Condor Scalability and Management at Brookhaven National Laboratory

Alexander Withers
alexw@bnl.gov

CondorWeek 2007

Overview of Condor at BNL

- RHIC/USATLAS Computing facility
  - Condor is the primary batchsystem in use
  - LSF still used by some users (provides global license counters)
- Not the only BNL group to use Condor
  - PHENIX experiment uses Condor to help power their 600MB/s DAQ/production facility
  - USATLAS Physics Applications Software group
- 4800+ processors running Condor
  - 5 pools, 3 central managers, 1 quill server
  - 4 grid gatekeepers, 100+ submit nodes

Policies in Use

- In general pools use either suspension or machine `Rank` with `MaxJobRetirementTime` to define a notion of priority
- Users add custom flags to their jobs to define the type of job
    - Other flags are added by Condor upon submission
    - The startd enforces restrictions by also looking at `Owner` and other job attributes
- `Preempt` for out of control jobs
- `Preemption_Requirements` and `MaxJobRetirementTime` for fairness between users

Example START Expression

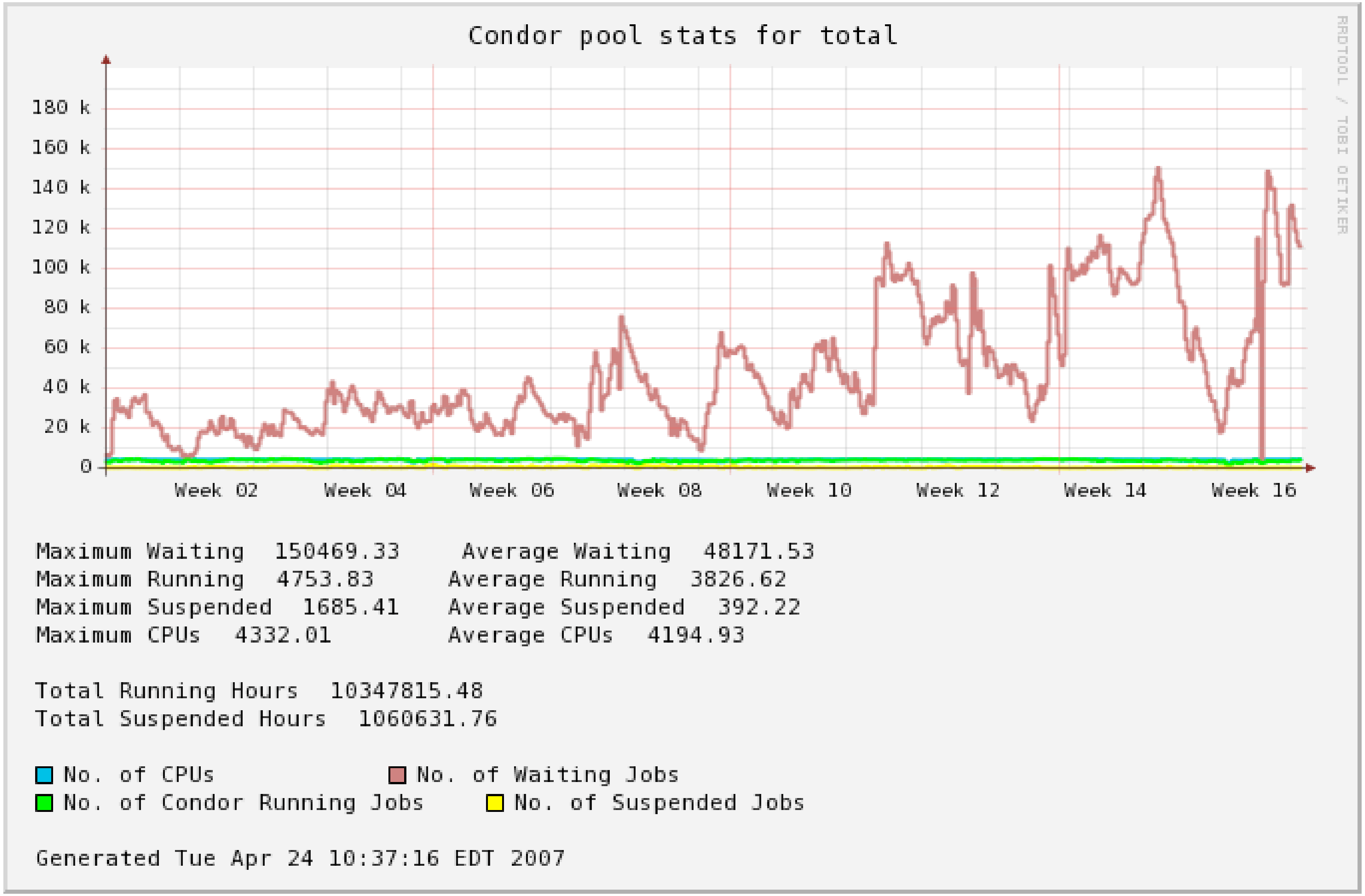
```
Start = (((RealExperiment == "atlas") && (VirtualMachineID >= 7) && ((TARGET.RACF_Group =?= "short" ||
TARGET.RACF_Group =?= "dial" || Owner =?= "usatlas2" || (stringListMember("acas0201",
"acas0200,acas0201,acas0202,acas0203,acas0204") && TARGET.RACF_Group =?= "lcg-ops") || (stringListMember
("acas0201", "acas0200,acas0201,acas0202,acas0203,acas0204") && TARGET.RACF_Group =?= "lcg-dteam")) &&
(RemoteWallClockTime < 5400))) || ((RealExperiment == "atlas") && ((VirtualMachineID < 7) &&
(VirtualMachineID >= 5)) && ((TARGET.RACF_Group =?= "usatlas" || TARGET.RACF_Group =?= "usatlas-grid" ||
(stringListMember("acas0201", "acas0200,acas0201,acas0202,acas0203,acas0204") && TARGET.RACF_Group =?=
