

Distributed Computing: Introduction to Condor

Kevin Meagher

12 June 2018

IceCube Bootcamp



WISCONSIN
UNIVERSITY OF WISCONSIN-MADISON

Distributed Computing

- In order to meet IceCubes processing requirement it is often necessary to use computing clusters
- Processing IceCube data is often “Embarrassingly Parallel”: Each event can be processed independently of every other event
- Processing is done in parallel by sending out jobs to a cluster of computers that independently process data

Condor Cluster

- IceCube has an HTCondor computer cluster referred to as NPX
- located at the UW-Madison computing center
- Explain condor
- ~7600 HT CPU cores
- ~400 GPUs
- Use submitter.icecube.wisc.edu to submit your jobs to NPX

```
mylaptop ~ $ ssh pub.icecube.wisc.edu  
pub1 ~ $ ssh submitter.icecube.wisc.edu
```

submitter.icecube.wisc.edu

```
submitter ~ $ pwd  
/home/kmeagher
```

```
submitter ~ $ ls /data/  
ana  exp  sim  user  wipac
```

```
submitter ~ $ ls /cvmfs/icecube.opensciencegrid.org/  
builddall.sh  data  distrib  iceprod  py2-v1  py2-v2  py2-v3  
py2-v3_early_access  README  setup.sh  standard
```

```
submitter ~ $ mkdir /scratch/kmeagher
```

Simple job script

- Use the provided scratch space.
- Logfiles in network filesystems (/home, /data/user) can generate instability in Condor.
- keep your job logfiles in local disk (typically /scratch)

```
sub-1 ~ $ mkdir /scratch/gmerino/; cd /scratch/gmerino/

sub-1 ~ $ wget http://icecube.wisc.edu/~gmerino/bootcamp/job.sh

sub-1 ~ $ cat job.sh
#!/bin/bash
printf "Start time: "; /bin/date
printf "Job is running on node: "; /bin/hostname
printf "Job running as user: "; /usr/bin/id
printf "Job is running in directory: "; /bin/pwd
echo "Working hard..."
sleep $1
echo "Job complete!"

sub-1 ~ $ chmod +x job.sh
sub-1 ~ $ ./job.sh 5
```

```
sub-1 ~ $ cat job.sub
executable = job.sh
arguments = 10

log = job.log
output = job.out
error = job.err

request_cpus = 1
request_memory = 100MB
request_disk = 1GB
#request_gpus = 1

queue 1
```

- In order to submit a job to the cluster you need a job file
- Specify your executable and any command line arguments it requires
- Log: file created by condor to track job progress
- output/error: captures standard output and standard error
- Request the amount of CPU, memory, disk, and GPU
- Queue: keyword telling it to create 1 instance of the job

Requesting Resources

- You must have some idea of how much of each resource your job will use (if you don't know measure it)
- If you do not specify resources the default for the cluster will be used.
- The amount default amount will vary a lot from cluster to cluster
- Do not rely on defaults!
- It is important to request the appropriate amount of resources for your job:
 - Too little -> Your job will be killed if you go over on any resource
 - Too much -> You will wait too long because there will be fewer "job slots" matching your requirements

Submitting and monitoring

To submit a job give condor your submit file:

```
sub-1 ~ $ condor_submit job.sub
Submitting job(s).
1 job(s) submitted to cluster 12898721.
```

To view your current running jobs:

```
sub-1 ~ $ condor_q
-- Schedd: sub-1.icecube.wisc.edu : <128.104.255.232:9618?... @ 06/07/18 11:43:06
To monitor submitted jobs: condor_q
OWNER   BATCH_NAME          SUBMITTED   DONE   RUN    IDLE  TOTAL JOB_IDS
gmerino ID: 101524801  6/7  11:43      _     _     1      1 101524801.0
Total for query: 1 jobs; 0 completed, 0 removed, 1 idle, 0 running, 0 held, 0 suspended
Total for gmerino: 1 jobs; 0 completed, 0 removed, 1 idle, 0 running, 0 held, 0 suspended
Total for all users: 1736 jobs; 0 completed, 0 removed, 1548 idle, 188 running, 0 held, 0 suspended
```

