What’s new in HTCondor? What’s coming?

HTCondor Week 2017

Madison, WI -- May 3, 2017

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

HT CENTER FOR HIGH THROUGHPUT COMPUTING

HTCondor

Release Timeline

- Stable Series
  - HTCondor v8.6.x - introduced Jan 2017

    Currently at v8.6.2

    (Last year at v8.4.6)
- Development Series (*should be 'new features' series)*
  - HTCondor v8.7.x

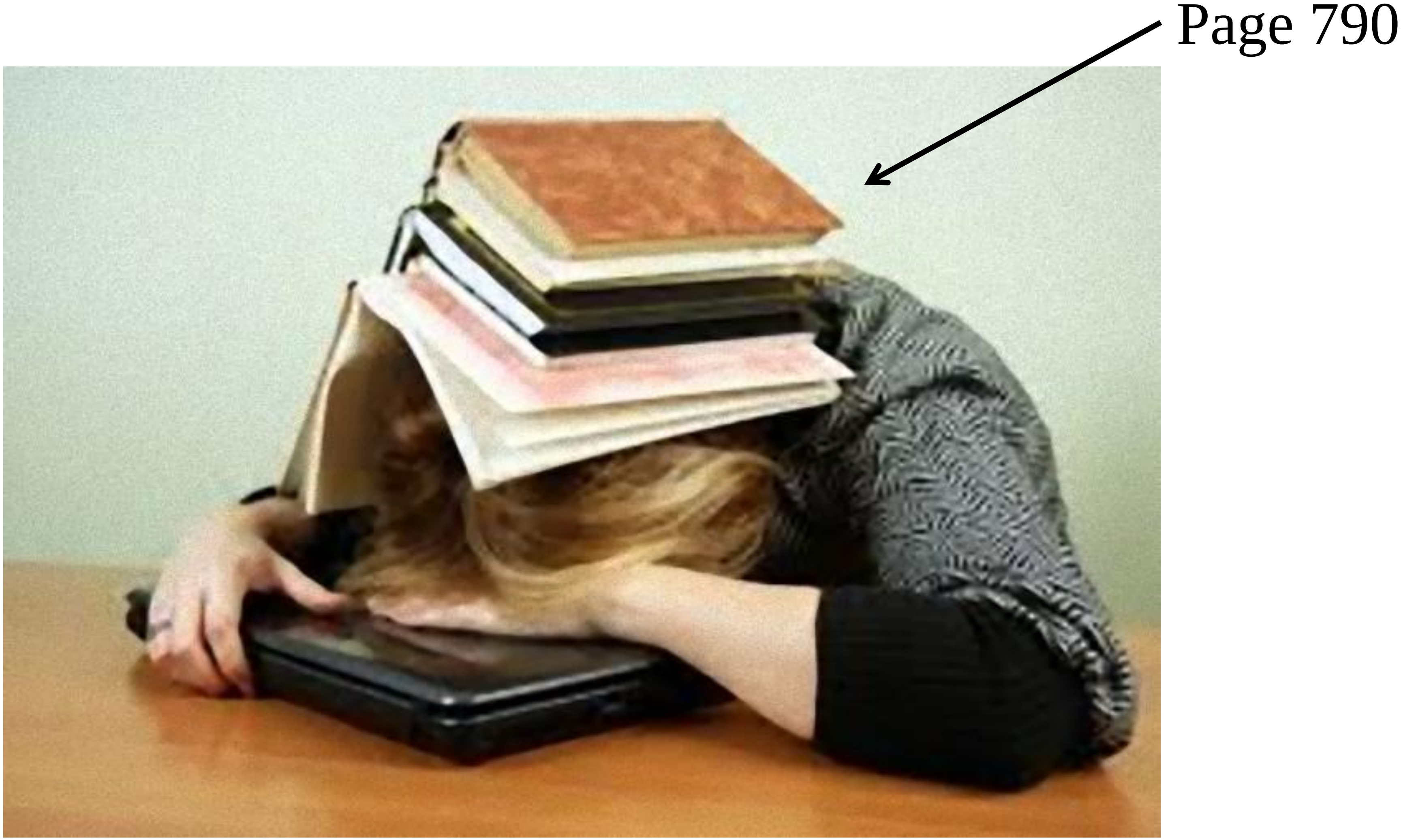
    Currently at v8.7.1

    (Last year at v8.5.4)