"lcg-atlas") || TARGET.RACF_Group =?= "bnl-local") && ((((vm7_Activity =?= "Busy") + (vm7_Activity =?=
"Retiring") + (vm8_Activity =?= "Retiring") + (vm8_Activity =?= "Busy"))) < 2))) || ((RealExperiment ==
"atlas") && ((VirtualMachineID >= 3) && (VirtualMachineID < 5)) && ((TARGET.RACF_Group =?= "grid" ||
(stringListMember("acas0201", "acas0200,acas0201,acas0202,acas0203,acas0204") =?= FALSE &&
TARGET.RACF_Group =?= "lcg")) && ((((vm7_Activity =?= "Busy") + (vm7_Activity =?= "Retiring") +
(vm8_Activity =?= "Retiring") + (vm8_Activity =?= "Busy")) + ((vm5_Activity =?= "Busy") + (vm5_Activity
=?= "Retiring") + (vm6_Activity =?= "Retiring") + (vm6_Activity =?= "Busy"))) < 2))) ||
(((RealExperiment == "atlas") || (RealExperiment =!= "atlas" && FALSE == FALSE && TRUE == FALSE &&
LoadAvg < 1.400000 && TotalVirtualMemory > 200000 && ((Memory * 1024) - ImageSize) > 100000)) &&
((VirtualMachineID >= 1) && (VirtualMachineID < 3)) && ((TARGET.RACF_Group =?= "gridgr01" ||
TARGET.RACF_Group =?= "gridgr02" || TARGET.RACF_Group =?= "gridgr03" || TARGET.RACF_Group =?= "gridgr04"
|| TARGET.RACF_Group =?= "gridgr05" || TARGET.RACF_Group =?= "gridgr06" || TARGET.RACF_Group =?=
"gridgrXX" || TARGET.RACF_Group =?= "gridgr08" || TARGET.RACF_Group =?= "gridgr09" || TARGET.RACF_Group
=?= "gridgr10" || TARGET.RealExperiment =!= "atlas") && ((((vm7_Activity =?= "Busy") + (vm7_Activity =?=
"Retiring") + (vm8_Activity =?= "Retiring") + (vm8_Activity =?= "Busy")) + ((vm5_Activity =?= "Busy") +
(vm5_Activity =?= "Retiring") + (vm6_Activity =?= "Retiring") + (vm6_Activity =?= "Busy")) +
((vm3_Activity =?= "Busy") + (vm3_Activity =?= "Retiring") + (vm4_Activity =?= "Retiring") +
(vm4_Activity =?= "Busy"))) < 2)))) && (Owner =!= "jalex" && Owner =!= "grau" && Owner =!= "smithj4") &&
FALSE == FALSE)
```

