



Grid Engine 6

Monitoring,
Accounting
& Reporting

BioTeam Inc.

info@bioteam.net

This module covers

- System Monitoring
- SGE Accounting File
- SGE Reporting
- Accounting & Reporting tools
- ARCo & 'sgeinspect'
- 3rd party tools & utilities

Grid Engine Accounting

SGE Accounting

- Who used what?
- Periodically SGE will write to
 - `$SGE_ROOT/$CELL/common/accounting`
 - This file is not rotated or truncated by default
 - Can grow very large
- The accounting file is plaintext
 - 1 line per entry, “:” delimited
 - Full format documented in `accounting (5)` man page
 - Warning: No internal unique key
 - Multiple lines can contain same JobID
 - (if a job was restarted, etc.)
- Contains lots of data but not everything you may care about
 - May have to derive/distill some values yourself

SGE Accounting

- SGE Parameters influencing accounting:
 - `$flush_time`
 - `$accounting_flush_time`
- By default:
 - `$flush_time` set to 15 seconds
 - `$accounting_flush_time` not set
 - SGE will honor `$flush_time` value in this case
 - Set `$accounting_flush_time` to decouple from reporting
 - Warning:
 - Setting of 00:00:00 disables buffering, not accounting!
 - To disable accounting
 - Add “accounting=false” to `reporting_params`

If you need to query accounting ...

```
$ qacct -help
GE 6.1beta
usage: qacct [options]
  [-A account_string]      jobs accounted to the given account
  [-b begin_time]         jobs started after
  [-d days]               jobs started during the last d days
  [-D [department]]      list [matching] department
  [-e end_time]           jobs started before
  [-g [groupid|groupname]] list [matching] group
  [-h [host]]             list [matching] host
  [-help]                display this message
  [-j [[job_id|job_name|pattern]]] list all [matching] jobs
  [-l attr=val,...]      request given complex attributes
  [-o [owner]]           list [matching] owner
  [-pe [pe_name]]        list [matching] parallel environment
  [-P [project]]         list [matching] project
  [-q [queue]]           list [matching] queue
  [-slots [slots]]       list [matching] job slots
  [-t taskid[-taskid[:step]]] list all [matching] tasks (requires -j option)
  [[-f] acctfile]        use alternate accounting file

begin_time, end_time    [[CC]YYMMDDhhmm[.SS]
queue                   [cluster_queue|queue_instance|queue_domain|pattern]
```

If you need to query accounting ...

- Start with builtin 'qacct'
 - Fairly good for simple stuff
 - Manpage or "qacct -help" covers usage
- Not too hard to roll your own
- Ruby analyzer.rb script
 - In CVS maintrunk `source/scripts/analyze.rb`
 - Also at:
 - <http://gridengine.sunsource.net/files/documents/7/82/analyze.rb.gz>

Ruby accounting analyzer

```
$ ./analyze.rb
```

```
usage: analyze.rb <options> accounting_file
```

```
-help
```

```
-r                records table
```

```
-u                users table
```

```
-h                hosts table
```

```
-q                queues table
```

```
-p                projects table
```

```
-c                categories table
```

```
-ts               timesteps table
```

```
-ts_c             categories per timestep
```

```
-ts_j             jobs per timestep
```

```
-t "first" | <first> "last" | <last> full analysis, these timesteps only
```

Ruby analyzer.rb - User report

■ analyze.rb -u (truncated)

```
$ ./analyze.rb -u /opt/sge61/default/common/accounting
```

```
... read 48 records
```

```
... debug did users
```

```
##### Table with 2 users #####
```

user	njobs	sum	pend	runtime	cpu	maxvmem	maxrss
dag	47		3689	1916	0	38986584	0
root	1		0	1	0	0	0

Ruby analyzer.rb - Timesteps

■ analyze.rb -ts

```
$ ./analyze.rb -ts /opt/sge61/default/common/accounting
```

```
...
```

```
1175795130      0 0   0   0      1 ended 33
1176471751      0 0   0   0     67662 submitted 34 started 34 ended 34
1176471797     10 1   0   0      46 submitted 35.3, 35.4, 35.9, 35.10, 35.1, 35.2, 35.7, 35.8,
    35.5, 35.6
1176471811      7 1   0   0      14 started 35.3, 35.1, 35.2 ended 35.3, 35.1, 35.2
1176471826      4 1   0   0      15 started 35.4, 35.5, 35.6 ended 35.4, 35.5, 35.6
1176471841      1 1   0   0      15 started 35.9, 35.7, 35.8 ended 35.9, 35.7, 35.8
1176471856      0 0   0   0      15 started 35.10 ended 35.10
1176673763      1 1   0   0     20190 submitted 36
1176673772      0 0   0   0       9 started 36 ended 36
```

```
...
```

Ruby analyzer.rb - Timestep by job

■ analyze.rb -ts_j

```
$ ./analyze.rb -ts_j /opt/sge61/default/common/accounting
```

```
...
```

```
##### Jobs at timestep 1176674102 #####
```

job	status	user	pending	category
38	running	dag	5	"-u dag -l ifort_compiler_lic=50"
39	pending	dag	79	"-u dag -l ifort_compiler_lic=50"
40	pending	dag	150	"-u dag -l ifort_compiler_lic=1"

Grid Engine Reporting

Reporting

- SGE can log additional information to a special file
 - `$SGE_ROOT/$CELL/common/reporting`
 - One line per entry, same “:” delimiter as accounting file
 - Also not rotated or truncated automatically
 - Disabled by default
- Multiple record types in same file
 - Second field of reporting entry defines the record type:
 - `new_job`
 - `job_log`
 - `queue`
 - `queue_consumable`
 - `host`
 - `host_consumable`
- Man page “reporting (5)” defines formats

Reporting file excerpt ...

```
# Version: 6.1beta
#
# DO NOT MODIFY THIS FILE MANUALLY!
#
1176858091:host:cd:1176858091:X:cpu=12.300000,np_load_avg=0.340820,mem_free=403.042969M,vir
tual_free=403.042969M

1176858136:queue_consumable:all.q:cd:1176858136::slots=1.000000=4.000000

1176858137:acct:all.q:cd:UNKNOWN:root:hostname:41:sge:0:1176858136:1176858136:1176858137:0:
0:1:0:0:0.000000:0:0:0:0:0:0:0.000000:2:4:125:0:11:0:NONE:defaultdepartment:NONE:1:0:0
.000000:0.000000:0.000000:-I y:0.000000:NONE:0.000000

1176858137:queue_consumable:all.q:cd:1176858137::slots=0.000000=4.000000
1176858181:host_consumable:global:1176858181:X:ifort_compiler_lic=10.000000=50.000000
1176858181:queue_consumable:all.q:cd:1176858181::slots=1.000000=4.000000
1176858181:host_consumable:global:1176858181:X:ifort_compiler_lic=20.000000=50.000000
1176858181:queue_consumable:all.q:cd:1176858181::slots=2.000000=4.000000
1176858181:host_consumable:global:1176858181:X:ifort_compiler_lic=30.000000=50.000000
1176858181:queue_consumable:testQueue:cd:1176858181::slots=1.000000=4.000000
```

Reporting file with joblog=true

```
# Version: 6.1beta
#
# DO NOT MODIFY THIS FILE MANUALLY!
#

1176859069:new_job:1176859069:55:1:NONE:simple.sh \
: dag: dag: : defaultdepartment: sge: 1024

1176859069:job_log:1176859069:pending:55:-1:NONE:: \
dag: cd: 0: 1024: 1176859069: simple.sh: dag: dag: : \
defaultdepartment: sge: new job

1176859070:job_log:1176859070:delivered:51:0:NONE:r: \
master: cd: 0: 1024: 1176859066: simple.sh: dag: dag: : \
defaultdepartment: sge: job received by execd
```

Historical context: Reporting

- Not widely used in Open Source community
- Primarily something to turn on when troubleshooting & debugging
- Can load qmaster host & generate massive files if not looked after
- Starting to change in '08-09
 - Especially via UnivaUD products

Historical context: Reporting

- Reporting subsystem usage likely to increase
- Reason:
 - ARCo joining open source codebase in SGE 6.1
 - Lots of people claim interest now that it is “free”
- Finally a reason to leave `reporting=true` enabled

How to enable reporting

1. Adjust “reporting_params” in SGE qmaster configuration
 - `reporting=true,`
`flush_time=00:00:15,`
`joblog=true|false`
2. Tell SGE what variables to report
 - Several places to do this, docs recommend global exec host config (“qconf -me global”)
 - `report_variables=cpu,np_load_avg,mem_free,_virtual_free`

A few slides on ARCo ...

