What's new in the HTCondor Software Suite (HTCSS) ? What's coming up?

European HTCondor Workshop 2022

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

Some New Terminology

- Releases
  - Long Term Support (LTS) Releases Channel
  - Feature Releases Channel
- HTCondor Software Suite (**HTCSS**)
  - Access Point (AP)
  - Execution Point (EP)
  - HTCondor Pool =

    Central Manager +

    Execution Point(s) (EP)
  - HTCondor Compute Entrypoint (HTCondor-CE)

Release Channels

- Long-Term Support (LTS) Releases (*formerly 'stable series'*)
  - Only bug fixes
  - vMajor.0.Update (e.g. 9.0.1, 9.0.2, … 9.0.15, 10.0.0, 10.0.1, 10.0.2, …)
  - Last year: v9.0 just released with 71 documented enhancements
  - Today: v9.0.17
- Feature Releases *(formerly 'developer series')*
  - Bug fixes plus new features
  - vMajor.Minor.Update (e.g. 9.1.0, 9.2.0, 9.3.0, 9.3.1, 9.4.0, …)
  - Today: v9.12.0 …. ***This is the v10.0.0 release candidate!***

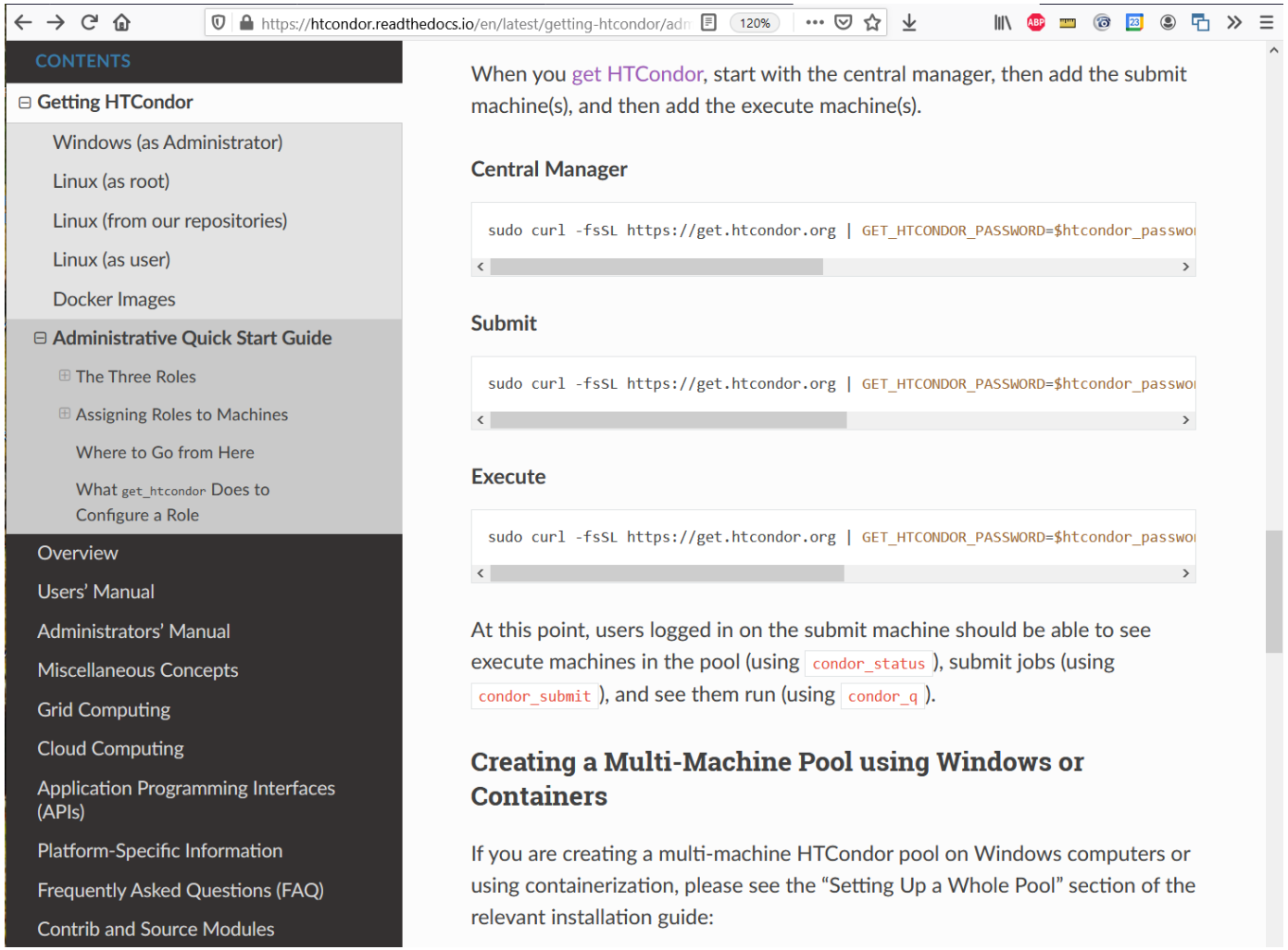
Update since last year

- Not an exhaustive list!
- For a complete list of "what has changed"
  - HTCondor
    - Highlights at: https://htcondor.org/htcondor/release-highlights/
    - Details at: https://htcondor.readthedocs.io/en/latest/version-history/development-release-series-91.html
  - HTCondor CE
    - Highlights at: https://htcondor.org/htcondor-ce/release-highlights/
    - Details at: https://htcondor.com/htcondor-ce/v5/releases/#htcondor-ce-5-version-history

What was introduced in HTCSS LTS v9.0.x?

Fast "one-line" install & configure

- Secure by default using ID tokens
- Demo and IDTokens explanation later this morning