Increase in Usage and Resources

- >400 users actively using Condor
- >10000 job slots
- Past 3 months: 2.8m jobs, 6.2m wallclock hours
- Computing resources added every year
  - New machines and Xen: even more job slots
  - Growth has been nonlinear, can we handle next year?
- Farm needs to be occupied with jobs
- Users need access to resources in a fair manner without significant delays
- Problem: one central manager may not be able to handle the load, how do we plan for the future?

Divide and Conquer

- Solution: divide the work load between three machines and divide the resources between five pools
  - Use flocking to create one virtual pool
  - Foreign jobs are immediately evicted if the resource is wanted by a local job
  - A user’s job will run on the other pools unless they prevent it from doing so
  - Response time has been very good, thus allowing growth
- Other measures to increase response time from negotiator:
  - SIGNIFICANT_ATTRS (now automatic)
  - Increased negotiation cycle



```
Maximum Waiting  150469.33    Average Waiting  48171.53
Maximum Running  4753.83     Average Running  3826.62
Maximum Suspended  1685.41   Average Suspended  392.22
Maximum CPUs  4332.01        Average CPUs  4194.93

Total Running Hours  10347815.48
Total Suspended Hours  1060631.76
```

🟦 No. of CPUs 🟫 No. of Waiting Jobs

🟩 No. of Condor Running Jobs 🟨 No. of Suspended Jobs

Generated Tue Apr 24 10:37:16 EDT 2007

Quill

- One quill server to handle all five pools
- First server (dual Xeon 3GHz, 4GB RAM, and SCSI drives with SW raid1) could not handle the load
  - `condor_q` would sometimes take 10 minutes
  - 100+ submit nodes being activity used
  - Optimizing postgresql didn't seem to help
- Investigated a variety of *small scale* storage hardware and configurations
  - Found it difficult to quantitatively measure Quill's performance
  - Used benchmarks to model the behavior of Quill that we were seeing

Quill, cont.

- Our tests involved a variety of factors: SATA vs. SAS, HW raid vs SW raid, etc.
- Baseline: SATA systems with SW raid10 and raid5 with minimum number of drives
- Results: SAS, HW raid, raid10 (no surprise), more spindles helps too
  - New server with 8GB of RAM, 6 drives for data, system and postgresql logs on other disks
- Other benefits: shared memory set to use half the system RAM and increased working RAM (postgresql specific parameters)
- Please contact me for specifics: alexw@bnl.gov