Grid Engine ARCo

- “Analysis & Reporting Console”
 - Web front end to reports generated by SGE data scraped into a SQL repository
- Formally a layered product for N1GE 6
- Now part of Grid Engine as of SGE 6.1
- Three main components
 - Sun Java Web Console (swc) **
 - SGE dbwriter
 - SGE ARCo

Sun Web Console

- Dedicated Sun web application server environment
 - Available for Linux, Solaris, Windows & HP-UX
 - All Sun “N1” systems management tools plug into this framework
- *As of March 2008*
 - *Sun webconsole is offered as a download optional extra when downloading the official SGE binaries*

SGE 'dbwriter'

- Part of SGE since 6.1 release
- Usable with SGE 6.0
 - *Take from N1GE 6 download on sun.com*
- Implemented in Java
- What it does
 1. *Scrapes accounting & reporting files*
 2. *Calculates new "derived" values*
 - *Can customize, create own derived values*
 3. *Speaks JDBC to a database resource*
 - *Oracle*
 - *PosgreSQL*
 - *MySQL 5 or later (requires views ...)*
 4. *Inserts new data into SQL, deletes "old" data per policy*

SGE “ARCo” module

- Packaged webapp for Sun Java Web Console
- Web front end to data stored in the dbwriter-created SQL repository
- Not particularly polished interface
 - Any level past the canned reports forces end-user to type SQL statements into a textarea box on web form
- My \$.02
 - Keep dbwriter including the SQL schemas it uses
 - Works well at what it does; don't reinvent wheel ...
 - Roll your own web front end

Grid Engine ARCo

The screenshot shows a web browser window titled "Log In - Sun Java(TM) Web Console". The address bar contains the URL "https://10.211.55.4:6789/console/faces/jsp/login/BeginLogin.jsp?Cor". The browser's address bar also shows "gridengine.info" and "Log In - Sun Java(TM) Web Con...". The page features a "VERSION" button on the left and a "HELP" button on the right. The main content area displays the "Java™ Web Console" logo and a yellow warning box with the message "Session Timed Out" and "Your user session has timed out. Log in again." Below this, there are input fields for "Server Name" (pre-filled with "vcentos-a"), "User Name" (pre-filled with "sge"), and "Password" (masked with "*****"). A "Log In" button is positioned below the password field. The Sun logo is visible in the bottom left corner of the page. At the bottom of the page, there is a copyright notice: "Copyright © 2006 Sun Microsystems, Inc. All rights reserved. U.S. Government Rights - Commercial software. Government users are subject to the Sun Microsystems, Inc. standard license agreement and applicable provisions of the FAR and its supplements. Use is subject to license terms. This distribution may include materials developed by third parties. Sun, Sun Microsystems, the Sun logo, Java, Netra, Solara, StarOffice, Sun StorEdge and Sun[tm] ONE are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries." The browser's status bar at the bottom shows "Done" on the left and "10.211.55.4:6789" on the right, along with icons for "s3", "Fox", and "S".

SGE training, consulting and special projects - BioTeam Inc. - <http://www.bioteam.net>

Grid Engine ARCo

The screenshot displays the Sun Java(TM) Web Console interface. The browser address bar shows the URL `https://10.211.55.4:6789/console/launch/Launch`. The page header includes navigation links for 'APPLICATIONS' and 'VERSION', and buttons for 'LOGOUT' and 'HELP'. The user is identified as 'sgc' on server 'vcentos-a'. The main content area is titled 'Java™ Web Console' and features a checkbox for 'Start Each Application in a New Window'. Below this, there are four sections: 'Systems' (containing 'SUN Grid Engine 6.2u3beta ARCo'), 'Desktop Applications' (with 'No applications available'), 'Storage' (with 'No applications available'), and 'Services' (with 'No applications available'). The status bar at the bottom indicates 'Done' and the IP address '10.211.55.4:6789'.

SGE training, consulting and special projects - BioTeam Inc. - <http://www.bioteam.net>

Grid Engine ARCo

The screenshot displays the SUN Grid Engine - ARCo web interface. The browser address bar shows the URL <https://10.211.55.4:6789/reporting/arcomodule/Index>. The page header includes navigation links for 'APPLICATIONS' and 'VERSION', and user information: 'User: sge Server: vcentos-a'. The main title is 'SUN Grid Engine - ARCo' with a 'Java' logo. A 'Cluster: p6444' dropdown is visible. The 'Overview' section is active, showing 'List all defined queries and results'. Below this, there are tabs for 'Query List' and 'Result List'. The 'Query List' tab is selected, displaying a table of 19 queries. The table has columns for Name, Category, LastModified, and Type. The queries listed include Accounting per AR, Accounting per Department, Accounting per Project, Accounting per User, Advance Reservation Attributes, Advance Reservation by User, Advance Reservation Log, Advance Reservation Time Usage, Average Job Turnaround Time, Average Job Wait Time, DBWriter Performance, Host Load, Job Log, Number of Jobs completed, Number of Jobs Completed per AR, Queue Consumables, Statistic History, Statistics, and Wallclock time.

Name	Category	LastModified	Type
Accounting per AR	Accounting	Thu May 07 20:29:16 EDT 2009	advanced
Accounting per Department	Accounting	Thu May 07 20:29:16 EDT 2009	advanced
Accounting per Project	Accounting	Thu May 07 20:29:16 EDT 2009	advanced
Accounting per User	Accounting	Thu May 07 20:29:16 EDT 2009	advanced
Advance Reservation Attributes	Advance Reservation	Thu May 07 20:29:16 EDT 2009	simple
Advance Reservation by User	Advance Reservation	Thu May 07 20:29:16 EDT 2009	simple
Advance Reservation Log	Advance Reservation	Thu May 07 20:29:16 EDT 2009	simple
Advance Reservation Time Usage	Advance Reservation	Thu May 07 20:29:16 EDT 2009	advanced
Average Job Turnaround Time	Job	Thu May 07 20:29:16 EDT 2009	advanced
Average Job Wait Time	Job	Thu May 07 20:29:16 EDT 2009	advanced
DBWriter Performance	Administration	Thu May 07 20:29:16 EDT 2009	advanced
Host Load	Cluster	Thu May 07 20:29:16 EDT 2009	advanced
Job Log	Job	Thu May 07 20:29:16 EDT 2009	simple
Number of Jobs completed	Job	Thu May 07 20:29:16 EDT 2009	advanced
Number of Jobs Completed per AR	Job	Thu May 07 20:29:16 EDT 2009	advanced
Queue Consumables	Resource Usage	Thu May 07 20:29:16 EDT 2009	advanced
Statistic History	Administration	Thu May 07 20:29:17 EDT 2009	advanced
Statistics	Administration	Thu May 07 20:29:17 EDT 2009	advanced
Wallclock time	Jobs	Thu May 07 20:29:17 EDT 2009	simple

SGE training, consulting and special projects - BioTeam Inc. - <http://www.bioteam.net>

