Les systèmes de batch

Formation LEM2I, Montage et gestion d'un centre de calcul

Bruno Bzeznik

CIMENT, UJF

Alger, 13/09/2011

Sommaire

1. Généralités
2. Du processus à la grille, et même au delà
   - Grappe de calcul
   - Grille de calcul
   - Informatique dans le nuage
   - Grappe, Grille, Cloud : récapitulons
3. Les RJMS
   - Caractéristiques
   - Quelques RJMS
4. Fonctionnement
   - Les jobs
   - Les ressources
   - Politiques d'ordonnancement
5. Visualisation

Outline

1. Généralités
2. Du processus à la grille, et même au delà
   - Grappe de calcul
   - Grille de calcul
   - Informatique dans le nuage
   - Grappe, Grille, Cloud : récapitulons
3. Les RJMS
   - Caractéristiques
   - Quelques RJMS
4. Fonctionnement
   - Les jobs
   - Les ressources
   - Politiques d'ordonnancement
5. Visualisation

HPC

High Performance Computing (Calcul Intensif), definit par :

Infrastructures :

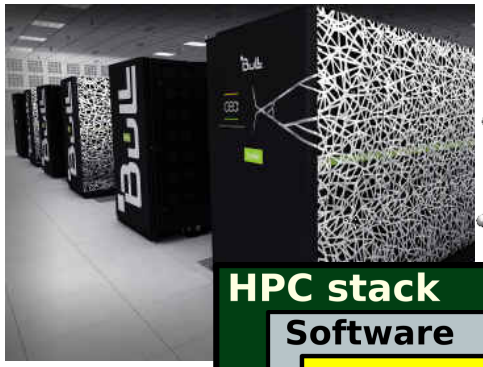
- Supercomputers, Clusters, Grids, Peer-to-Peer Systems and lately Clouds

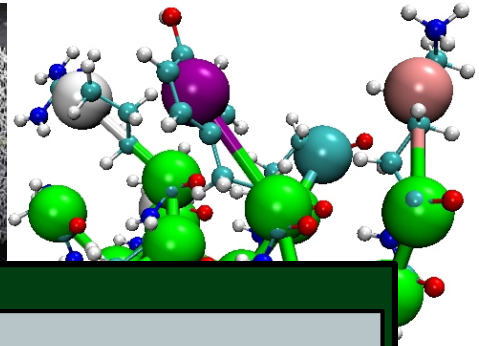
Applications :

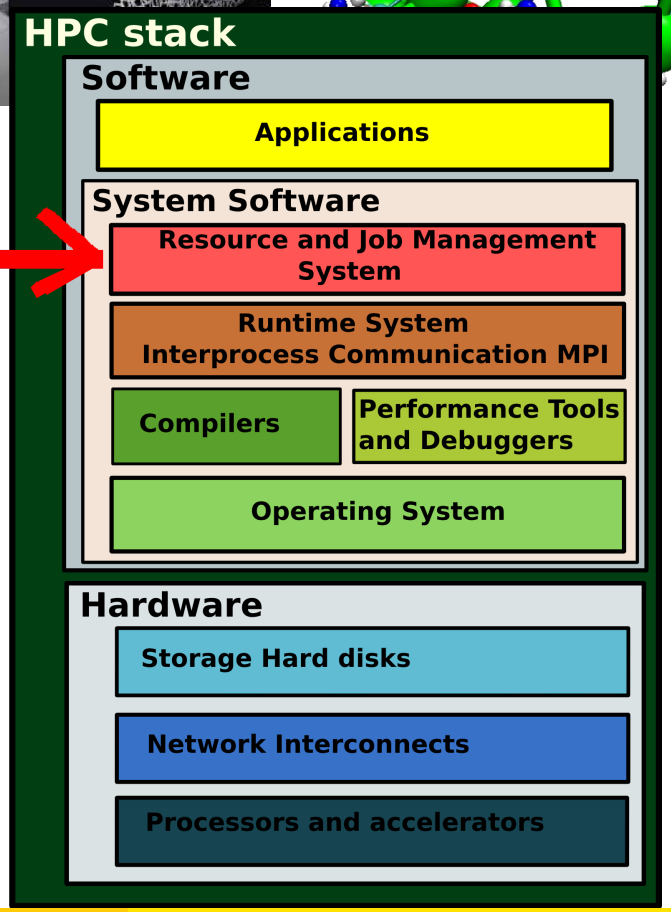
- Climate Prediction, Protein Folding, Crash simulation, High-Energy Physics, Astrophysics, Animation for movie and video game productions

System Software

- System Software : Operating System, Runtime system, Resource Management, I/O Systems, Interfacing to External Environments







Les gestionnaires de tâches et de ressources

- Différentes appellations :
  - Resource and Job Management Systems (RJMS)
  - Batch Schedulers
  - Systèmes de batch
  - Abusivement : Scheduler ou Ordonnanceur (l'ordonnanceur est un des multiples composants d'un RJMS)
- Contexte des "grappes" (clusters)
- Pour une "grille" ou un "cloud", on parle de "Middleware" (Intergiciel)

Bon, d'accord...