<http://research.cs.wisc.edu/htcondor/manual/v8.7/index.html>

Submitting and Monitoring

```
sub-1 ~ $ $ condor_q -nobatch
-- Schedd: sub-1.icecube.wisc.edu : <128.104.255.232:9618?... @ 06/07/18 11:50:11
ID            OWNER          SUBMITTED      RUN_TIME ST PRI SIZE CMD
101524801.0   gmerino          6/7  11:49    0+00:00:00 I  0   0.0 job.sh 10
```

Jobs can be submitted in batches

Individual jobs in a batch are identified by the number after the dot: the Process ID

JobID = ClusterID.ProcId

NOTE: Use `condor_q -all` if you want to see other user's jobs in the queue

Submitting and monitoring

```
sub-1 ~ $ $ condor_q -nobatch
-- Schedd: sub-1.icecube.wisc.edu : <128.104.255.232:9618?... @ 06/07/18 11:50:11
ID            OWNER          SUBMITTED      RUN_TIME  ST  PRI  SIZE  CMD
101524801.0   gmerino          6/7  11:49    0+00:00:00  I   0    0.0  job.sh 10
```

ST = status

Most Common Job status:

- **Idle “I”**: Job has not started yet... waiting in queue
- **Running “R”**: job is currently running
- **Completed**: If the job has completed, it will not appear in condor_q
- **Held “H”**: Stalled jobs. Something you need to fix

A job that goes on hold is interrupted (all progress is lost) and kept from running again. It remains in the queue in the “H” state.

Held Jobs removed by email

```
From: root <root@sub-1.icecube.wisc.edu>
Date: 14 June 2016 at 04:17
Subject: [htcondor] sub-1: held jobs removed
To:
12925663.0  gmerino      6/13 23:54 Policy violation. Memory limit exceeded: 2004 MB resident > 2000
MB requested. (by user condor)
12925674.0  gmerino      6/13 23:54 Policy violation. Memory limit exceeded: 2003 MB resident > 2000
MB requested. (by user condor)
12925692.0  gmerino      6/13 23:54 Policy violation. Memory limit exceeded: 2005 MB resident > 2000
MB requested. (by user condor)
```

We periodically scan for held jobs in the queue, remove them and notify users via email.

Some other typical hold reasons ...

```
12276425.0  gmerino      6/3  00:05 Error from slot1@e201.chtc.wisc.edu: Job failed to complete in 72 hrs
12015071.5  gmerino      3/23 14:33 Error from glidein_7164_75405897@a0437: STARTER at 10.80.2.181
failed to send file(s) to <128.104.255.232:59904>: error reading from
/home/icecu038/home_cream_966794713/CREAM966794713/glide_aXZBCc/execute/dir_34476/_condor_stdout: (errno
2) No such file or directory; SHADOW failed to receive file(s) from <134.93.174.12:35154>
```

Log File

```
sub-1 ~ $ cat job.log
000 (12898721.000.000) 06/13 21:50:14 Job submitted from host:
<128.104.255.232:58276?addrs=128.104.255.232-58276>
...
001 (12898721.000.000) 06/13 21:52:33 Job executing on host:
<144.92.166.137:27680?addrs=144.92.166.137-27680>
...
006 (12898721.000.000) 06/13 21:52:33 Image size of job updated: 1
  0 - MemoryUsage of job (MB)
  0 - ResidentSetSize of job (KB)
...
005 (12898721.000.000) 06/13 21:52:43 Job terminated.
  (1) Normal termination (return value 0)
    Usr 0 00:00:00, Sys 0 00:00:00 - Run Remote Usage
    Usr 0 00:00:00, Sys 0 00:00:00 - Run Local Usage
    Usr 0 00:00:00, Sys 0 00:00:00 - Total Remote Usage
    Usr 0 00:00:00, Sys 0 00:00:00 - Total Local Usage
314 - Run Bytes Sent By Job
281 - Run Bytes Received By Job
314 - Total Bytes Sent By Job
281 - Total Bytes Received By Job
Partitionable Resources :   Usage  Request  Allocated
  Cpus :                   1         1
  Disk (KB) :              12    102400    940361
  Memory (MB) :            0         100         100
```

Output File

```
sub-1 ~ $ cat job.out
Start time: Wed Jun 7 15:53:27 GMT 2017
Job is running on node: hibat0106.cmsaf.mit.edu
Job running as user: uid=10125(osg01) gid=10125(osg01) groups=10125(osg01),10005(osg)
Job is running in directory: /export/data1/condor/execute/dir_144460/glide_Rvbozm/execute/
dir_111261/glidein/execute.18.12.6.106-112180/dir_122685

Working hard...

Job complete!
```