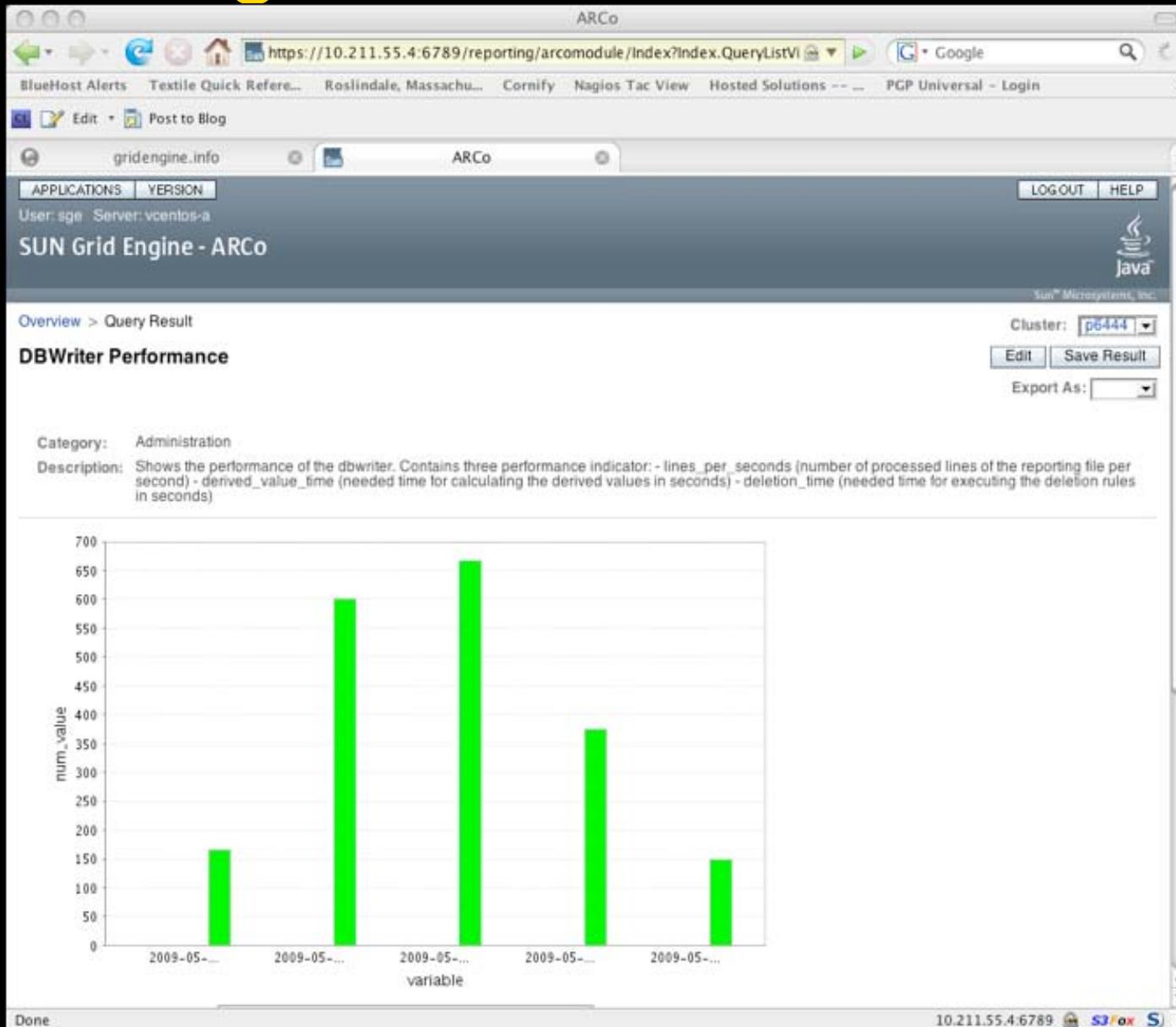
Grid Engine ARCo

The screenshot displays the SUN Grid Engine ARCo web interface. The browser address bar shows the URL `https://10.211.55.4:6789/reporting/arcomodule/Index?Index.QueryListVI`. The page title is "SUN Grid Engine - ARCo". The interface includes a navigation menu with "APPLICATIONS" and "VERSION" tabs, and a "LOG OUT" button. The main content area shows the "Overview > Query Result" section for "Accounting per User". The query is categorized as "Accounting" and is described as "Shows the monthly accounting per user for the interval of one year." The SQL query is: `SELECT date_format(start_time, '%Y-%m-01') AS time, username, SUM(cpu) AS cpu, SUM(mem) AS mem, SUM(io) AS io FROM view_accounting WHERE start_time > (current_timestamp - interval 1 year) GROUP BY date_format(start_time, '%Y-%m-01'), username`. A "Pivot Table" is displayed with the following data:

Pivot Table			
2009-05-01			
	cpu	mem	io
dag	0.581907	0.00314	0.0

The interface also includes a "Cluster:" dropdown menu set to "pb444", "Edit" and "Save Result" buttons, and an "Export As:" dropdown menu. The status bar at the bottom shows "Done" and the IP address "10.211.55.4:6789".