New in v9.0 for end-users

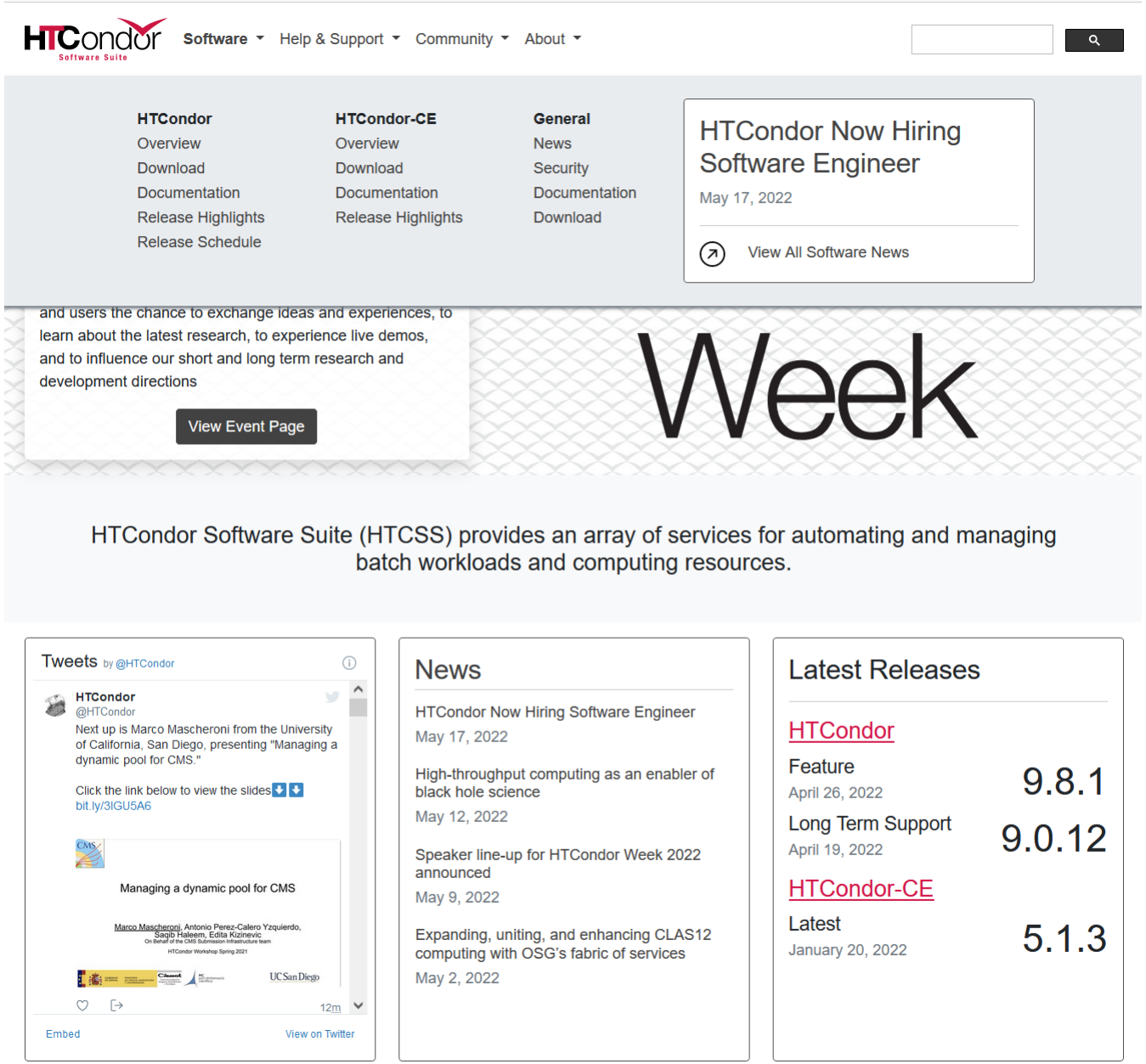
- Transfer job data to/from web servers, Box.com, Amazon S3, Google Drive, MS OneDrive via file transfer plugins now supporting uploads, authentication, and improved error handling
- Improved support for GPUs, including GPU utilization info and support for using GPUs in containers
- New condor_watch_q tool that efficiently provides live job status updates
- New tools and mechanisms to support jobs that checkpoint
- Fixed interactive submission of Docker jobs
- Many improvements to the Python API, including Python 3 support and new bindings for DAGMan and chirp
- DAGMan now provides a method for inline jobs to share submit descriptions
- You may now change some DAGMan throttles while the DAG is running
- You can assign priorities to DAGMan (scheduler universe) jobs
- And more…

New in v9.0 for sys admins

- Fundamental security model changes:
  - Secure by default
  - Authorization via Tokens: IDTOKENS, SciTokens
  - Oauth2 workflow to interoperate with other services
- New packaging
  - Native packaging has releases, release candidates, and daily builds
  - Official containers on Docker Hub designed to work well with k8s
  - Conda
- Push job information into Elastic Search
- More powerful classad transform language for HTCondor-CE, schedd
- Dataflow mode for jobs
- Container support improvements
- Cgroup limit improvements
- Cap the number of cores allocated to individual users
- New platforms (*e.g.* EL8, Ubuntu 20)
- And more…

What is coming in HTCSS v10.0.x ?

