Grid'5000 Cheat Sheet

Text between double brackets are wiki pages.
See https://www.grid5000.fr/

For events and maintenance on platform
See https://www.grid5000.fr/status/

v0.10.1 – 2018/10/16

[[Cluster_experiment]] [[Advanced_OAR]]

Jobs states

```
oarstat
oarstat -f -j JOB_ID
oarstat -u G5K_LOGIN
```

Nodes states

```
oarnodes
oarnodes --sql "cpucore='4'"
```

Submission : Interactive

```
oarsub -I
env | grep OAR
cat $OAR_NODE_FILE
```

Reserve IPs

```
oarsub -I -l slash_22=1
g5k-subnets
```

20 nodes on griffon during 2h with 20G ib cards

```
oarsub -I -l nodes=20,walltime=2 \
  -p "cluster='griffon' and ib20G='YES'"
```

Submission : Passive

```
oarsub ~/my-script
```

5 nodes during 2h with 10G ib cards

```
oarsub -l nodes=5,walltime=2 -p "ib10G='YES'" ~/prog
cat OAR.OAR_JOB_ID.std{err,out}
```

Connection to a running job

```
oarsub -C OAR_JOB_ID
```

on a node in your reservation

```
oarsh node.fqdn
```

Submission : Reservation (passive mode)

```
oarsub -r '2011-05-16 14:20:00' \
  -l nodes=10,walltime=0:10:00 ~/my-script
```

Reservation with deploy type (interactive mode)

```
oarsub -t deploy -r '2011-05-16 14:30:00' \
  -l nodes=5,walltime=2 -p "ib10G='YES'" -n "Prog42"
```

Delete a reservation

```
oardel OAR_JOB_ID
```

Oar Grid [[Grid_experiment]]

Discovering resources

```
disco cluster_name
disco site1 site2
```

Jobs Grid stats

```
oargridstat
oargridstat GRID_JOB_ID
```

Submission : Interactive

```
oargridsub -t allow_classic_ssh \
  -w '0:20:00'CLUSTER1:rdef="/nodes=2",CLUSTER2:rdef="/nodes=3"
```

Create a node file

```
oargridstat -w -l GRID_JOB_ID | sed '/^\$/d' > ~/nodes
```

Distribute node file

```
OAR_JOB_ID=CLUSTER_JOB_ID oarcp -i \
  /tmp/oargrid/oargrid_ssh_key_LOGIN_GRID_JOB_ID~/machines \
  'head -n 1 machines':
```

Connect on first node

```
OAR_JOB_ID=CLUSTER_JOB_ID oarsh -i \
  /tmp/oargrid/oargrid_ssh_key_LOGIN_GRID_JOB_ID ' head -n 1 machines'
```

Ending

```
oargriddel GRID_JOB_ID
```

Submission : Reservation (passive mode)

```
oargridsub -t allow_classic_ssh CLUSTER1:rdef="/nodes=1",\
  CLUSTER2:rdef="/nodes=4" -s '2011-05-16 14:20:00'\
  -w '0:10:00' -p /prog42/helloworld
```

View results

```
tail -f OAR.CLUSTER_JOB_ID.std{err,out}
```

Hardware Overview [[Special:G5KHardware]]

		Nodes	Cpu Intel | AMD	Memory	Disks	GPU	Network
Grenoble							
dahu	(2018)	32	2x16cores @2.1Ghz	192Gb	223Gb SSD, 447Gb SSD, 3726Gb HDD		IB100G EDR 1x10G
yeti	(2018)	4	4x16cores @2.1Ghz	768Gb	446Gb SSD, 3x1862Gb HDD		IB100G EDR 1x10G
Lille							
chetemi	(2016)	15	2x10cores @2.2Ghz	256Gb	2x279Gb HDD		2x10G
chiclet	(2018)	8	2x16cores @2.2Ghz	128Gb	447Gb SSD, 2x3726Gb HDD		2x10G
chifflet	(2016)	8	2x14cores @2.4Ghz	768Gb	2x372Gb SSD, 2x3726Gb HDD	2xGTX 1080 Ti	2x10G
chifflot	(2018)	8	2x12cores @2.6Ghz	192Gb	2x447Gb SSD, 4x3726Gb HDD	2xTesla P100	2x10G
Luxembourg							
granduc	(2011)	22	2x4cores @2.0Ghz	16Gb	136Gb HDD		1x10G
petitprince	(2013)	16	2x6cores @2.0Ghz	32Gb	232Gb HDD		2x10G
Lyon							
hercule	(2012)	4	2x6cores @2.0Ghz	32Gb	3x1863Gb HDD		1x10G
nova	(2016)	23	2x8cores @2.1Ghz	64Gb	557Gb HDD		1x10G
orion	(2012)	4	2x6cores @2.3Ghz	32Gb	557Gb HDD	1xTesla M2075	1x10G
sagittaire	(2006)	32	2x1cores @2.4Ghz	2Gb	68Gb HDD		
taurus	(2012)	16	2x6cores @2.3Ghz	32Gb	557Gb HDD		1x10G
Nancy							
graoully	(2016)	16	2x8cores @2.4Ghz	128Gb	2x558Gb HDD		IB56G FDR 1x10G
graphene	(2011)	126	1x4cores @2.55Ghz	16Gb	298Gb HDD		IB20G DDR
graphique	(2015)	6	2x6cores @2.4Ghz	64Gb	278Gb HDD	2xTitan Black	IB56G FDR 1x10G
graphite	(2013)	4	2x8cores @2.0Ghz	256Gb	2x279Gb SSD		IB56G FDR 1x10G
grele	(2017)	14	2x12cores @2.2Ghz	128Gb	2x278Gb HDD	2xGTX 1080 Ti	IB100G EDR 1x10G
griffon	(2009)	11	2x4cores @2.5Ghz	16Gb	298Gb HDD		
grimani	(2016)	6	2x6cores @1.6Ghz	64Gb	931Gb HDD	2xTesla K40M	IB100G EDR 1x10G
grimoire	(2016)	8	2x8cores @2.4Ghz	128Gb	5x558Gb HDD, 186Gb SSD		IB56G FDR 4x10G
grisou	(2016)	51	2x8cores @2.4Ghz	128Gb	2x558Gb HDD		4x10G
grvingt	(2018)	64	2x16cores @2.1Ghz	192Gb	931Gb HDD		IB100G EDR 1x10G
Nantes							
econome	(2014)	22	2x8cores @2.2Ghz	64Gb	1863Gb HDD		1x10G
ecotype	(2017)	48	2x10cores @1.8Ghz	128Gb	372Gb SSD		2x10G
Rennes							
paranoia	(2014)	8	2x10cores @2.2Ghz	128Gb	5x558Gb HDD		2x10G
parapide	(2010)	21	2x4cores @2.95Ghz	24Gb	465Gb HDD		IB20G DDR
parapluie	(2010)	18	2x12cores @1.7Ghz	48Gb	232Gb HDD		IB20G DDR
parasilo	(2015)	28	2x8cores @2.4Ghz	128Gb	5x558Gb HDD, 186Gb SSD		2x10G
paravance	(2015)	72	2x8cores @2.4Ghz	128Gb	2x558Gb HDD		2x10G
Sophia							
suno	(2010)	45	2x4cores @2.25Ghz	32Gb	557Gb HDD		
uvb	(2011)	44	2x6cores @2.95Ghz	96Gb	232Gb HDD		IB40G QDR