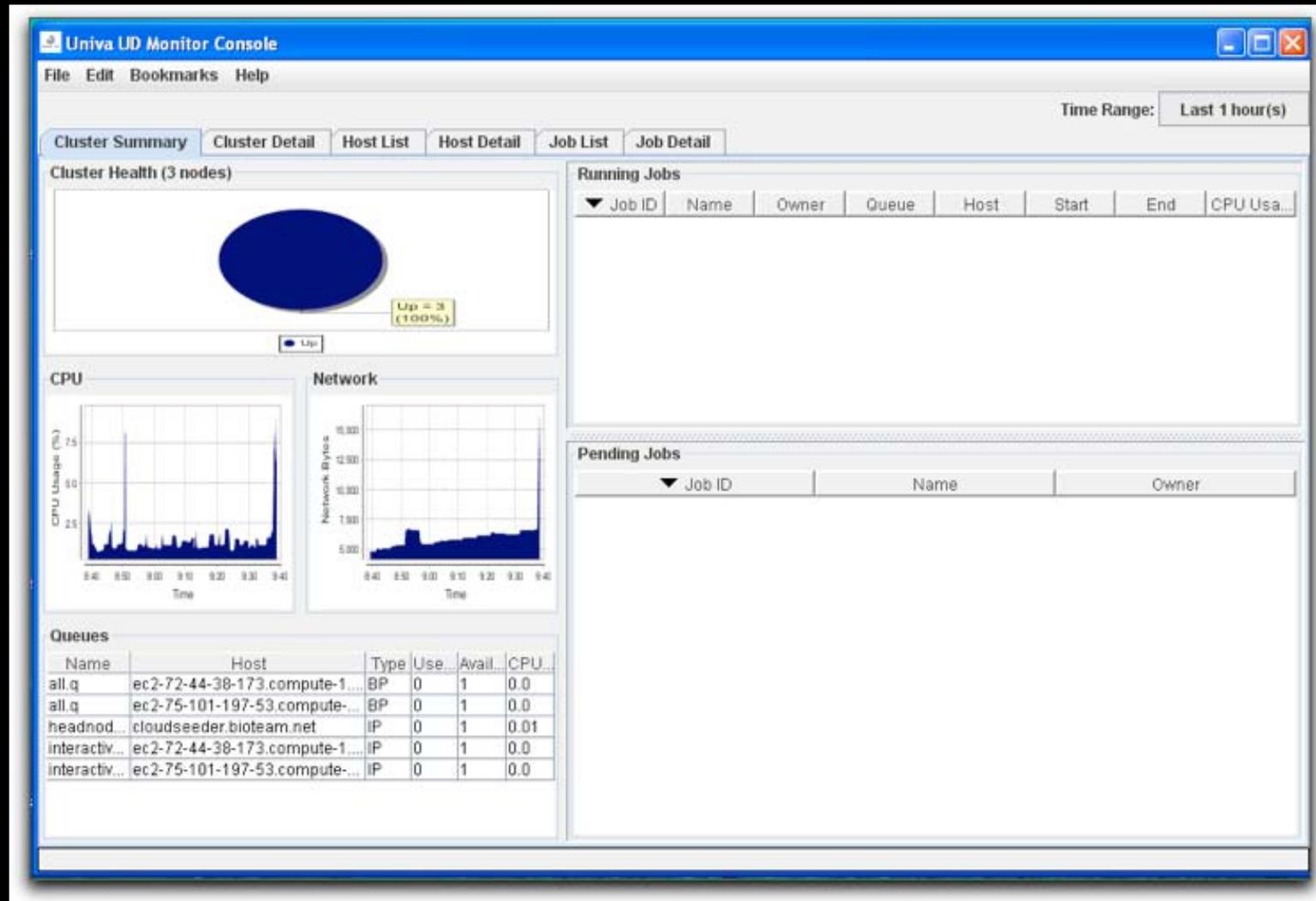
Grid Engine ARCo



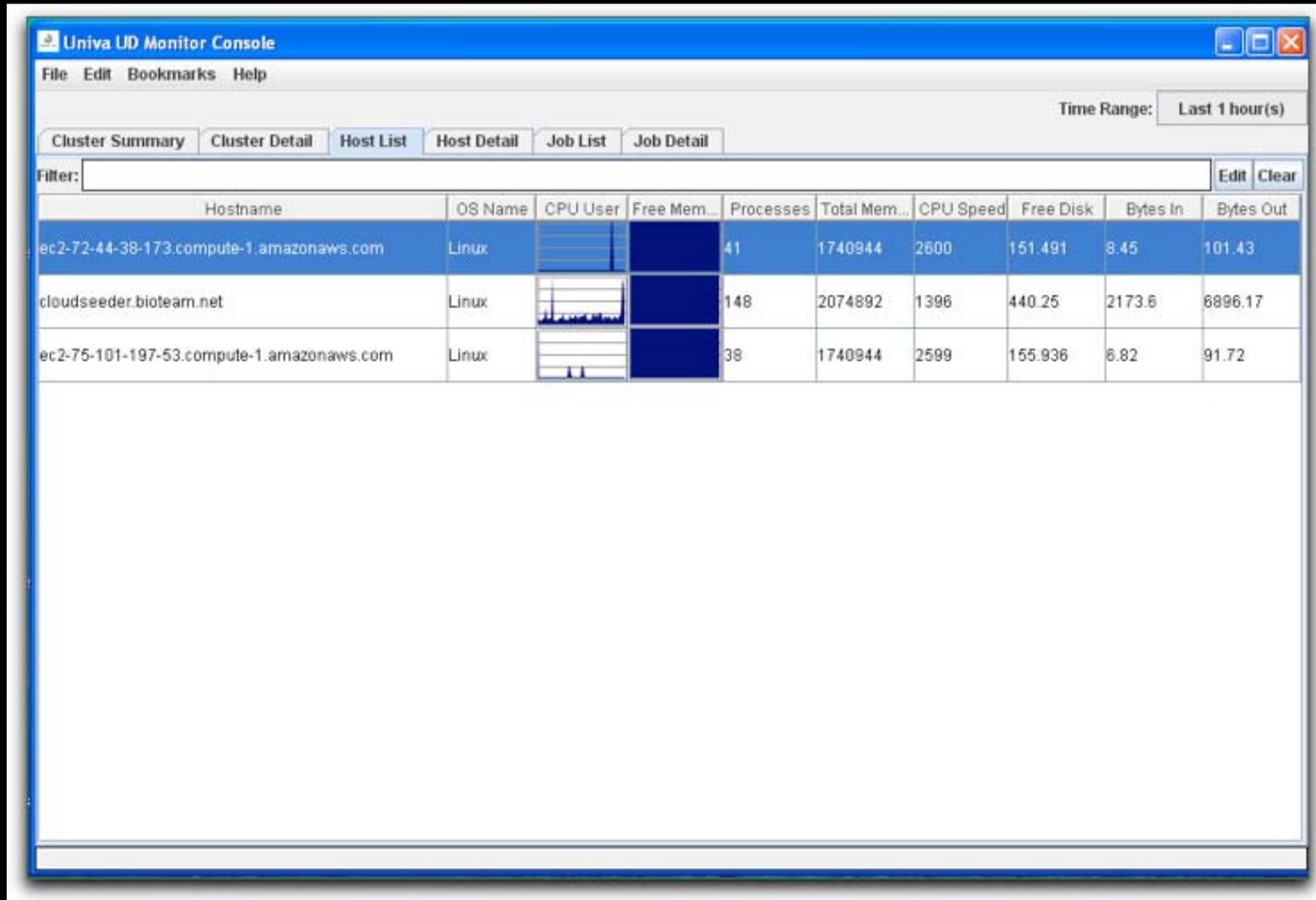
UnivaUD's SGE Reporting

- UnivaUD has a single reporting framework that combines data from:
 - Ganglia
 - SGE 'qstat'
 - SGE accounting file
 - SGE reporting file
 - SGE ARCo system
- One of the main reasons I like UniCluster
 - *{I think} This is a Windows app only so far ...*

UnivaUD SGE Monitoring



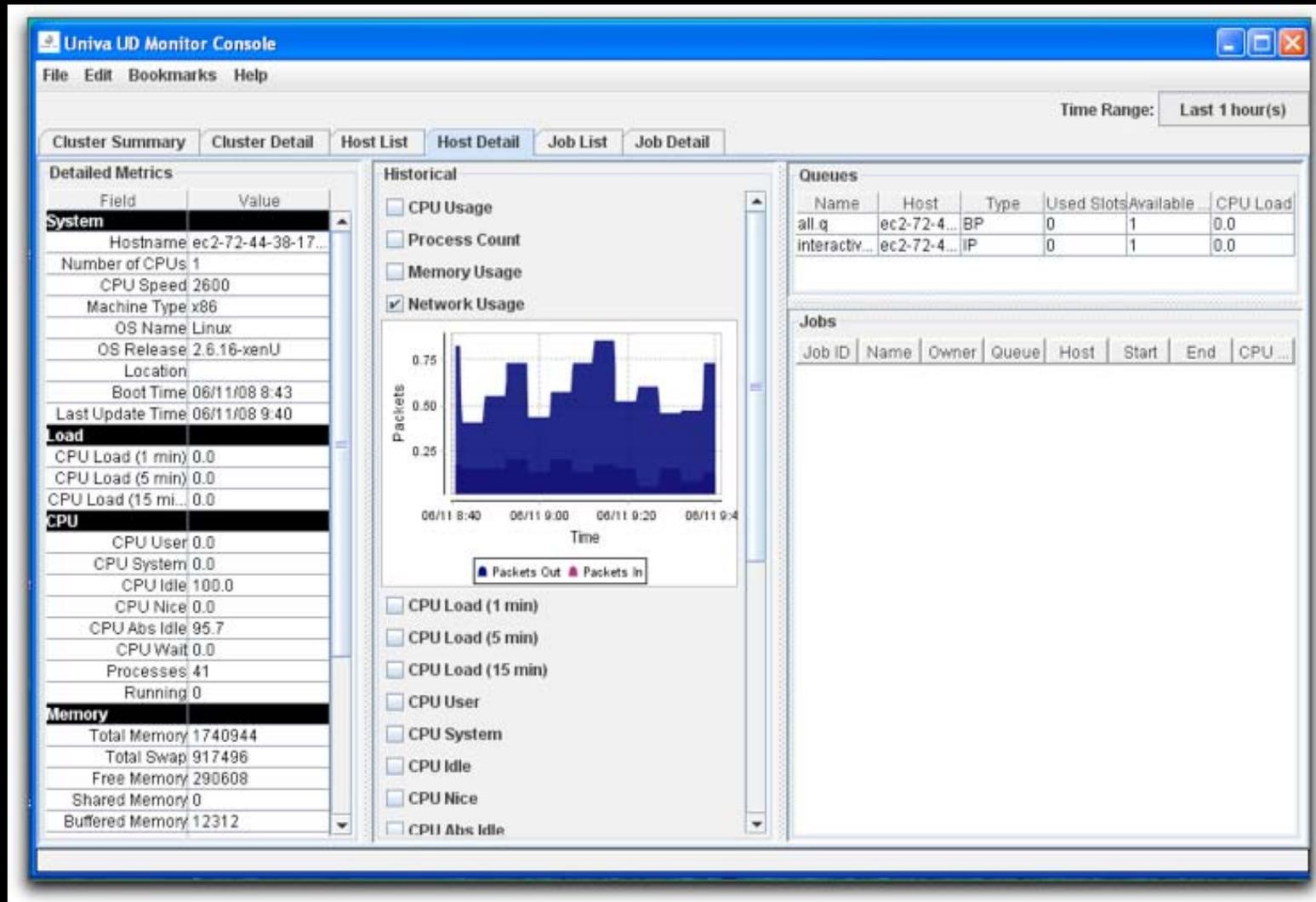
UnivaUD SGE Monitoring



The screenshot shows the Univa UD Monitor Console interface. At the top, there is a menu bar with 'File', 'Edit', 'Bookmarks', and 'Help'. Below the menu bar, there are navigation tabs: 'Cluster Summary', 'Cluster Detail', 'Host List', 'Host Detail', 'Job List', and 'Job Detail'. The 'Host List' tab is currently selected. On the right side, there is a 'Time Range' dropdown menu set to 'Last 1 hour(s)'. Below the navigation tabs, there is a 'Filter:' input field with 'Edit' and 'Clear' buttons. The main content area displays a table with the following columns: Hostname, OS Name, CPU User, Free Mem..., Processes, Total Mem..., CPU Speed, Free Disk, Bytes In, and Bytes Out. The table contains three rows of data, each with a small line graph in the 'CPU User' column.