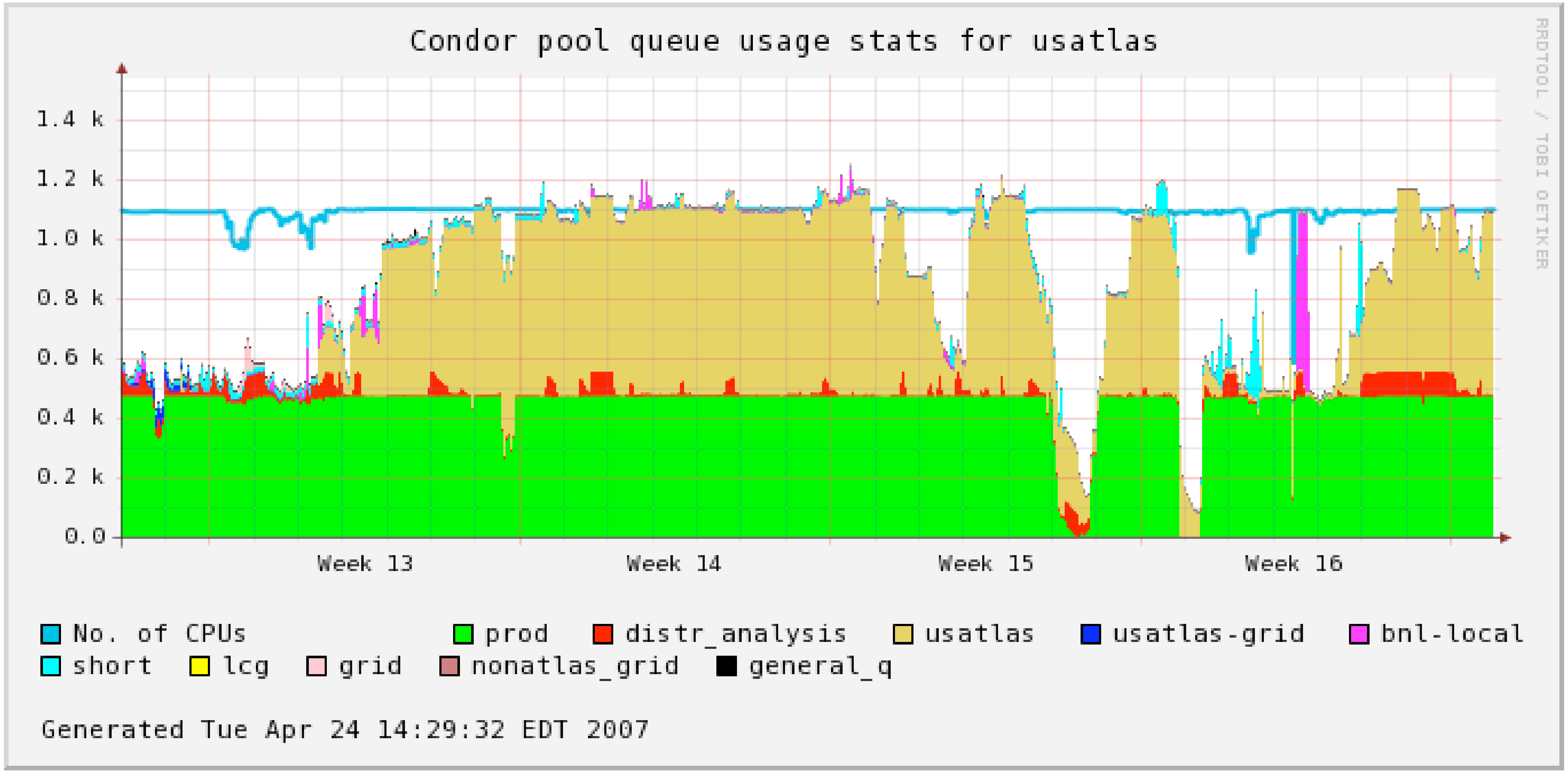
Monitoring and Maintenance

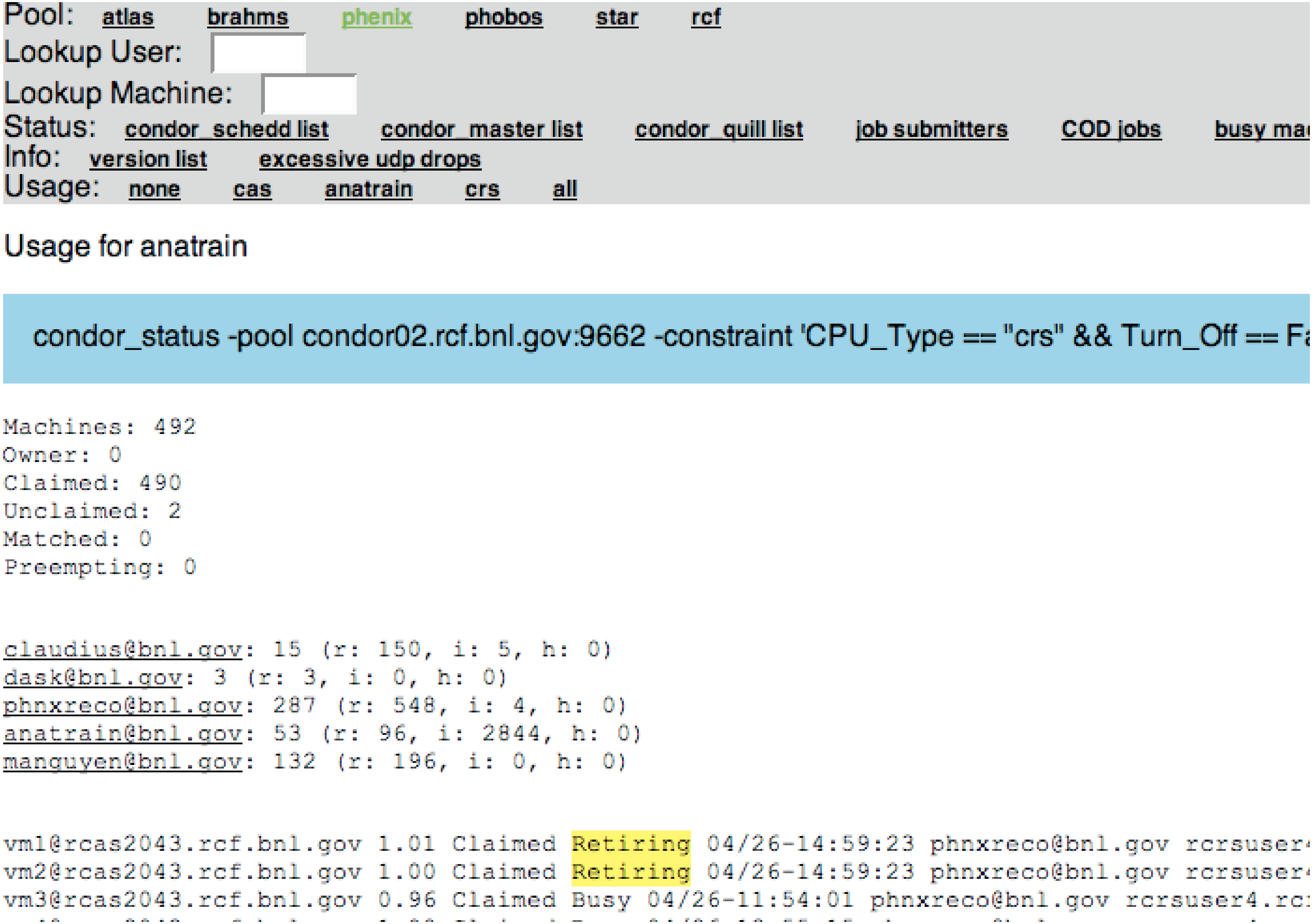
- 5 pools each with its own complex policy
  - Important to monitor and record usage
- We use several features in Condor to make monitoring easy
- Historical data stored in MySQL and RRDs
  - Quill is used as well to collect historical data
  - Quill's schema is sometimes difficult to deal with

Making Queries Easy

- Many submit machines: not easy to query the schedds
- User uses custom job attributes to target job
- Insert job's attributes into machine's classad: `STARTD_JOB_EXPRS`
- Make queries using these inserted attributes to show how many jobs are running where
- Insert other attributes to get an idea of who is using the resources, how much memory, disk usage, etc.

```
[root@acas0010 ~]# condor_status -constraint 'RACF_Group == "short"'

Name          OpSys       Arch   State      Activity   LoadAv Mem   ActvtyTime

vm13@acas0015 LINUX       INTEL  Claimed    Busy       0.970  1024  0+00:07:43
vm16@acas0015 LINUX       INTEL  Claimed    Busy       0.990  1024  0+01:13:16
vm13@acas0016 LINUX       INTEL  Claimed    Busy       0.950  1024  0+00:09:55
.
.
.
vm15@acas0110 LINUX       INTEL  Preempting Vacating   0.930  1024  0+00:00:10
vm16@acas0110 LINUX       INTEL  Claimed    Busy       0.960  1024  0+00:52:49
vm7@acas0188. LINUX       INTEL  Claimed    Busy       1.020  1024  0+01:01:05
vm8@acas0188. LINUX       INTEL  Claimed    Busy       0.980  1024  0+00:52:30
vm7@acas0189. LINUX       INTEL  Claimed    Busy       0.330  1024  0+01:01:44
vm8@acas0190. LINUX       INTEL  Claimed    Busy       1.010  1024  0+01:01:58

                     Total Owner Claimed Unclaimed Matched Preempting Backfill

         INTEL/LINUX   200     0     190         0       0         10        0

               Total   200     0     190         0       0         10        0
```