Updates to Web and Manual to reflect new terminology and the Suite idea



Upcoming OS and Architecture Support

Enterprise Linux 8 / Stream 8

- Available Now
- Based on Alma Linux

ARM and PowerPC architectures

- Available with v10.0.0
- Based on Alma Linux

Enterprise Linux 9 / Stream 9

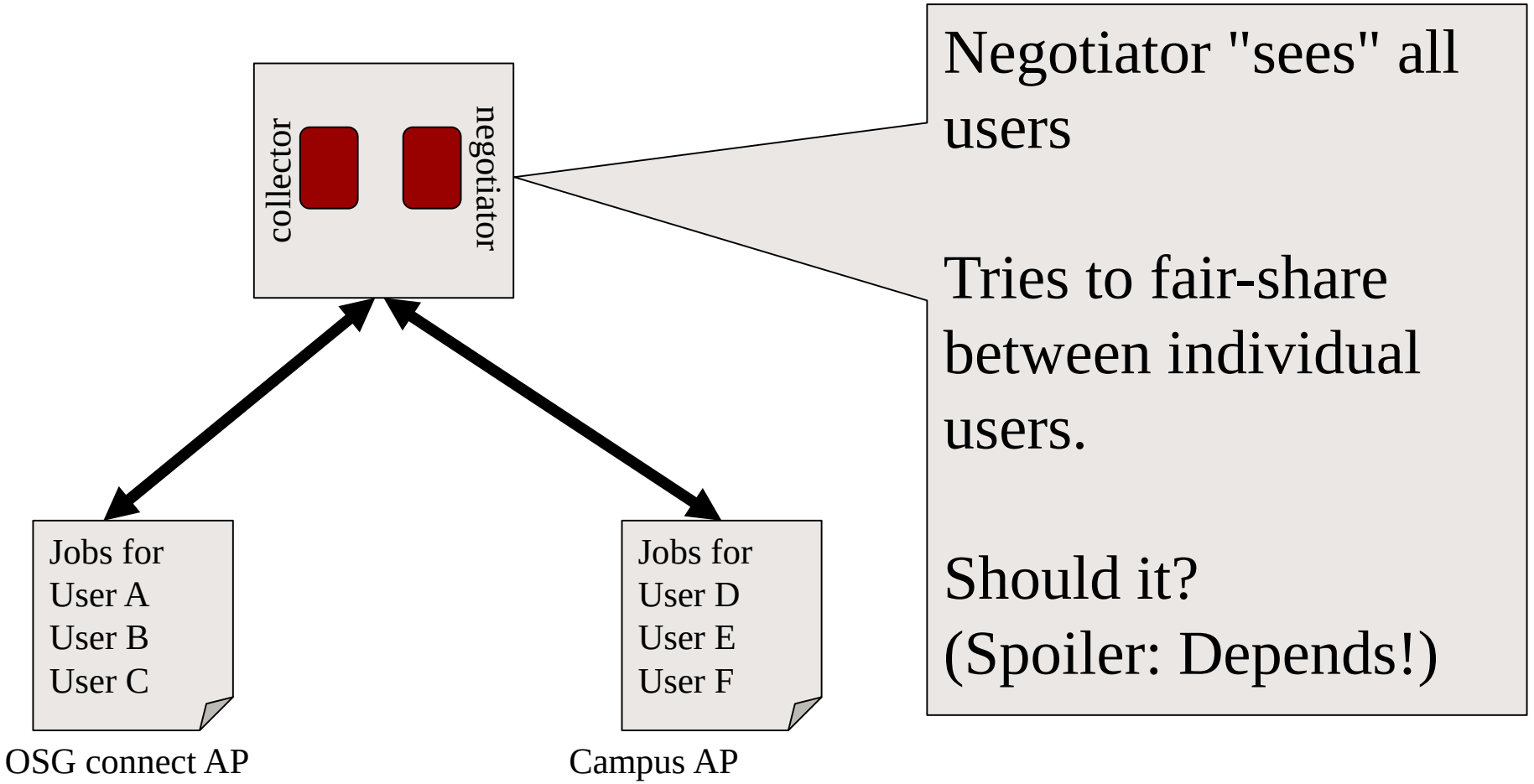
- Challenges due to shifts in several libraries (e.g. OpenSSL, cgroups)
- Interoperability with HTCSS v8.x may be limited (upgrade!)
- ETA: End of the year