Hostname	OS Name	CPU User	Free Mem...	Processes	Total Mem...	CPU Speed	Free Disk	Bytes In	Bytes Out
ec2-72-44-38-173.compute-1.amazonaws.com	Linux			41	1740944	2600	151.491	8.45	101.43
cloudseeder.bioteam.net	Linux			148	2074892	1396	440.25	2173.6	6896.17
ec2-75-101-197-53.compute-1.amazonaws.com	Linux			38	1740944	2599	155.936	6.82	91.72

UnivaUD SGE Monitoring



Grid Engine Scheduler Monitoring & Profiling

Scheduler Profiling

- Relatively undocumented
 - <http://gridengine.sunsource.net/source/browse/gridengine/doc/devel/rfe/profiling.txt?rev=1.1&view=markup>
- Add “`profile=1`” to the “`params`” line of the scheduler configuration
- Result
 - More profiling data added to
 - `$SGE_ROOT/$CELL/spool/qmaster/schedd/messages`

Scheduler Profiling

```
$ tail ../spool/qmaster/schedd/messages
```

```
04/17/2007 22:23:32|schedd|cd|P|PROF: job ticket calculation: init: 0.000
    s, pass 0: 0.000 s, pass 1: 0.000, pass2: 0.000, calc: 0.000 s
04/17/2007 22:23:32|schedd|cd|P|PROF: normalizing job tickets took 0.000 s
04/17/2007 22:23:32|schedd|cd|P|PROF: create active job orders: 0.000 s
04/17/2007 22:23:32|schedd|cd|P|PROF: job-order calculation took 0.000 s
04/17/2007 22:23:32|schedd|cd|P|PROF: create pending job orders: 0.000 s
04/17/2007 22:23:32|schedd|cd|P|PROF: scheduled in 0.000 (u 0.000 + s 0.000
    = 0.000): 0 sequential, 0 parallel, 2 orders, 2 H, 2 Q, 2 QA, 0 J(qw), 0
    J(r), 0 J(s), 0 J(h), 0 J(e), 0 J(x), 0 J(all), 48 C, 1 ACL, 1 PE, 2 U,
    1 D, 1 PRJ, 0 ST, 0 CKPT, 0 RU, 1 gMes, 0 jMes, 1/1 pre-send, 0/0/0 pe-
    alg
04/17/2007 22:23:32|schedd|cd|P|PROF: send orders and cleanup took: 0.010
    (u 0.000,s 0.000) s
04/17/2007 22:23:32|schedd|cd|P|PROF: schedd run took: 0.010 s (init: 0.000
    s, copy: 0.000 s, run:0.010, free: 0.000 s, jobs: 0, categories: 0/0)
```

Scheduler Monitoring

- Also relatively undocumented
 - http://gridengine.sunsource.net/nonav/source/browse/~checkout~/gridengine/doc/devel/rfe/resource_reservation.txt?content-type=text/plain
 - Man page for “`sched_conf`”
- Add “`monitor=true`” to the “`params`” line of the scheduler configuration
- Result
 - New file created
 - Not truncated or rotated
 - Location:
 - `$SGE_ROOT/$CELL/common/schedule`

Scheduler Monitoring Output

::::::::::

3127:1:STARTING:1077903416:30:G:global:license:4.000000

3127:1:STARTING:1077903416:30:Q:all.q@carc:slots:1.000000

3128:1:RESERVING:1077903446:30:G:global:license:5.000000

3128:1:RESERVING:1077903446:30:Q:all.q@bilbur:slots:1.000000

3129:1:RESERVING:1077903476:31:G:global:license:1.000000

3129:1:RESERVING:1077903476:31:Q:all.q@es-ergb01-01:slots:1.000000

::::::::::

3127:1:RUNNING:1077903416:30:G:global:license:4.000000

3127:1:RUNNING:1077903416:30:Q:all.q@carc:slots:1.000000

3128:1:RESERVING:1077903446:30:G:global:license:5.000000

3128:1:RESERVING:1077903446:30:Q:all.q@es-ergb01-01:slots:1.000000

3129:1:RESERVING:1077903476:31:G:global:license:1.000000

3129:1:RESERVING:1077903476:31:Q:all.q@es-ergb01-01:slots:1.000000

::::::::::

Scheduler Monitoring Format

<jobid>: The job id.
<taskid>: The array task id or 1 in case of non-array jobs.
<state>: One of RUNNING/SUSPENDED/MIGRATING/STARTING/RESERVING.
<start_time>: Start time in seconds after 1.1.1970.
<duration>: Assumed job duration in seconds.
<level_char>: One of {P,G,H;Q} standing for {PE,Global,Host,Queue}.
<object_name>: The name of the PE/global/host/queue.
<resource_name>: The name of the consumable resource.
<utilization> The resource utilization debited for the job.

A line "::::::::::" marks the begin of a new schedule interval.

Solaris DTRACE support ...

- SGE specific dtrace scripts & tools appeared with 6.1 distribution
- Aimed at bottleneck identification and better performance profiling
 - Could be significant
- `$SGE_ROOT/dtrace/`
- Solaris-only feature

Grid Engine Monitoring

SGE Monitoring

- Not many options
 - `qstat`
 - `qhost`
 - `qselect`
 - `qping`
 - Log files
 - Abort/Error emails

qstat

- 'qstat'
 - Best all around tool, especially with XML output
 - Good “big picture” view
 - Good targeted views
 - Resource attribute values, load report data, etc.
- If you are rolling your own tools, this is the binary to wrap

qstat: Overall Status

```
queuename                qtype used/tot. load_avg arch          states
-----
all.q@bioteam.pcc.example.org  BIP  0/2      0.14   darwin
-----
all.q@node001.cluster.private  BIP  0/2      0.00   darwin
-----
all.q@node002.cluster.private  BIP  0/2      0.10   darwin
-----
all.q@node003.cluster.private  BIP  0/2      0.05   darwin
-----
all.q@node005.cluster.private  BIP  0/2      0.02   darwin
-----
all.q@node006.cluster.private  BIP  0/2      0.00   darwin
-----
all.q@node007.cluster.private  BIP  0/2      0.06   darwin
-----
all.q@node008.cluster.private  BIP  0/2      0.01   darwin
```

qhost:

```
$ qhost
```

HOSTNAME	ARCH	NCPU	LOAD	MEMTOT	MEMUSE	SWAPTO	SWAPUS
global	-	-	-	-	-	-	-
bioteam	darwin	2	0.14	2.0G	697.0M	0.0	0.0
node001	darwin	2	0.00	1.5G	579.0M	0.0	0.0
node002	darwin	2	0.10	2.0G	630.0M	0.0	0.0
node003	darwin	2	0.04	2.0G	628.0M	0.0	0.0
node005	darwin	2	0.01	2.0G	604.0M	0.0	0.0
node006	darwin	2	0.00	2.0G	603.0M	0.0	0.0
node007	darwin	2	0.06	2.0G	604.0M	0.0	0.0
node008	darwin	2	0.01	2.0G	607.0M	0.0	0.0