Condor pool queue usage stats for usatlas
1.4 k
1.2 k
1.0 k
0.8 k
0.6 k
0.4 k
0.2 k
0.0
Week 13
Week 14
Week 15
Week 16
No. of CPUs
prod
distr_analysis
usatlas
usatlas-grid
bnl-local
short
lcg
grid
nonatlas_grid
general_q
Generated Tue Apr 24 14:29:32 EDT 2007
RRDTOOL / TOBI OETIKER



Pool: atlas brahms phenix phobos star rcf

Lookup User:

Lookup Machine:

Status: condor_schedd list condor_master list condor_quill list job submitters COD jobs busy ma

Info: version list excessive udp drops

Usage: none cas anatrain crs all

Usage for anatrain

```
condor_status -pool condor02.rcf.bnl.gov:9662 -constraint 'CPU_Type == "crs" && Turn_Off == F
```

```
Machines: 492
Owner: 0
Claimed: 490
Unclaimed: 2
Matched: 0
Preempting: 0


claudius@bnl.gov: 15 (r: 150, i: 5, h: 0)
dask@bnl.gov: 3 (r: 3, i: 0, h: 0)
phnxreco@bnl.gov: 287 (r: 548, i: 4, h: 0)
anatrain@bnl.gov: 53 (r: 96, i: 2844, h: 0)
manguyen@bnl.gov: 132 (r: 196, i: 0, h: 0)


vm1@rcas2043.rcf.bnl.gov 1.01 Claimed Retiring 04/26-14:59:23 phnxreco@bnl.gov rcrsuser
vm2@rcas2043.rcf.bnl.gov 1.00 Claimed Retiring 04/26-14:59:23 phnxreco@bnl.gov rcrsuser
vm3@rcas2043.rcf.bnl.gov 0.96 Claimed Busy 04/26-11:54:01 phnxreco@bnl.gov rcrsuser4.rc
```

Dynamic Policy Changes

- Complex policy on each pool that allows a wide variety of job types to run
- Convenient to restrict certain jobs from running on certain nodes
- Solution: special machine attributes that can be set remotely
    - `SETTABLE_ATTRS_CONFIG, HOSTALLOW_CONFIG, ENABLE_*_CONFIG`
    - Machine attribute is placed in `START, RANK,` etc. expression
    - `condor_config_val -name rcas6006 -startd -set "CRS_Turn_Off = True"`

