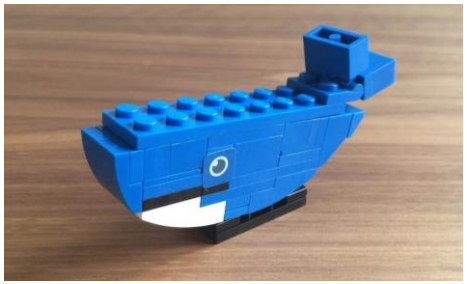
Transformation of job ad upon submit

- Allow admin to have the schedd add/edit/validate job attributes upon job submission in v8.5.7+

  (use case: insert trusted accounting group attributes based upon the user submitting)
- In v8.5.1+ can also set attributes as immutable by the user
  - Prevent user from editing protected attributes with condor_qedit or chirp

Docker Universe Enhancements

- Docker jobs get usage updates (i.e. network usage) reported in job classad
- Admin can add additional volumes
  - Why?
    - CVMFS
    - Large shared data



- DOCKER_DROP_ALL_CAPABILITIES config knob
  - Expression evaluated in context of job ad
  - If True, privilege escalation (like setuid binaries) in the container are prohibited

SELinux and systemd

- SELinux
  - (On by default in RHEL 7)
- Systemd Integration
  - Port Reservation - Systemd will reserve 9618 for HTCondor
  - Watchdog - If masters stops responding, systemd will restart it
  - Status messages - display via `systemctl status`
  - Logging - Daemon log messages can go to systemd-journald

DAGMan Improvements

- Splice Pin connections
  - Allows more flexible parent/child relationships between nodes in the workflow graph
  - Parsed when DAGMan starts up
- INCLUDE directive
- Set ClassAd attributes in DAG
- Set Batch Name

Scalability

- Starting campaign with CMS: scaling with 500k cores
- Scale the negotiation cycle
  - Fetch job request information from the schedds in parallel (in v8.5 already)
  - Parallelize Matchmaking Algorithm - scale by number of cores in central manager (in v8.5 already)
- Non-blocking authentication, smarter updates to the collector, faster ClassAd processing

Other improvements made in v8.5.x series

- Continued to enhance Python bindings
  - Bindings for draining, easier job submission
- New ClassAd functions, export ads as json, …
- See URL

http://htcondor.org/manual/v8.5/10_2Development_Release.html

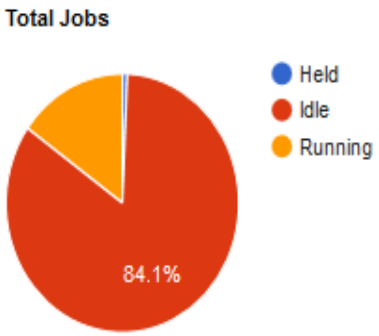
Data Management: Cache at worker node

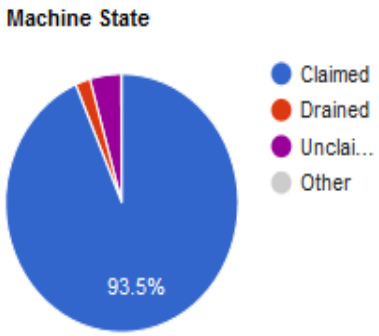
- Job scratch directory per *claim*, not just per job
- Two condor_cached implementations in R&D. See
  - http://htcondor.org/HTCondorWeek2015/presentations/MillerZ_HTCache.pptx
  - http://htcondor.org/HTCondorWeek2015/presentations/WeitzelD_CacheDPres.pdf
- Perform HTCondor file transfer of shared files via HTTP to enable reverse proxy (Squid) caching
  - http://htcondor.org/HTCondorWeek2015/presentations/VuosaloC_FileTransCachingProxy.pdf

