

Projet INFO2 :

Affichage d'un diagramme de Gantt de l'état des réservations d'un cluster

Camille Coti (LIPN), Nicolas Grenèche (DSI)
camille.coti@lipn.univ-paris13.fr
nicolas.greneche@univ-paris13.fr

Contexte

L'université Paris 13 dispose d'un système de calcul appelé *cluster*, constitué de douze nœuds équipés chacun de douze cœurs et de 24 Go de mémoire vive ainsi que d'un gros nœud équipé de quarante cœurs et de 512 Go de mémoire vive, le tout relié par un réseau GigaEthernet et un réseau rapide InfiniBand. Les utilisateurs du cluster appartiennent à toutes les composantes de l'Université. La page d'accueil du cluster se situe à l'adresse suivante : <http://www.univ-paris13.fr/calcul/wiki>.

Afin d'organiser et d'ordonnancer l'utilisation des ressources de calcul de façon exclusive par les utilisateurs, un système de gestion et d'allocation de ressources appelé SLURM¹ est installé sur la frontale du cluster. Lorsqu'un utilisateur souhaite utiliser des ressources du cluster, il doit impérativement en faire la demande auprès du système d'allocation de ressources.

Objectif

Les utilisateurs disposent d'un certain nombre d'outils pour obtenir des informations sur l'état du système. Par exemple, la commande `squeue` permet de savoir quels sont les jobs en train de tourner ou en file d'attente :

```
[camille.coti@magi1 ~]$ squeue
JOBID PARTITION   NAME     USER  ST        TIME  NODES NODELIST(REASON)
77849   B500 lna-dddd sami.eva  R   22:18:22     1 magi11
77893   B500 nodelist gilles.g  R    3:51:09     7 magi[12-18]
77559   SMP shallow. liyang47  PD         0:00     1 (Resources)
77164   SMP TestPort yanik.ng  R 13-16:55:12     1 magi10
77580   SMP E-kQKP lucas.le  R  2-04:18:31     1 magi10
77705   SMP TestPort yanik.ng  R  1-18:04:31     1 magi10
```

On peut obtenir des informations individuelles sur un nœud; par exemple :

¹Simple Linux Utility for Resource Management, <https://computing.llnl.gov/linux/slurm>

```
[camille.coti@magi1 ~]$ scontrol show node magi11
NodeName=magi11 Arch=x86_64 CoresPerSocket=1
CPUAlloc=12 CPUErr=0 CPUTot=12 Features=(null)
Gres=(null)
NodeAddr=magi11 NodeHostName=magi11
OS=Linux RealMemory=1 Sockets=12
State=ALLOCATED ThreadsPerCore=1 TmpDisk=0 Weight=1
BootTime=2012-11-21T12:13:24 SlurmdStartTime=2013-01-08T13:54:25
Reason=(null)
```

On peut également obtenir des informations plus détaillées sur un job en cours :

```
[camille.coti@magi1 ~]$ scontrol show job 77849
JobId=77849 Name=lna-dddd
UserId=sami.evangelista(6011) GroupId=sami.evangelista(6011)
Priority=4294825665 Account=(null) QOS=normal
JobState=RUNNING Reason=None Dependency=(null)
Queue=1 Restarts=0 BatchFlag=1 ExitCode=0:0
RunTime=1-06:15:04 TimeLimit=UNLIMITED TimeMin=N/A
SubmitTime=2013-02-05T15:08:02 EligibleTime=2013-02-05T15:08:02
StartTime=2013-02-05T15:08:02 EndTime=Unknown
PreemptTime=None SuspendTime=None SecsPreSuspend=0
Partition=B500 AllocNode:Sid=magi1:31189
ReqNodeList=(null) ExcNodeList=(null)
NodeList=magi11
BatchHost=magi11
NumNodes=1 NumCPUs=12 CPUs/Task=12 ReqS:C:T=*:*:~
MinCPUsNode=12 MinMemoryNode=0 MinTmpDiskNode=0
Features=(null) Gres=(null) Reservation=(null)
Shared=OK Contiguous=0 Licenses=(null) Network=(null)
Command=/glusterfs/home-volume/sami.evangelista/svn/sami/helena/benchs/run
-delta-ddd.sh
WorkDir=/glusterfs/home-volume/sami.evangelista/svn/sami/helena/benchs
```

Cependant, cette approche a pour défaut de manquer de convivialité et rebute souvent des utilisateurs qui ne sont pas issus de la communauté informatique. Un affichage graphique serait plus visuel et permettrait aux utilisateurs de se faire une idée précise de l'état du cluster en un seul regard.

