SGCI Webinar: Empowering job management in Science Gateways with HTCondor

August 9, 2017

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

- Open source distributed high throughput computing
- Schedule, provision, manage compute resources, containers, jobs, and workflows
- Primary objective: assist the scientific community with their high throughput computing needs
- Mature technology…
Mature... but actively developed

- Regular releases, both a stable (bug fixes only) and new features series
- Open source development model
- Evolve to meet the needs of the science community in an ever-changing computing landscape

<table>
<thead>
<tr>
<th></th>
<th>All Time</th>
<th>12 Month</th>
<th>30 Day</th>
</tr>
</thead>
<tbody>
<tr>
<td>Commits:</td>
<td>39067</td>
<td>2349</td>
<td>141</td>
</tr>
<tr>
<td>Contributors:</td>
<td>152</td>
<td>21</td>
<td>10</td>
</tr>
<tr>
<td>Files Modified</td>
<td>11588</td>
<td>1665</td>
<td>169</td>
</tr>
<tr>
<td>Lines Added:</td>
<td>12352208</td>
<td>444401</td>
<td>29395</td>
</tr>
<tr>
<td>Lines Removed</td>
<td>6810332</td>
<td>187595</td>
<td>7835</td>
</tr>
</tbody>
</table>

Source: https://www.openhub.net/p/condorproject
Why am I presenting?

- Our goal is for folks who work on science gateways to focus on the interaction with the science and scientists, and not worry about provisioning, scheduling, and job/workflow management.
- Job management work has been done, why reinvent?
- Many organizations have built gateways on top of HTCondor:
  - LHC CMS, Pegasus, BMRB, NEOS, Leica, Dreamworks, Hubble Telescope Operations, NOAA, ..
THREE THINGS YOU SHOULD KNOW ABOUT HTCONDOR

if you work on science gateways
Take Away Item #1: HTCondor embodies principle "Submit Local, Run Global"

What does this mean?
Well, HTCondor is different…

Unlike a typical scheduler, HTCondor does not require:

• A shared file system (NFS, Gluster, …) between submit and execute (worker) nodes
• Scheduler services installed by root and running as root
• Unified user logins between submit and execute nodes
• Full network communication between nodes
• Static and reliable set of compute servers

Result: Once a job is submitted into HTCondor, it can run almost anywhere
Science Gateway

Submit jobs or workflows

"Submit local... and run global!"

Local HTCondor cluster

Remote HTCondor cluster

Backfill SLURM, PBS, SGE campus clusters

Open Science Grid (OSG)

EXCEDE

Amazon EC2
Submitting Jobs

› Submit a single batch or interactive job, a "bag" of jobs, or a workflow via
  • Command line tools or
  • Python module

› Can specify resources required, files to transfer (if required) or stream, job priority, submitting user and group, retry policy, ...

› Monitor progress of your remote job via
  • command-line tools
  • Python module
  • streaming stdout / stderr
  • job event log file (both human readable and machine readable)
  • condor_ssh_to_job
Example of a job submit

% condor_submit  job.submit

job.submit

executable = analyze.exe
arguments = file.in file.out
transfer_input_files = file.in

log = job.log

queue 1
Example of a bag of 500 jobs with InitialDir

<table>
<thead>
<tr>
<th>job.submit</th>
<th>job0/</th>
<th>job1/</th>
<th>job2/</th>
</tr>
</thead>
<tbody>
<tr>
<td>analyze.exe</td>
<td>file.in</td>
<td>file.in</td>
<td>file.in</td>
</tr>
<tr>
<td></td>
<td>job.log</td>
<td>job.log</td>
<td>job.log</td>
</tr>
<tr>
<td></td>
<td>job.err</td>
<td>job.err</td>
<td>job.err</td>
</tr>
<tr>
<td></td>
<td>file.out</td>
<td>file.out</td>
<td>file.out</td>
</tr>
</tbody>
</table>

```plaintext
job.submit

executable = analyze.exe
initialdir = job$(ProcId)
arguments = file.in file.out
transfer_input_files = file.in

log = job.log
terror = job.err

queue 500
```
Workflows with condor_dagman

Basic DAG input file:

JOB nodes, PARENT-CHILD edges

my.dag

JOB A A.sub
JOB B1 B1.sub
JOB B2 B2.sub
JOB B3 B3.sub
JOB C C.sub

PARENT A CHILD B1 B2 B3
PARENT B1 B2 B3 CHILD C
Submitting a DAG to the queue

• Submission command:

```
$ condor_submit_dag dag_file
```

---

File for submitting this DAG to HTCondor: mydag.dag.condor.sub
Log of DAGMan debugging messages: mydag.dag.dagman.out
Log of HTCondor library output: mydag.dag.lib.out
Log of HTCondor library error messages: mydag.dag.lib.err
Log of the life of condor_dagman itself: mydag.dag.dagman.log

Submitting job(s).
1 job(s) submitted to cluster 87274940.
Take Away Item #2: Leverage the fact that HTCondor is a NoSQL database!

What does this mean?
Avoid challenges of synchronizing two databases

Science Gateway (SG)

Submit jobs or workflows

SG database

HTCondor database

19
Avoid challenges of synchronizing two databases

Science Gateway (SG)

Submit jobs or workflows

HTCondor database

HTCondor
Avoid challenges of synchronizing two databases

Science Gateway (SG)

Submit jobs or workflows

HTCondor database
Take Away Item #3: We've already done the work to handle job management *for you!*

What does this mean?
For me? Yes, for you!

- HTCondor is open source, funded primarily by NSF, developed within a University computing center
  - Proven, mature, but actively evolving
  - Multiple options for support
  - If you give HTCondor your workflow, many ways we can schedule and monitor. Doesn't already do something you need? Lets talk!

- Helping the scientific community is our primary focus.
Thank You!

Questions?