Some Access Point Enhancements

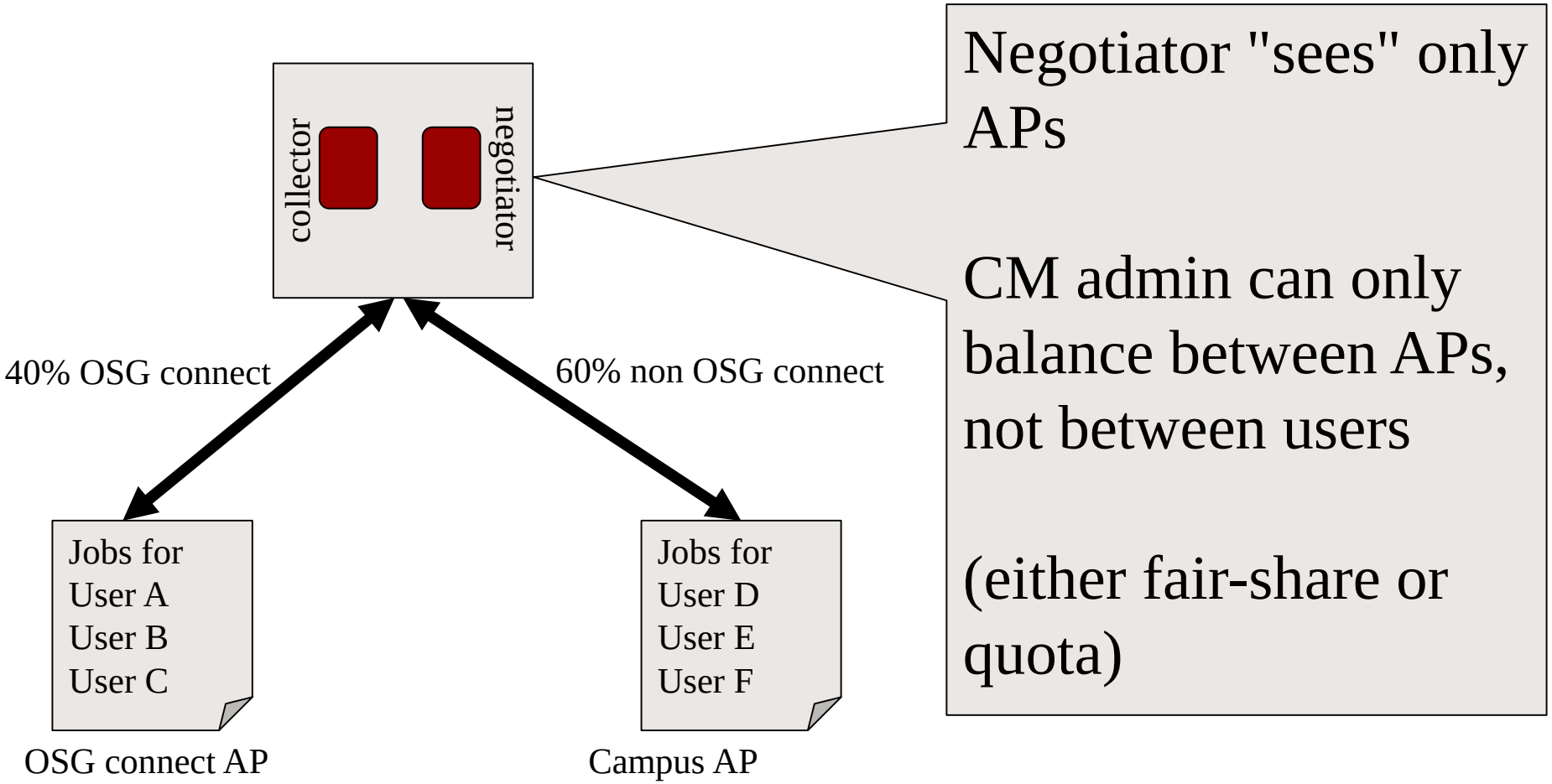
Stop Runaway ~~Trains~~ Jobs!