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 RPM / DEB packaging
- Systemd / SELinux compatibility

Some enhancements in HTCondor v8.6


Page 790

Enabled by default and/or easier to configure

- Enabled by default: shared port, cgroups, IPv6
  - Have both IPv4 and v6? Prefer IPv4 for now
- Configured by default: Kernel tuning
- Easier to configure: Enforce slot sizes
  - `use policy: preempt_if_cpus_exceeded`
  - `use policy: hold_if_cpus_exceeded`
  - `use policy: preempt_if_memory_exceeded`
  - `use policy: hold_if_memory_exceeded`

Easier to retry jobs if you shower

- Dew drinker? Use old way

```
executable = foo.exe
on_exit_remove = \
(ExitBySignal == False && \
 ExitCode == 0) || \
 NumJobStarts >= 3
queue
```

- Shower regularly? Use new way

```
executable = foo.exe
max_retries = 3
queue
```

New condor_q default output

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

```
---- Schedd: submit-3.batlab.org : <128.104.100.22:50004?... @ 05/02/17 11:19:41
OWNER    BATCH_NAME         SUBMITTED   DONE   RUN    IDLE   HOLD   TOTAL JOB_IDS
tannenba CMD: /bin/python  4/27 11:58    463    87   19450      5   20000 9.463-467
tannenba mydag.dag+10      4/27 19:13   9824     1       _      _    9825 10.0

29900 jobs; 10287 completed, 0 removed, 19450 idle, 88 running, 5 held, 0 suspended
```

Schedd Job Transforms

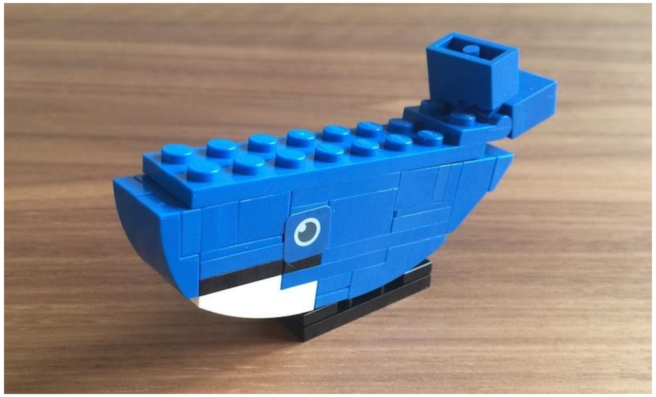
Transformation of job ad upon submit

- Allow admin to have the schedd securely add/edit/validate job attributes upon job submission
  - Can also set attributes as immutable by the user, e.g. cannot edit w/ condor_qedit or chirp
- Get rid of condor_submit wrapper scripts!
- *One use case: insert accounting group attributes based upon the submitter*

```
use feature: AssignAccountingGroup( filename )
```

Docker Universe Enhancements

- Docker jobs get usage updates (i.e. network usage) reported in job classad
- Admin can add additional volumes
  - That all docker universe jobs get
  - Why?
    - Large shared data
- Condor Chirp support



Also new knob:

- DOCKER_DROP_ALL_CAPABILITIES

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

And lots more…

- JSON output from condor_status, condor_q, condor_history via "-json" flag
  - `condor_history -since` <*jobid* or *expression*>
- Config file syntax enhancements (includes, conditionals, …)
- …

Some enhancements in HTCondor v8.7 and beyond

Smarter and Faster Schedd

- User accounting information moved into ads in the Collector
  - Enable schedd to move claims across users
- Non-blocking authentication, smarter updates to the collector, faster ClassAd processing
- *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)

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, Azure, OpenStack, Cobalt

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

Jaime:
Grid
Jedi

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)