API [[API_Main_Pratical]] [[API]]

API Sid

- https://api.grid5000.fr/sid/ui/index.html

Grid'5000 Nodes API

- https://api.grid5000.fr/stable/ui/nodes.html

Tutorials

- http://grid5000.github.io/tutorials/

KaVLAN [[Kavlan]]

Submission

```
oarsub -t deploy -l {"type='kavlan'"}/vlan=1+nodes=2\
walltime=2 -I
```

Deploy

```
kadeploy3 -f $OAR_NODEFILE -e env -k --vlan 'kavlan -V'
```

Find out in which vlan is a node

```
kavlan -g -m node.fqdn.fr
```

List nodes (kavlan fqdn of a reservation)

```
kavlan -l -j jobid
```

Resources

kavlan-local not routed (1..3)

kavlan routed localy (4..9)

kavlan-global routed (one per site)

* With electrical consumption. See https://helpdesk.grid5000.fr/supervision/lyon/wattmetre/

[[Deploy_environment-OAR2]] [[Advanced_Kadeploy]]

Locate a suitable image

```
kaenv3 -l
kaenv3 -l -u LOGIN
kaenv3 -p wheezy-x64-min -u deploy
```

Use deploy type for your job

```
oarsub -I -t deploy -l nodes=2
cat $OAR_NODEFILE
```

Deploy an environment

```
kadeploy3 -e wheezy-x64-base -m node.site.grid5000.fr -k
kadeploy3 -e wheezy-x64-base -f $OAR_NODEFILE -k ssh_key.pub
```

Save your deployed environment with tgz-g5k (available on gforge, or installed on environments)

```
tgz-g5k login@frontend:image.tgz (from node)
ssh root@node tgz-g5k > image.tgz (from frontend)
```

Connection to the deployed environment

```
ssh root@node.site.grid5000.fr # password "grid5000"
```

with console (useful if network doesn't work)

```
kaconsole3 -m node.site.grid5000.fr
```

Deploy and save your environment Generate a desciption file

```
kaenv3 -p wheezy-x64-base -u deploy > image.env
```

(edit file image.env to update with your values) Deploy

```
kadeploy3 -f $OAR_NODEFILE -a image.env
```

Save your image

```
kaenv3 -a image.env
```

Multi-sites deployment

```
kadeploy3 -e wheezy-x64-base -f ~/gridnodes --multi-server -k
```

Easy use with public share

```
kadeploy3 -f $OAR_NODEFILE\
-f http://public.nancy.grid5000.fr/~login/image.env -k
```

Links

DrawGantt (Nodes states in a temporal diagram)

- https://intranet.grid5000.fr/oar/site/drawgantt.cgi

Monika (Nodes states with properties)

- https://intranet.grid5000.fr/oar/site/monika.cgi

Ganglia (Nodes metrics)

- https://helpdesk.grid5000.fr/ganglia/

Grid'5000 API

- https://api.grid5000.fr/

UMS (Account, quotas extensions)

- https://api.grid5000.fr/ui/account

Grid'5000 Software

- [Grid5000:Software] on wiki.

DrawGanttGlobal

- https://www.grid5000.fr/gridstatus/oargridgantt.cgi

MonikaGlobal

- https://www.grid5000.fr/gridstatus/oargridmonika.cgi

Public share access from outside g5k (with http auth)

- https://api.grid5000.fr/sid/grid5000/sites/site/public/login/

Public share access from inside g5k

- https://public.site.grid5000.fr/~login/

Public share (populate your own public share)

- drop files in your /public/ folder (see README in there)

Restfully, g5k-campaign

- http://github.com/crohr/restfully/
- http://g5k-campaign.gforge.inria.fr/

Grid'5000 software

- https://www.grid5000.fr/mediawiki/index.php/Grid5000:Software