...mais finalement, c'est quoi une grappe, une grille ou un cloud ?

Outline

1 Généralités

2 Du processus à la grille, et même au delà
- Grappe de calcul
- Grille de calcul
- Informatique dans le nuage
- Grappe, Grille, Cloud : récapitulons

3 Les RJMS
- Caractéristiques
- Quelques RJMS

4 Fonctionnement
- Les jobs
- Les ressources
- Politiques d'ordonnancement

5 Visualisation

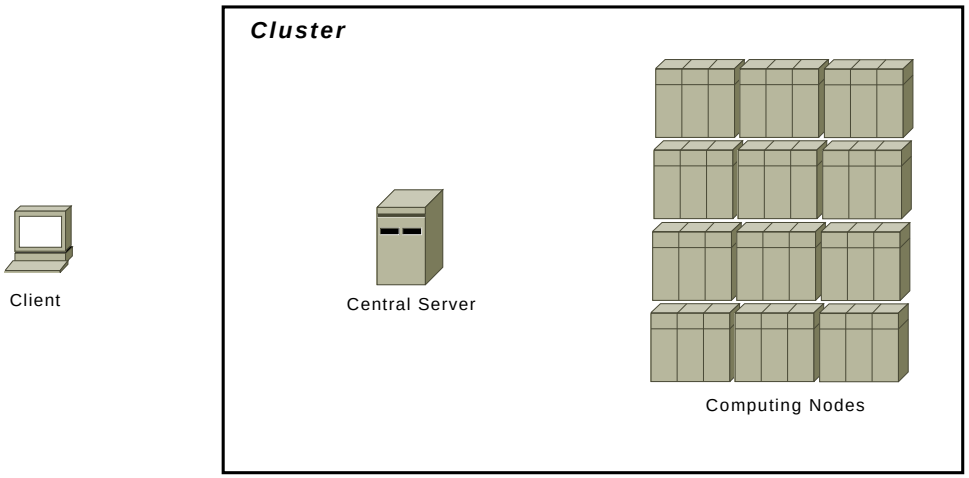
Outline

1. Généralités
2. Du processus à la grille, et même au delà
   - Grappe de calcul
   - Grille de calcul
   - Informatique dans le nuage
   - Grappe, Grille, Cloud : récapitulons
3. Les RJMS
   - Caractéristiques
   - Quelques RJMS
4. Fonctionnement
   - Les jobs
   - Les ressources
   - Politiques d'ordonnancement
5. Visualisation

Definition

Grappe / Cluster

Dans notre contexte (HPC), une grappe est un ensemble de n *noeuds* qui sont interconnectés de manière à permettre l'éxécution simultanée de plusieurs *jobs séquentiels* ou *parallèles*. Une grappe peut aussi être appellée *parallel supercomputer* (super-ordinateur parallèle).



Grappe de calcul

Processus

Les processus tournent sur les CPUs.

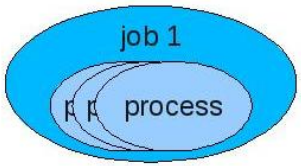
- Un processus est un programme qui est chargé en mémoire et qui est en cours d'éxécution
- Sous UNIX, plusieurs processus peuvent tourner sur un ou plusieurs processeur (multi-tâche). Ils sont hiérarchiques et appartiennent à un utilisateur particulier.
- Chaque processus UNIX possède un identifiant unique appellé le PID.

Grappe de calcul

Jobs

Les processus peuvent être grouppés en jobs.

- Un *job* peut être un processus, un groupe de processus ou encore un batch.
- Dans notre contexte (HPC clusters), un job est un ensemble de processus qui ont été automatiquement lancés par un gestionnaire de tâches, via la soumission d'un script utilisateur.
- Un job peut donner N instances d'un même programme sur N noeuds ou processeurs d'une grappe.

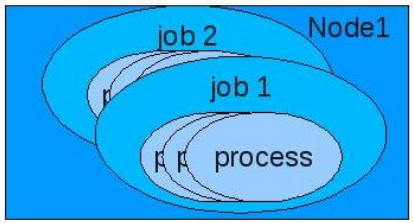


Grappe de calcul

Noeuds

Les jobs tournent sur les noeuds.