Finding Job Attributes

```
sub-1 ~ $ condor_q -long 12887688.0
```

```
JobStatus = 2
```

```
LastJobStatus = 1
```

```
User = "gmerino@icecube.wisc.edu"
```

```
Err = "/scratch/gmerino/Data/2013/logs/12887688.err"
```

```
Out = "/scratch/gmerino/Data/2013/logs/12887688.log" NumJobStarts = 1
```

```
Args = "-g /data/exp/IceCube/2013/filtered/level2/0505/Run00122300/Level2_IC86.2013_data_Run00122300_0505_0_9_GCD.i3.gz -i /data/exp/IceCube/2013/filtered/level2/0505/Run00122300/Level2_IC86.2013_data_Run00122300_Subrun00000070.i3.bz2 -o /data/ana/Cscd/StartingEvents/exp/IC86_2013/burnsample/l3/00122300/Level2_IC86.2013_data_Run00122300_Part00000070.i3.bz2"
```

```
RemoteHost = "slot1@e281.chtc.wisc.edu"
```

```
ResidentSetSize_RAW = 1200308
```

```
DiskUsage_RAW = 891690
```

```
RemoteUserCpu = 7669.0
```

Useful Job Attributes

JobStatus: number indicating Idle (1), Running (2), Held (5), etc

RemoteHost: where the job is running

ResidentSetSize_RAW: Maximum observed physical memory in use by the job in KiB while running.

DiskUsage_RAW: Maximum observed physical memory in use by the job in KiB while running.

RemoteUserCpu: The total number of seconds of user CPU time the job has used. **EnteredCurrentStatus:** time of last status change

NumJobStarts: number of times the job started executing

http://research.cs.wisc.edu/htcondor/manual/v8.5/12_Appendix_A.html

Displaying Job Attributes

Use the “-autoformat” option for condor_q

```
sub-1 $ condor_q -af JobStatus ClusterID ProcId RemoteHost ResidentSetSize_RAW
2 12892531 0 slot1@glidein_235900_283164684@cabinet-0-0-7.t2.ucsd.edu 1272208
2 12892986 0 glidein_10345_623188368@jux7c.zeuthen.desy.de 1290364
2 12893002 0 slot1@e137.chtc.wisc.edu 1181296
```

The “-constraint” option can also be handy

```
sub-1 $ condor_q -c jobstatus==2 -af JobStatus ClusterID ProcId RemoteHost
ResidentSetSize_RAW
2 12892531 0 slot1@glidein_235900_283164684@cabinet-0-0-7.t2.ucsd.edu 1272208
2 12892986 0 glidein_10345_623188368@jux7c.zeuthen.desy.de 1290364
2 12893002 0 slot1@e137.chtc.wisc.edu 1181296
```


Displaying Machine Attributes

```
sub-1 $ condor_status -long -limit 1
...
Cpus = 4
Gpus = 0
Memory = 16000
...
GLIDEIN_Site = "UMD"
OpSysAndVer = "Ubuntu14"
...
GLIDEIN_Max_Walltime = 172500
MonitorSelfAge = 44406
...
```

Advanced Topics

IceCube has access to more computer resources at other sites besides UW-Madison, referred to as “The Grid”

- The Grid also uses HTCondor but does not have access to the `/data/sim` and `/data/exp` directories so files must be transferred using grid certificates

DAGMan is a tool that comes bundled with HTCondor. It can do two useful things:

- Control number of running jobs
- Handle inter-job dependencies

General Advice

- Never submit anything to the cluster without first estimating the memory usage and run time
- Always write your data to the provided scratch directory: `${_CONDOR_SCRATCH_DIR}`
- Always write logs to the scratch directory (not `/data/user`)

Links

Good tutorials can be found in the last [HTCondor week 2018](#):

- [An Introduction to Using HTCondor](#)
- [Introduction to Workflows with DAGMan](#)
- [HTCondor advanced job submission](#)

IceCube-specific information:

- [Grid Computing](#): pages in the IceCube wiki
- [Condor](#): pages in the IceCube wiki

Contact

Email:

- Problems/questions: help@icecube.wisc.edu
- discussion/information (mailing list): icecube-computing@icecube.wisc.edu

Slack channels:

- #icecube-it - Questions about HTCondor, the grid or any computing infrastructure
- #software - Questions about icetray or any software