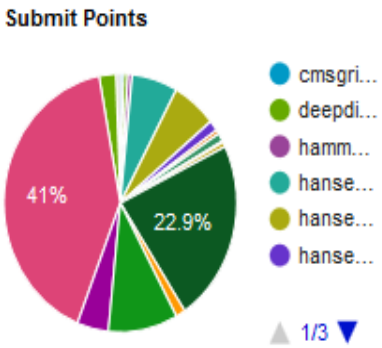
Monitoring improvements

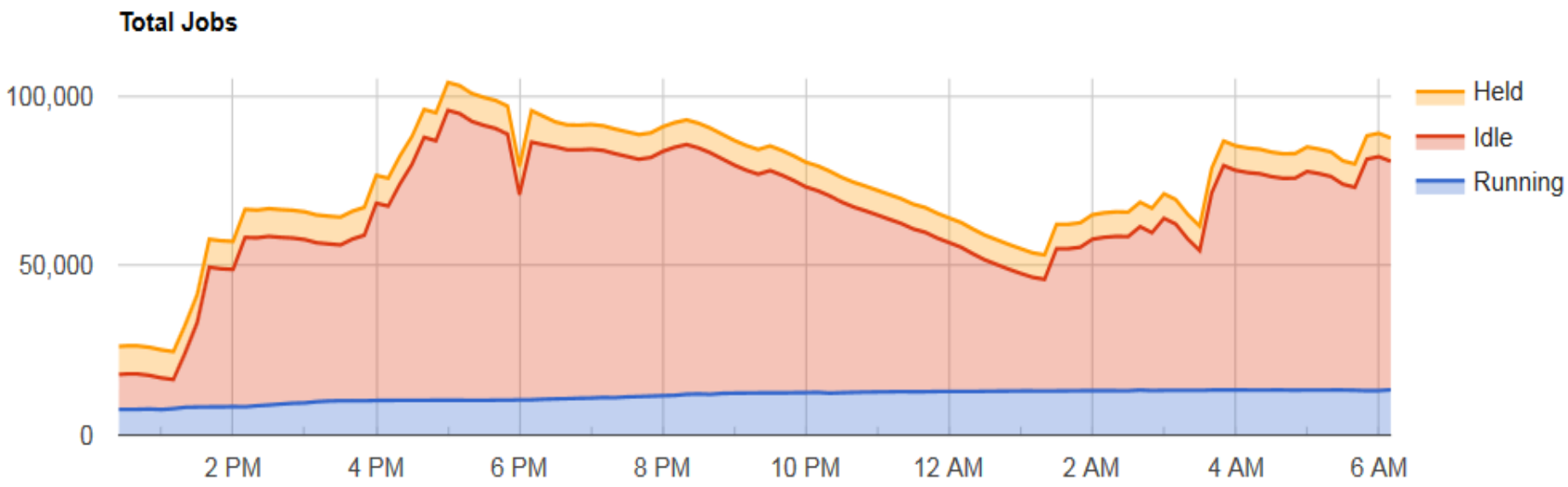
- Aggregate and send them to Ganglia!
  - condor_gangliad introduced in v8.2
  - See manual or my talk at http://bit.ly/1YBBO3P
- In addition to (or instead of) sending to Ganglia, aggregate and make available in JSON format over HTTP
  - condor_gangliad rename to condor_metricd
- View some basic historical usage out-of-the-box by pointing web browser at central manager (modern CondorView)…
- Or upload to influxdb, graphite for Grafana

HTCondor View









HTCondor Singularity Integration

- What is Singularity?

  http://singularity.lbl.gov/

  Like Docker but…
  - No root owned daemon process, just a setuid
  - No setuid required (post RHEL7)
  - Easy access to host resources incl GPU, network, file systems
- Sounds perfect for glideins/pilots!
  - Maybe no need for UID switching

Smarter and Faster Schedd

- User accounting information moved into ads in the Collector
  - Enable schedd to move claims across users?
- *Late materialization* of jobs in the schedd to enable submission of very large sets of jobs
  - More jobs materialized once number of idle jobs drops below a threshold (like DAGMan throttling)

Thank You!

P.S. Interested in working on HTCondor full time? Talk to me! We are hiring! https://is.gd/rAwNGT Email me or email htcondor-jobs@cs.wisc.edu