- Stop runaway jobs!
- Following can go in your submit file:
  - allowed_execute_duration = <seconds>
    (from exec to exit)
  - allowed_job_duration = <seconds>
    (include job transfers)

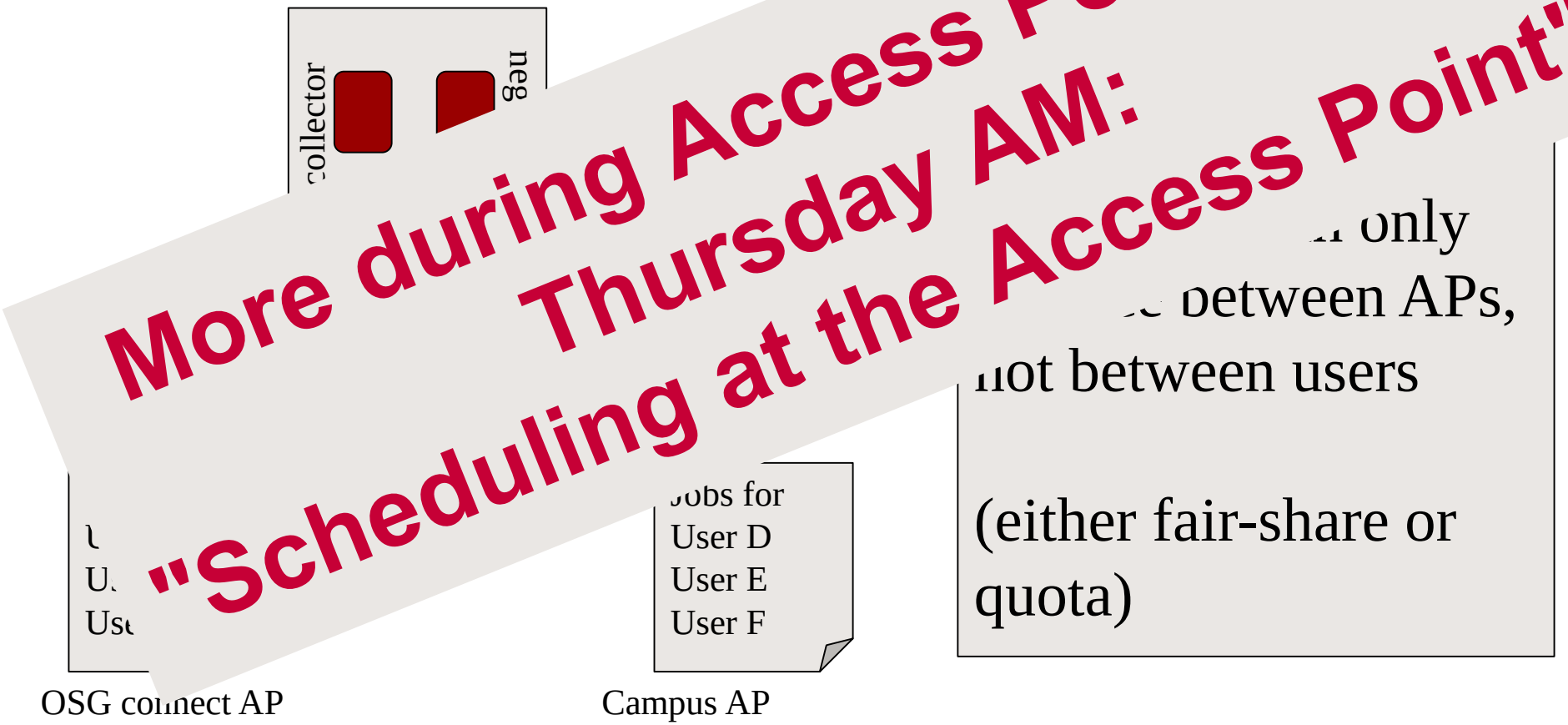
AP/CM interactions: As it is Today



A new mode for AP/CM interactions



A new mode for AP/CM interactions



More during Access Point talks Thursday AM:

"Scheduling at the Access Point"

Submit File Enhancements

- Can make "+CustomAttributes" first class with knobs EXTENDED_SUBMIT_[COMMANDS|HELPFILE]:

Instead of:

```
+SiteJobType = "analysis"
queue
```

Just:

```
SiteJobType = analysis
queue
```

- Define job submit templates:

```
use template: Matlab( mymatrix.m )
queue
```

Experimenting with new command line user interface

- *htcondor <noun> <verb>*
  - *"htcondor job submit", "htcondor job status", …*

New Command Line User Interface, cont

$ htcondor job status 123.45

Job 123.45 is currently running.
It started running 2.1 hours ago.
It was submitted 3.6 hours ago.
Its current memory usage is 2.5 GB out of 4.0 GB requested.
Its current disk usage is 3.8 GB out of 5.5 GB requested.
It has restarted 2 times.

Experimenting with new comma

More Weds AM: "Submitting Sets of Jobs"

- *htcondor <r*
  - *"htcondor job*
  - *"htcondor dag submit", "htcondor dag status", …*
  - *"htcondor jobset submit", "htcondor jobset status", …*
  - *"htcondor annex create", "htcondor annex status", …*
- Legacy tools (condor_q, condor_submit, …) not going anywhere…

Experimenting with new comma

More Thurs AM: "Bring Your Own Capacity Demo"

- *htcondor <r*
  - *"htcondor job*
  - *"htcondor dag submit", "htcondor dag status", …*
  - *"htcondor jobset submit", "htcondor jobset status", …*
  - *"htcondor annex create", "htcondor annex status", …*
- Legacy tools (condor_q, condor_submit, …) not going anywhere…

Some Execution Point Enhancements

New "Container Universe"

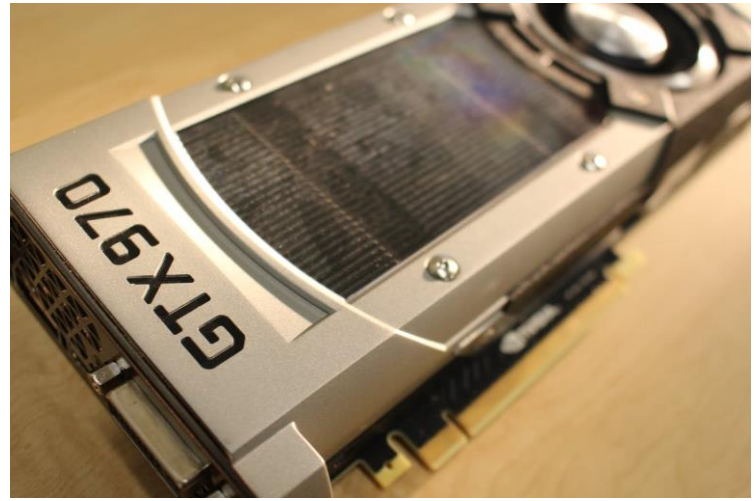
- EP advertises container runtimes available, and uses whichever one can get the job done

- New world order:


```
universe = container
container_image = /cvmfs/my/image/dir/
# Or container_image = docker://Debian
# Or container_image = myImage.sif
# Or container_image = http://xxx/image.sif
```

GPU Support Improvements

- HTCondor has long been able to detect GPU devices and schedule GPU jobs
- More recently also:
  - Monitor/report job GPU processor utilization
  - Monitor/report job GPU memory utilization



- ***New in v9.8.0: Support for heterogenous GPUs in one server***
  - E.g. a server with two different models of GPU cards
  - NVIDIA Multi-Instance GPU (MIG) partitioning

Submit File Example:

```
Executable = foo.exe
RequestGPUs = 1
RequireGPUs =
    Capability > 7.0
Queue
```