Il est notamment particulièrement intéressant pour les utilisateurs, lorsque le cluster est utilisé par d'autres et qu'il n'y a pas suffisamment de place pour que leurs jobs soient exécutés immédiatement, de pouvoir voir où a été positionné leur job et d'ainsi estimer quand leur job pourra passer.

Détail du travail demandé

Les informations de soumission de Slurm sont enregistrées dans une base de données. Des requêtes SQL permettent donc d'y accéder. On peut ainsi obtenir les soumissions en cours d'exécution, mais également les soumissions en attente.

Il sera demandé de développer une application Web qui, à partir de ces informations, dessine un diagramme de Gantt représentant en ordonnée les cœurs disponibles sur le cluster, et en abscisse le temps. Les jobs soumis auprès de Slurm seront représentés par des rectangles de couleurs différentes afin de les différencier les uns des autres. Enfin, lorsqu'un rectangle sera survolé par la souris, les informations détaillées sur le job concerné seront affichées.

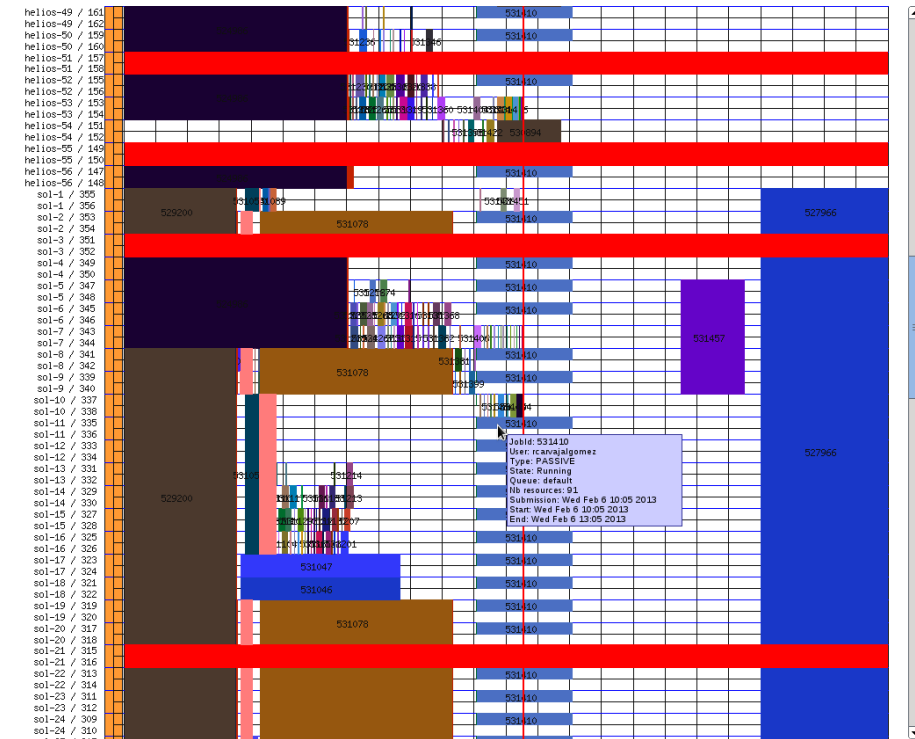


Figure 1: Diagramme de Gantt d'OAR

Une application de ce type est disponible dans l'outil OAR². La figure 1 présente une capture d'écran de l'outil fourni par OAR.

Connaissances requises

Programmation web, HTML, php ou Python, bases de Javascript (optionnel), SQL.

Informations administratives

| | | |
|---------------|-----------------------------------|----------------------------------|
| Nom : | Camille Coti | Nicolas Grenèche |
| Coordonnées : | Bureau A107, LIPN | Bureau A003, DSI |
| Email : | camille.coti@lipn.univ-paris13.fr | nicolas.greneche@univ-paris13.fr |
| Téléphone : | 01 49 40 40 83 | 01 49 40 40 35 |

²<http://oar.imag.fr>