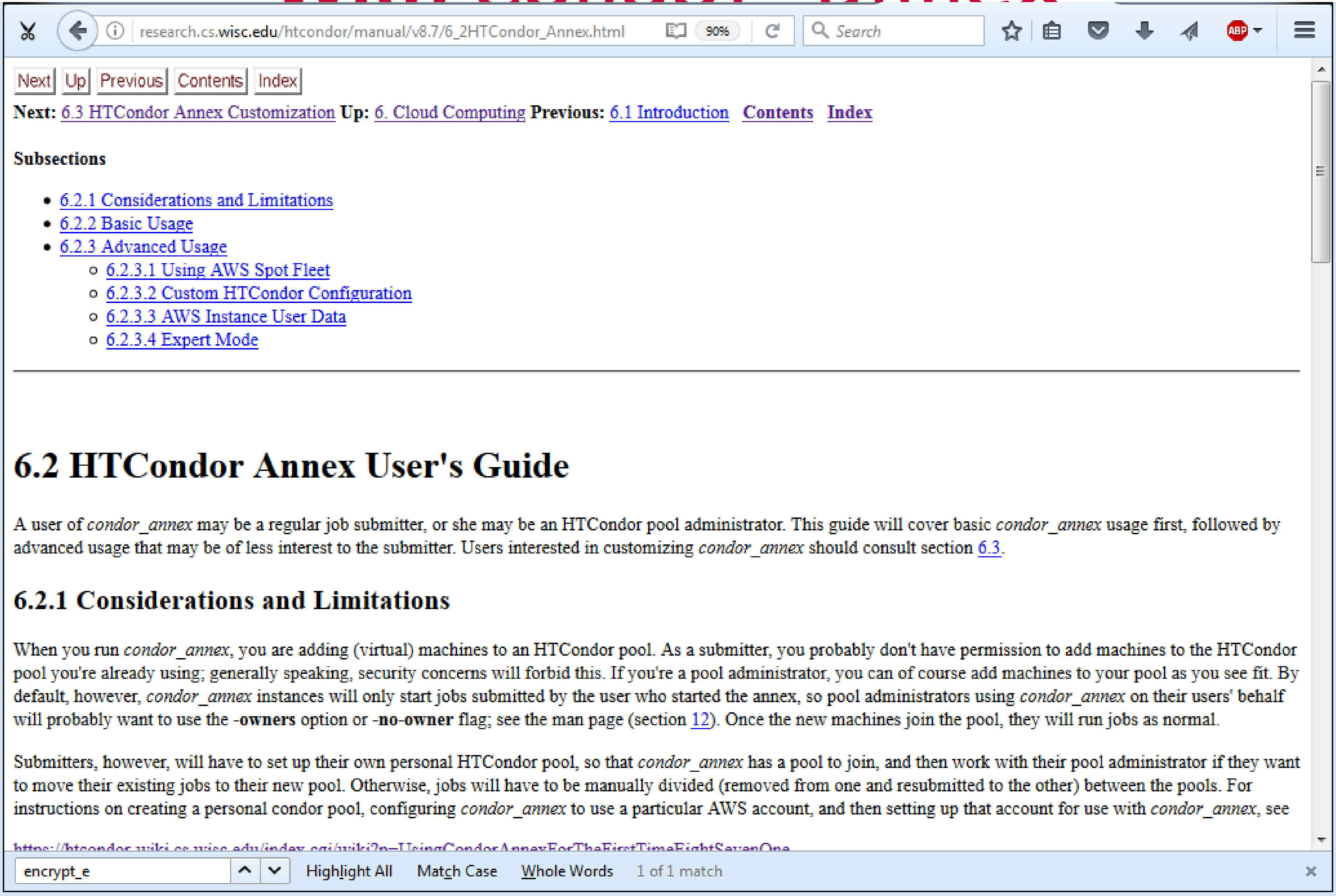
+ Decide on a bid for each instance type, according to its location (or pay more).

+ 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



Next: 6.3 HTCondor Annex Customization **Up:** 6. Cloud Computing **Previous:** 6.1 Introduction **Contents** **Index**

Subsections

- 6.2.1 Considerations and Limitations
- 6.2.2 Basic Usage
- 6.2.3 Advanced Usage
  - 6.2.3.1 Using AWS Spot Fleet
  - 6.2.3.2 Custom HTCondor Configuration
  - 6.2.3.3 AWS Instance User Data
  - 6.2.3.4 Expert Mode

6.2 HTCondor Annex User's Guide

A user of *condor_annex* may be a regular job submitter, or she may be an HTCondor pool administrator. This guide will cover basic *condor_annex* usage first, followed by advanced usage that may be of less interest to the submitter. Users interested in customizing *condor_annex* should consult section 6.3.

6.2.1 Considerations and Limitations

When you run *condor_annex*, you are adding (virtual) machines to an HTCondor pool. As a submitter, you probably don't have permission to add machines to the HTCondor pool you're already using; generally speaking, security concerns will forbid this. If you're a pool administrator, you can of course add machines to your pool as you see fit. By default, however, *condor_annex* instances will only start jobs submitted by the user who started the annex, so pool administrators using *condor_annex* on their users' behalf will probably want to use the **-owners** option or **-no-owner** flag; see the man page (section 12). Once the new machines join the pool, they will run jobs as normal.