Dynamic Policy Example

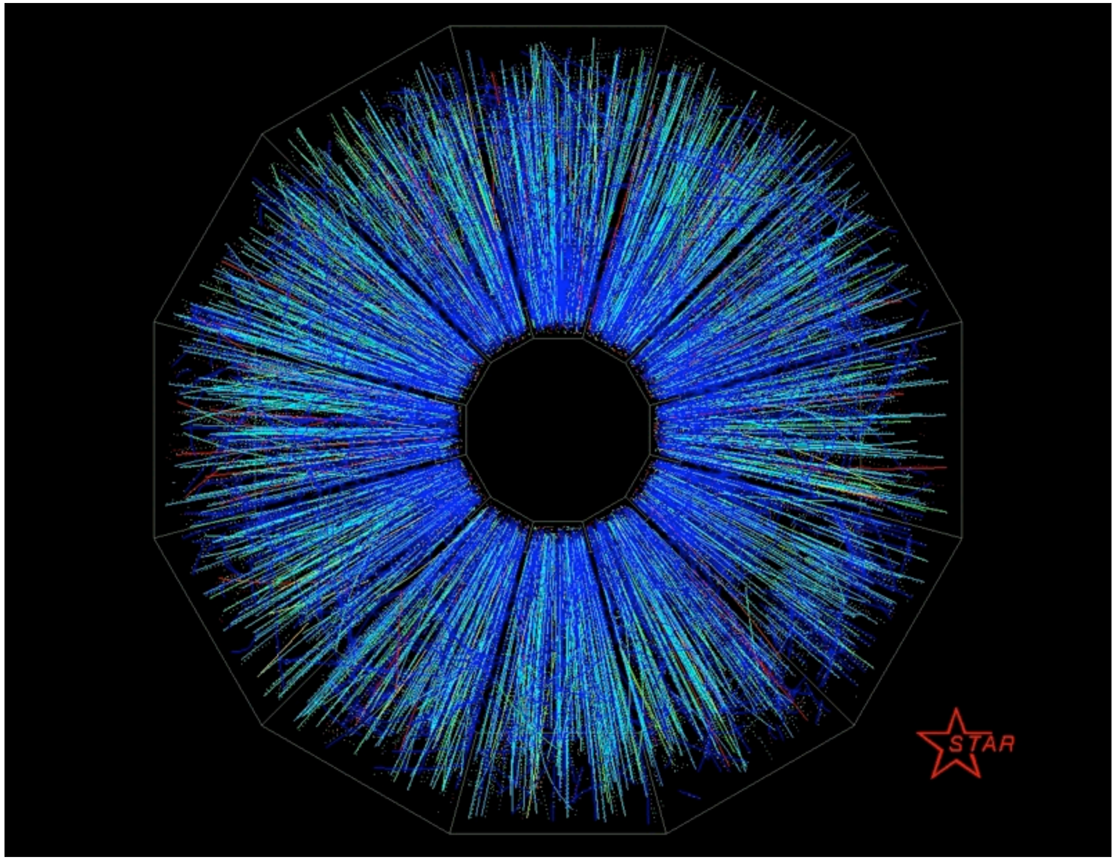
Prevent “crs” jobs from running but allow the current ones to finish

```
# local job start expression
CRS_Turn_Off = False
LOCAL_JOB = (RealExperiment == $(CPU_Experiment) && \
             ((Job_Type == "cas" && (VirtualMachineID == 1 || VirtualMachineID == 2)) || \
              (Job_Type == "osg" && (VirtualMachineID == 1 || VirtualMachineID == 2)) || \
              (Job_Type == "crs" && Owner == $(CPU_User) && $(CRS_Turn_Off) == False)))
```

```
[root@condor01 CONFIG]# condor_status -constraint 'CRS_Turn_Off == True'

Name          OpSys       Arch   State      Activity   LoadAv Mem   ActvtyTime

rcas6004.rcf. LINUX       INTEL  Claimed    Busy       2.720  8192  2+04:40:42
rcas6006.rcf. LINUX       INTEL  Owner      Idle       3.160  8192  0+00:20:04
.
.
.
rcas6115.rcf. LINUX       INTEL  Unclaimed  Idle       0.000  8192  0+03:00:04
rcas6156.rcf. LINUX       INTEL  Unclaimed  Idle       0.140  8192  0+01:50:04
```

Extending Condor

- Make heavy use of Condor’s cron facility
- Insert useful machine attributes such as 5 min. and 15 min. load
  - Can’t use these attributes in any startd expressions
  - Usually rely on `NEGOTIATOR_REQUIREMENTS`
- Other attributes are used by jobs
  - One example: projected disk usage
    - User transfer text file predicting how much disk space they will use (based on file placed in `_CONDOR_SCRATCH_DIR`)
    - Other jobs avoid machines where disk space



First Gold Beam-Beam Collision Events at RHIC at 30+30 GeV/c per beam recorded by STAR

Questions, Comments?

alexw@bnl.gov