GPU Support Improvemen

- HTCondor has long been able to detect GPU devices and schedule GPU jobs
- More recently also:
  - Monitor/report job GPU pr
  - Monitor/report job
- ***New in GPU***
  - E.g. a cards
  - NVIDIA (MIG) partitioning

More during Execution Point talks Friday AM: "Worker nodes with GPUs"

Submit File Example:

```
Executable = foo.exe
RequestGPUs = 1
RequireGPUs =
    Capability > 7.0
Queue
```

Some Data Management Enhancements

File Transfer Error Propagation

- Hold Reason and Reason Codes now useful

12 [TransferOutputError]	An error occurred while transferring job output files or checkpoint files.	The Unix errno number.
13 [TransferInputError]	An error occurred while transferring job input files.	The Unix errno number.

Example of "condor_q –hold" (output directory missing on the Access Point)
Old Message:
Error from slot1@TODDS480S: STARTER at 127.0.0.1 failed to send file(s) to <127.0.0.1:50288>; SHADOW at 127.0.0.1 failed to write to file C:\condor\test\not_there\blah: (errno 2) No such file or directory

New Message:
Transfer output files failure at access point SUBMIT1 while receiving files from execution point slot3@NODE5. Details: writing to file C:\condor\test\not_there\blah: (errno 2) No such file or directory

Todd's Favorite File Transfer Enhancements

- Submit macro preserve_relative_paths = True

```
transfer_output_files  = result_data/x
```

  ends up creating 'result_data/x' on EP job sandbox, not 'x'.

- Config knob STARTD_ENFORCE_DISK_LIMITS = True
  - Create an ephemeral filesystem for the job sandbox on the EP.
  - Uses LVM or makes a loop-back file system
  - Reserves space on disk for the job
  - Enables EP to perform fast usage queries and cleanup

Todd's Favorite File Transfer Enha ements

- Submit macro preserve_relati
  `transfer_output_f`
  ends up creating
- Config k ITS = True
  - C job sandbox on the EP.
  - Us ck file system
  - Res sk for the job
  - Enabl perform fast usage queries and cleanup

More during Weds AM: "The HTCSS Data Story"

Security Enhancements

- Host-based security is no longer the default. ☺
- Turn-key installation with "get_htcondor" tool sets up all the security knobs for you
- For IDTOKENS authentication, no longer need a signing key on every EP host in your pool
  - EP belongs to a pool and trusts the Central Manager
  - Central Manager can give capabilities for admin tools
- TOFU ! "Trust on First Use" ability for SSL
- Improvements against replay attacks

Security Enhancements

- Host-based security is no long
- Turn-key installation wi the security knobs f
- For IDT ed a signing key on e
  - EP ts the Central Manager
  - Cent give capabilities for admin tools
- TOFU ! on First Use" ability for SSL
- Improvements against replay attacks

More during Security focus talks Weds PM

HTCondor-CE Work

- New syntax for writing job routes (classad transform syntax) has nice benefits including:
  - Statements evaluated in the order the are written
  - Use of variables that are not included in resultant job ad
  - Use of simple case statements
  - See https://htcondor.com/htcondor-ce/v5/configuration/writing-job-routes/
- Mechanism for a (hosted) CE to access capacity needing 2fa with ssh instead of just ssh alone
- Ability for a job route to insert an IDTOKEN

GSI to Tokens Transition

- Grid Community Toolkit support is going away, so HTCSS use of GSI authentication phased out
  - Use tokens instead (SCITOKENS, IDTOKENS)
  - Full details here:

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

- NOTE: HTCSS will still delegate and manage an X.509 proxy with the job, but the GSI authentication method is gone from v9.4 +

Thank you!

Follow us on Twitter!
https://twitter.com/HTCondor

This work is supported by NSF under Cooperative Agreement OAC-2030508 as part of the PATh Project. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the NSF

PATh PARTNERSHIP to ADVANCE THROUGHPUT COMPUTING