Submitters, however, will have to set up their own personal HTCondor pool, so that *condor_annex* has a pool to join, and then work with their pool administrator if they want to move their existing jobs to their new pool. Otherwise, jobs will have to be manually divided (removed from one and resubmitted to the other) between the pools. For instructions on creating a personal condor pool, configuring *condor_annex* to use a particular AWS account, and then setting up that account for use with *condor_annex*, see

https://htcondor-wiki.cs.wisc.edu/index.cgi/wiki?p=UsingCondorAnnexForTheFirstTimeEightSevenOne

…Live demo of late job materialization and HTCondor Annex to EC2...

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

DAGMan Improvements

- ALL_NODES
  - `RETRY ALL_NODES 3`
- Flexible DAG file command order
- Splice Pin connections
  - Allows more flexible parent/child relationships between nodes within splices

New condor_status default output

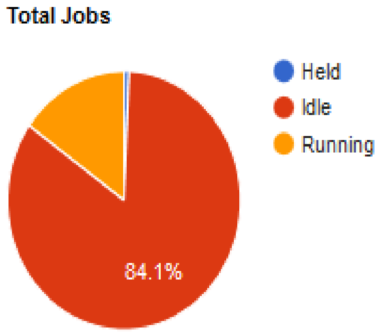
- Only show one line of output per machine
- Can try now in v8.5.4+ with "-compact" option
- The "-compact" option will become the new default once we are happy with it

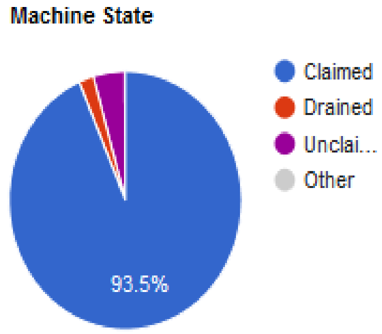
```
Machine      Platform     Slots Cpus Gpus  TotalGb FreCpu  FreeGb  CpuLoad ST

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
```

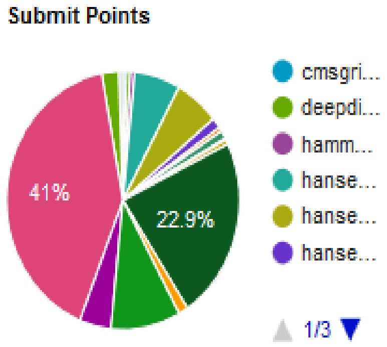
More backends for condor_gangaliad

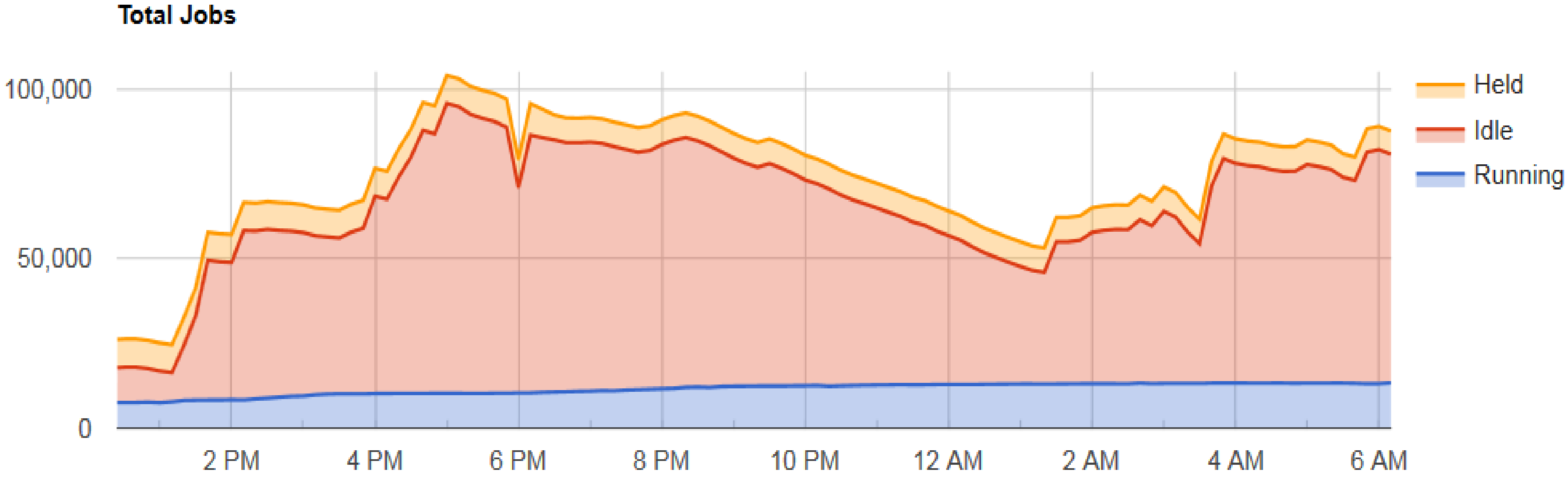
- In addition to (or instead of) sending to Ganglia, aggregate and make available in JSON format over HTTP
  - condor_gangliad rename to condor_metricd
- View some basic historical usage out-of-the-box by pointing web browser at central manager (modern CondorView)…
- Or upload to influxdb, graphite for Grafana