qstat: Targeted resource

```
$ qstat -F ifort
```

queuename	qtype	used/tot.	load_avg	arch	states
all.q@bioteam.pcc.example.org	BIP	0/2	0.12	darwin	
gc:ifort=2					
all.q@node001.cluster.private	BIP	0/2	0.01	darwin	
gc:ifort=2					
all.q@node002.cluster.private	BIP	0/2	0.10	darwin	
gc:ifort=2					
all.q@node003.cluster.private	BIP	0/2	0.05	darwin	
gc:ifort=2					
all.q@node005.cluster.private	BIP	0/2	0.00	darwin	
gc:ifort=2					

qstat: Targeted resource, XML

```
$ qstat -F ifort -xml
```

```
<?xml version='1.0'?>
<job_info xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <queue_info>
    <Queue-List>
      <name>all.q@bioteam.pcc.example.org</name>
      <qtype>BIP</qtype>
      <slots_used>0</slots_used>
      <slots_total>2</slots_total>
      <load_avg>0.10156</load_avg>
      <arch>darwin</arch>
      <resource name="ifort" type="gc">2.000000</resource>
    </Queue-List>
    ...
    ...
  </Queue-List>
</queue_info>
</job_info>
```

'sgeinspect' GUI

- Brand new in SGE 6.2 Update 3 (beta)
 - Java GUI for:
 - Monitoring Service Domain Management ('SDM')
 - Monitoring Grid Engine Clusters
 - Queue, Host, Job views

'sgeinspect' GUI

- Looks very promising
 - Requires a JMX-enabled SGE install
 - Requires Java
- In the current form, however:
 - Can be hard to install (keystore, etc.)
 - Since 6.2u3 beta the docs in wikis.sun.com have greatly improved

'sgeinspect' GUI - SDM monitoring

The screenshot displays the SGE Inspect application window. The main content area shows the following information:

- Overview** (with checkboxes for Saved data and Details)
- PID:** 31345
- Host:** localhost
- Main class:** com.sun.grid.grm.bootstrap.JVMImpl
- Arguments:**
- JVM:** Java HotSpot(TM) 64-Bit Server VM (11.3-b02, mixed mode)
- Java Home:** /opt/jdk1.6.0_13/jre
- JVM Flags:**
- Heap dump on OOME:** disabled

Below the overview, there are statistics:

- Thread Dumps: 0
- Heap Dumps: 0
- Profiler Snapshots: 0

A **Modules** tab is active, showing a table of loaded modules:

	Name	Version	Vendor
	common	1.0u3beta	Sun Microsystems
	cloud-adapter	1.0beta	Sun Microsystems
	security	1.0u3beta	Sun Microsystems
	gridengine-adapter	1.0u3beta	Sun Microsystems

The left sidebar shows a tree view of SGE Clusters and SDM Systems, with the selected node being [hedeby] : [SYSTEM] : [cs_vm] : [vcentos-a]. The bottom-left pane shows a Services view for [cs_vm@vcentos-a] with sub-services vcentos-a, ge, and spare_pool.

'sgeinspect' GUI - SDM monitoring

The screenshot displays the SGE Inspect application window. The main content area shows the details for the 'spare_pool' service. The 'Details' tab is active, showing a table of component properties. The 'Service state' is 'RUNNING' and the 'Component state' is 'STARTED'. The 'Resources' and 'History' tabs are also visible but empty. The 'Cached resources' tab is also empty.

Component property | **Value**

Name	spare_pool
Hostname	vcentos-a
Service state	RUNNING
Component state	STARTED
JVM	rp_vm

Resources

Name	Type	Status
------	------	--------

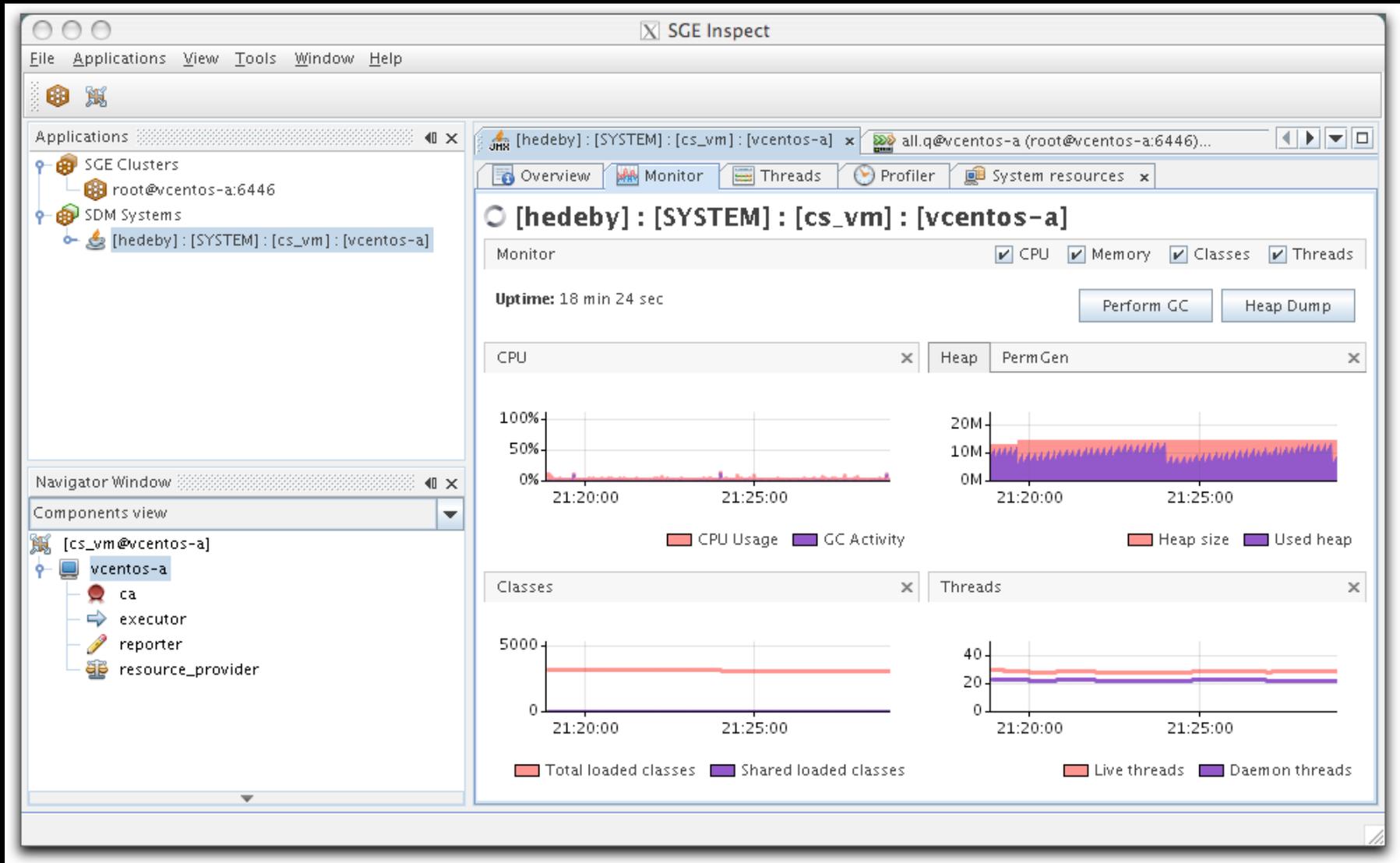
History

Date	Event type	Serv
05/06/2009...	RESOURCE_REQUEST	spare_pool
05/06/2009...	REQUEST_QUEUED	spare_pool
05/06/2009...	REQUEST_PROCESS	spare_pool

Cached resources

Name	Type	Status
------	------	--------

'sgeinspect' GUI - SDM monitoring



'sgeinspect' GUI - SGE Monitoring

The screenshot displays the 'sgeinspect' GUI for monitoring an SGE cluster. The main window is titled 'SGE Inspect' and contains several panels:

- Applications:** A tree view showing the cluster hierarchy: SGE Clusters > root@vcentos-a:6446 > SDM Systems > [hedeby]: [SYSTEM]: [cs_vm]: [vcentos-a].
- Navigator Window:** A 'Queue View' showing the queue 'all.q' under the cluster 'root@vcentos-a:6446'.
- Queue View:** Displays statistics for the 'all.q' queue:
 - Used Slots: 1
 - Available Slots: 0
 - Reserved Slots: 0
 - Total Slots: 1
- Slot Usage Graph:** A line graph showing 'Slot Usage' over time from 21:28:00 to 21:32:00. The y-axis ranges from 0 to 1. A red line represents 'Slots Used' and a blue line represents 'Slots Total'. The usage is 0 until approximately 21:30:00, then jumps to 1.
- Running Jobs Table:** A table listing the current running jobs:

ID	Job Name	Owner	Priority	Slots	Start Time	Tas
4	Sleeper	hedeby	0.555	1	Wed Ma...	
- Health Status:** A panel for 'root@vcentos-a:6446' showing:
 - Availability: 0%
 - Slot Usage: 100%
 - Overload: 45%
 - Jobs:**
 - Running: 1
 - Pending: 1
 - Finished: 3

'sgeinspect' GUI - SGE Monitoring

The screenshot displays the SGE Inspect GUI interface. The main window is titled "SGE Inspect" and contains several panels:

- Applications:** A tree view on the left showing "SGE Clusters" with "root@vcentos-a:6446" selected, and "SDM Systems" with "[hedeby]: [SYSTEM]: [cs_v...]" listed below it.
- Queue View:** A sub-panel showing "root@vcentos-a:6446" with a sub-entry for "all.q".
- Overview:** The central panel displays "root@vcentos-a:6446" and includes checkboxes for "Cluster Queue Summary" and "Host Summary".
- Cluster Queue Summary:** A table showing the status of the "all.q" queue.
- Host Summary:** A table showing the hardware and resource details for the "vcentos-a" host.
- Health Status:** A panel on the right showing "Availability: 0%", "Slot Usage: 100%", and "Overload: 47%". It also lists "Jobs: Running: 1, Pending: 1, Finished: 3".

The following tables are extracted from the GUI:

Cluster Queue Summary

Cluster Q...	Load	Used Slots	Reserved ...	Available ...	Total Slots	Temporar...	Manual In...
all.q (root...	0.47	1	0	0	1	0	0

Host Summary

Host	Arch	#CPU	Mem Us...	Mem To...	Swap U...	Swap T...	Virtual ...	Virtual ...
vcentos-a	lx24-a...	1	436.7	498.5	13	1,024	449.68	1,522....

3rd Party Monitoring Tools

- Joe's XML::Simple examples
- Qstat CGI wrappers
- xml-qstat

Perl XML::Smart Example(s)

- Provided by Joe Landman @ Scalable Informatics
- Nice, quick & simple way to get at targeted SGE state or status information
 - Especially if you know perl and don't want to get really deep into XML document handling

Perl XML::Smart Example - I

```
use XML::Smart;
my ($xml,$qstat);

$qstat=`/opt/gridengine/bin/lx24-amd64/qstat -xml`;
$xml = XML::Smart->new($qstat);

foreach ($xml->{job_info}->{queue_info}->{job_list}('@') )
{
    # stuff with each job. All the per job attributes are now available as
    # $_->{attribute_name}
    #
}
```

Perl XML::Smart Example - II

```
use XML::Smart;
my ($xml,$qstat,@jobs);

$qstat=`/opt/gridengine/bin/lx24-amd64/qstat -xml`;
$xml = XML::Smart->new($qstat);
@jobs = $xml->{job_info}->{queue_info}->{job_list}('@');

# Sort on attribute (JB_Owner in this case ...)
foreach ( sort { $a->{JB_Owner} cmp $b->{JB_Owner} } @jobs )
{
    # All the per job attributes are now available as
    # $_->{attribute_name}.
    #
}
```

Perl XML::Smart Example - III

- Deriving execution time from JAT_start_time since this value is not in XML output ...

```
use Date::Manip;
my ($d,$t,$olddate,$delta,$dt,$date);

# ... some place later in the code ...
($d,$t)=split(/\s+/, $_->{JAT_start_time} );

if ($d =~ /(\d+)\./(\d+)\./(\d+)/) {
    $date = sprintf "%.4i%.2i%.2i", $3,$1,$2; }
if ($t =~ /(\d+):(\d+):(\d+)/)
    { $date .= sprintf "%i%i%i", $1,$2,$3; }

$olddate = ParseDate($date );

$delta = DateCalc($olddate,$today);
$dt = Delta_Format($delta,0,qw(%st));
printf "%.1f second(s)\n", $dt;
```

Many sites CGI wrap qstat ...

The screenshot shows a web browser window titled "iNquiry Bioinformatics Portal" with the URL <http://workgroupcluster.apple.com/bipod/index.html>. The page features a navigation menu with links for Home, Admin, About, and Logoff. A "Welcome bioteam" message is displayed. On the left, there is a sidebar with "View Simple Forms" and a list of applications including Clustalw, EMBOSS, BioTeam, NCBI, Glimmer, HMMer, MPIBLAST, plink, Wise2, Utilities, R, and All Applications. Below this is a "Monitors" section with links for QueueStatus, Ganglia, and Links. The main content area is titled "Cluster Queue Status" and contains a table with columns: Queue (Job, N, Name), Type (User), Slots (State), Load (Date), Arch (Time), and States (SubJobs). The table lists 10 jobs with their respective details. Below the table is a "Pending Jobs" section with a similar table structure. A note at the bottom of the pending jobs section states "Auto-updated every 10 seconds." The browser's address bar and tabs are visible at the top, and the status bar at the bottom shows "Done".

Cluster Queue Status

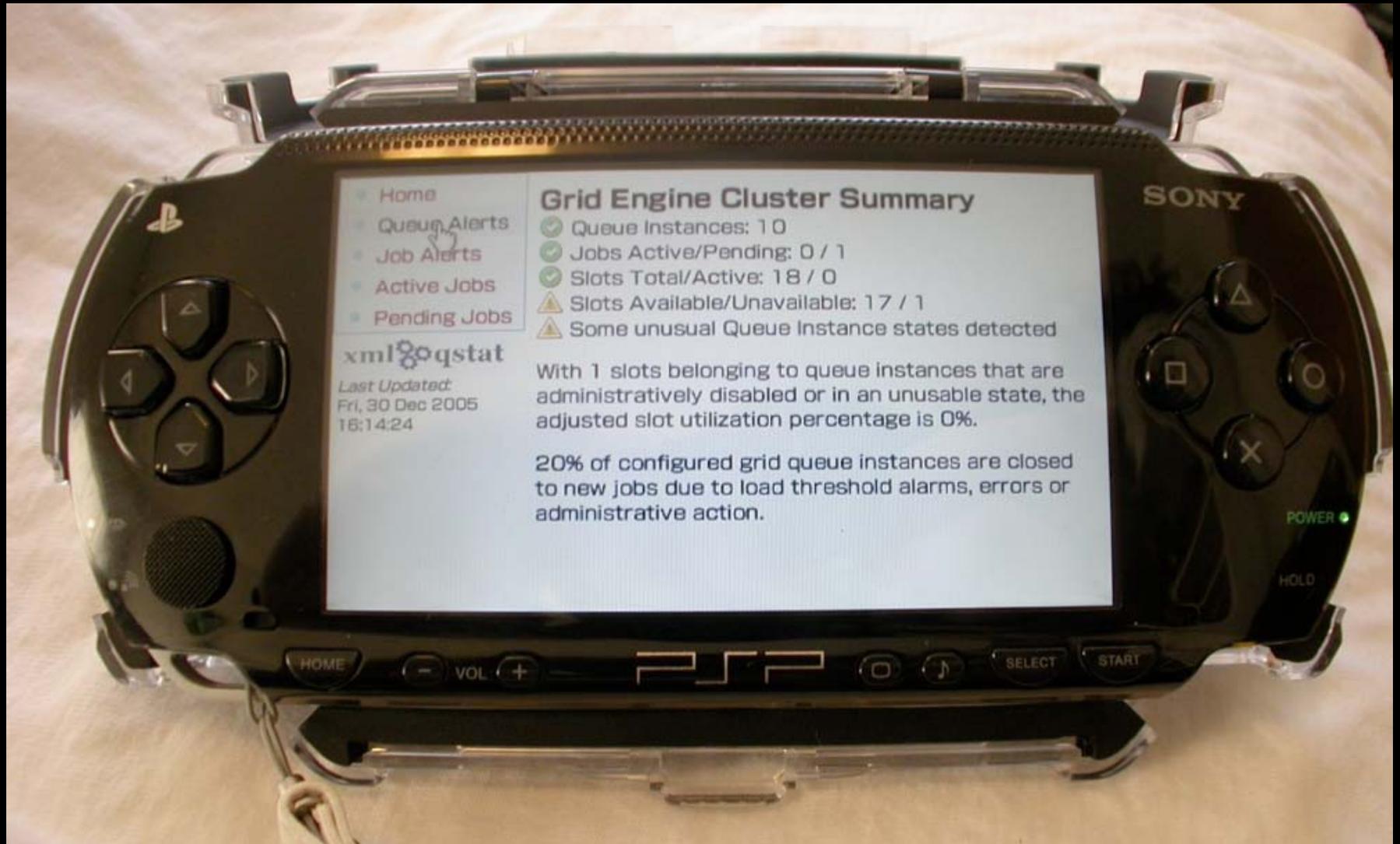
Queue			Type	Slots	Load	Arch	States
Job	N	Name	User	State	Date	Time	SubJobs
all.q@node001.cluster.private			BIP	0/2	0.22	darwin	
all.q@node002.cluster.private			BIP	0/2	0.33	darwin	
all.q@node003.cluster.private			BIP	0/2	0.29	darwin	
all.q@node004.cluster.private			BIP	0/2	0.28	darwin	
all.q@node005.cluster.private			BIP	0/2	0.23	darwin	
all.q@node006.cluster.private			BIP	0/2	0.22	darwin	
all.q@node007.cluster.private			BIP	0/2	0.26	darwin	
all.q@workgroupcluster.apple.c			BIP	0/2	0.10	darwin	
test@node001.cluster.private			BIP	0/1	0.22	darwin	