HTCondor View


Total Jobs


Machine State


Submit Points


Total Jobs

Potential Future Docker Universe Features?

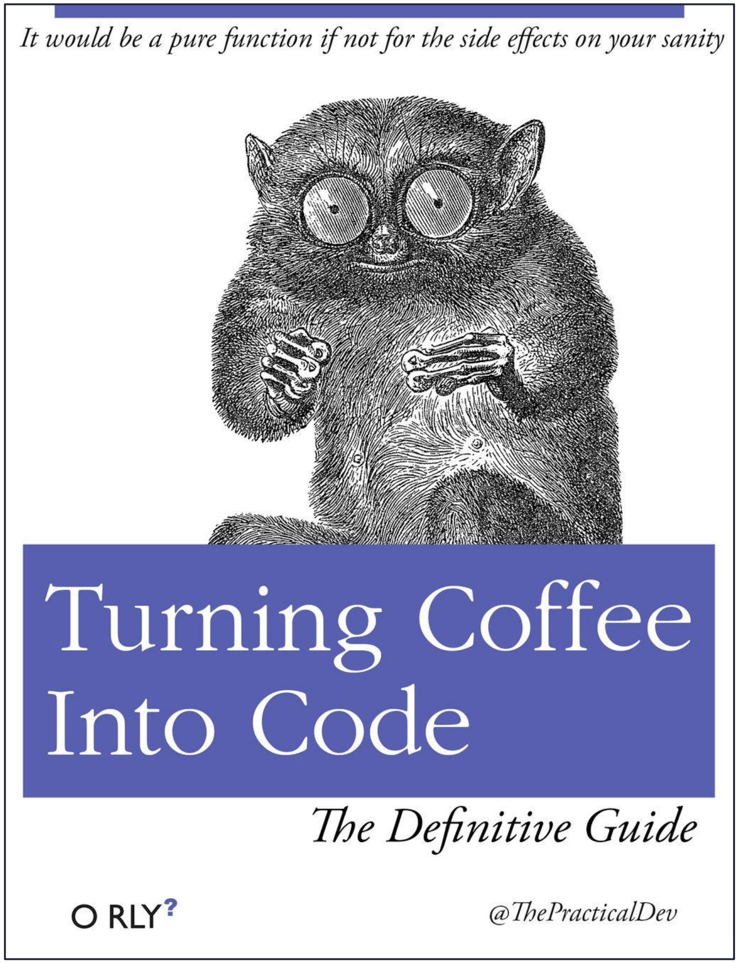
- Advertise images already cached on machine ?
- Support for condor_ssh_to_job ?
- Package and release HTCondor into Docker Hub ?
- Network support beyond NAT?
- Run containers as root??!?!?
- Automatic checkpoint and restart of containers! (via CRIU)

The future

- Working with the cloud : elasticity into the cloud.
- Scalability.
- More manageable, monitoring.
- Containers.
- Data, incl storage management options
- More Python interfaces

Thank You!

P.S. Interested in working on HTCondor full time? Talk to me! We are hiring! htcondor-jobs@cs.wisc.edu



It would be a pure function if not for the side effects on your sanity

Turning Coffee Into Code

The Definitive Guide

O RLY?

@ThePracticalDev