Pending Jobs

Job	N	Name	User	State	Date	Time	SubJobs
-----	---	------	------	-------	------	------	---------

Auto-updated every 10 seconds.

xml-qstat



xml-qstat

- Open source web front end to Grid Engine qstat XML output
- The XML community “approved” way to transform raw XML into useful formats
 - HTML, XHTML, Text, PDF, ...
- XML is transformed to XHTML via buzzword-compliant technology:
 - XSL, XPATH, XSLT

xml-qstat - How it works

- XML captured from Grid Engine
- Grouped with an appropriate XSL stylesheet
- Feed both XML and XSL into an XSLT engine
 - The XSL document is where the “magic” is defined
 - XSL is the language for guiding the transformation of XML from one format to another
- XML is transformed into a new format
 - In this case XHTML+CSS+DHTML for a fancy web interface
 - -or- XML RSS news feed

xml-qstat - Tech & Terminology

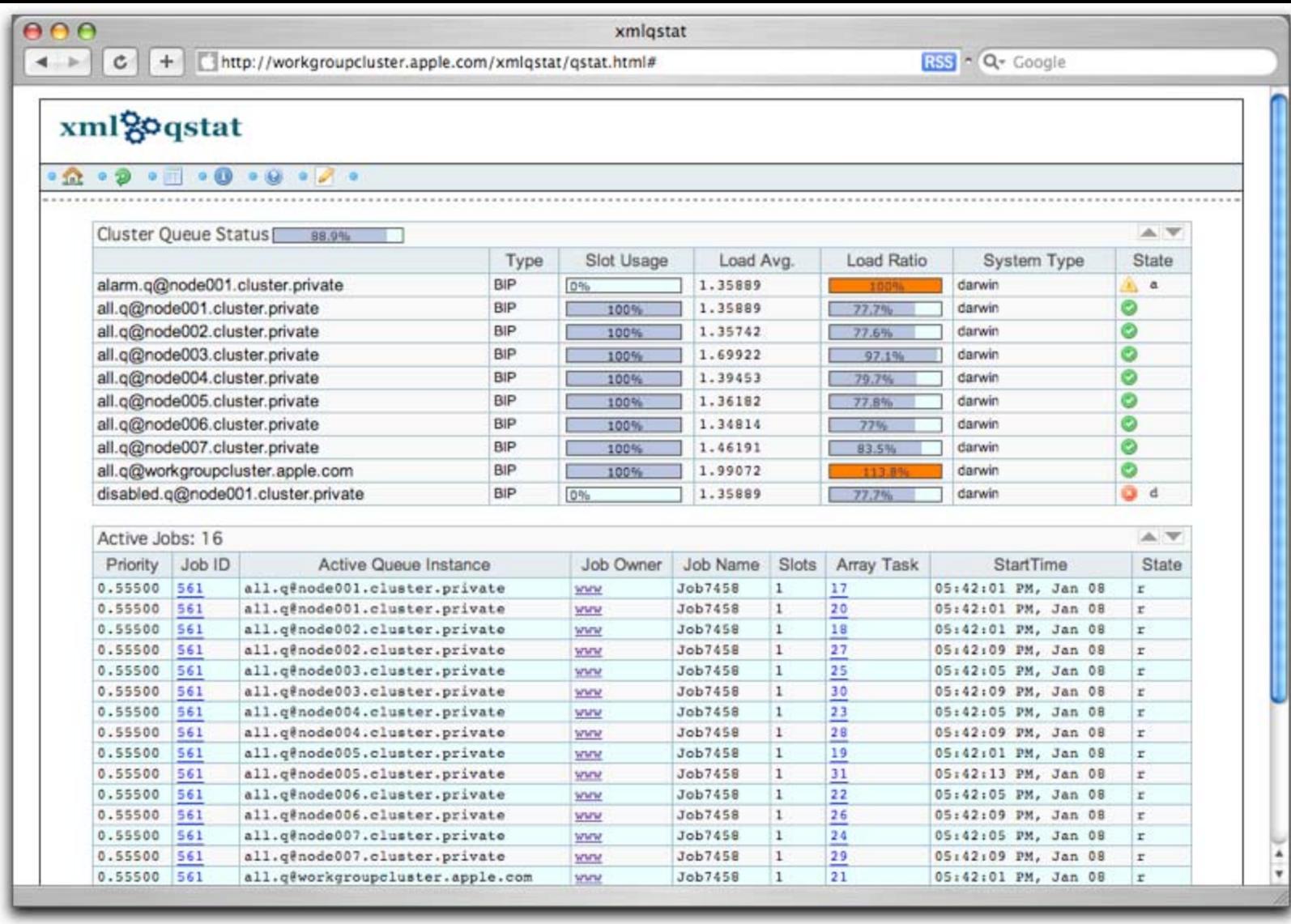
- All of these are W3C Standards:
 - XSL - Extensible Stylesheet Language
 - Format for writing stylesheets
 - XSLT - XSL Transformations
 - Rules for transforming XML documents
 - XPATH - XML Path Language
 - Query into an XML document for a particular node or attribute

xml-qstat: Technology

- Many available XSLT processing engines
 - Including FPGA accelerated hardware (!)
 - Many large institutions use hardware accelerated XSLT engines for facilitating data exchange
 - Common open source implementations:
 - Xalan-C, Xalan-C++, Xalan-Java, Saxon (java)
 - Gnome Project: libxml2, libxslt
 - Perl modules: XML::LibXML, XML::LibXSLT

xml-qstat: Technology

- xml-qstat runs under Apache Cocoon
 - <http://cocoon.apache.org>
 - Java based XML publishing framework
 - Trivial to install anywhere with a JRE
- Recommended XML/XSLT resource:
 - "Learning XSLT" by Michael Fitzgerald, 2nd ed. (2004), O'Reilly



Questions?