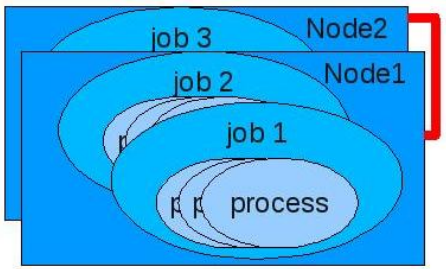
- Un *noeud* est un ordinateur qui possède p CPU, un certain montant de mémoire, une ou plusieurs interfaces réseau et qui peut avoir une unité de stockage locale (disque ou ssd).



Grappe de calcul

Réseau de calcul

Les noeuds sont interconnectés via un réseau de calcul, en général un réseau à faible latence (Myrinet, Infiniband, Numalink,...) mais cela peut être un simple réseau gigabit-ethernet pour les grappes les plus modestes.

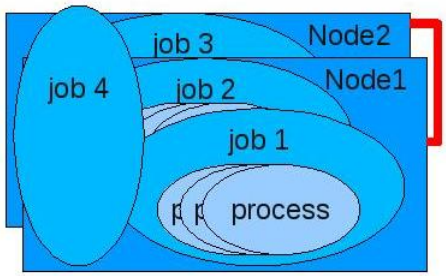


Grappe de calcul

Types de jobs

Les jobs peuvent être ”parallèles” ou ”séquentiels”. Un job parallèle tourne sur plusieurs noeuds, exploitant le réseau de calcul pour communiquer entre les noeuds.

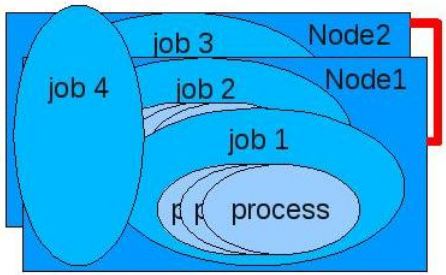
- un *Job Sequentiel* est un processus unique qui tourne sur un processeur unique sur un unique noeud.
- *Job Parallèle* : Plusieurs processus ou threads qui peuvent communiquer via une librairie spécifique (MPI, openMP, threads,...). On distingue les jobs parallèles à ’mémoire partagée’ (ils tourne sur un unique noeud multiprocesseur) et les jobs parallèles à ’mémoire distribuée’ (ils peuvent exploiter



Grappe de calcul

Attention

Un job qui lance plusieurs processus indépendants (qui ne communiquent pas) n'est pas considéré comme un job "parallèle". C'est un ensemble de jobs séquentiels. On parle aussi de jobs **embarrassingly parallel** (mais ces derniers peuvent être des ensembles de jobs parallèles qui ne communiquent pas entre eux !)



Grappe de calcul

Types de jobs

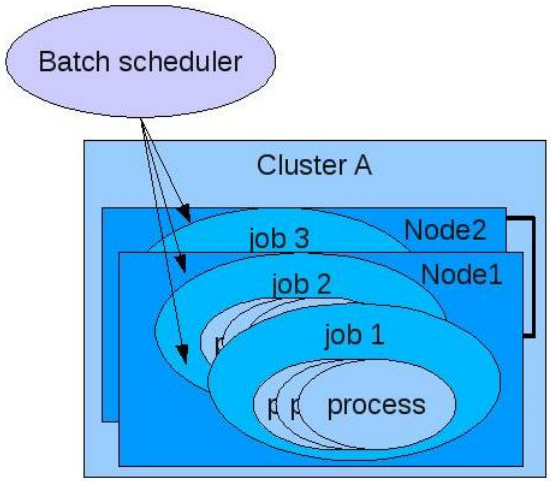
Les jobs peuvent être du type **Batch** ou **Interactif**

- **batch**
  - Un job de type *batch* est un script shell : une liste de commandes shell à éxécuter dans un ordre donné, inscrite dans un fichier.
  - Les shells d'aujourd'hui sont si sophistiqués que vous pouvez créer de véritables programmes avec des variables, des contrôles de structure et des boucles.
  - Les scripts peuvent aussi être des programmes écrits dans un language interprété (perl, php, python, ruby,...)
- **interactif**
  - Un job *interactif* est une allocation d'un ou plusieurs noeuds à la suite de laquelle l'utilisateur obtient un shell interactif sur l'un des noeuds.
  - Les jobs interactifs sont généralement utilisés pour la mise au point et le débug
  - Les jobs interactifs peuvent avoir des contraintes différentes (temps limité, nombre maxmimum de resources allouées,...)

Grappe de calcul

Gestionnaire de tâches et de ressources

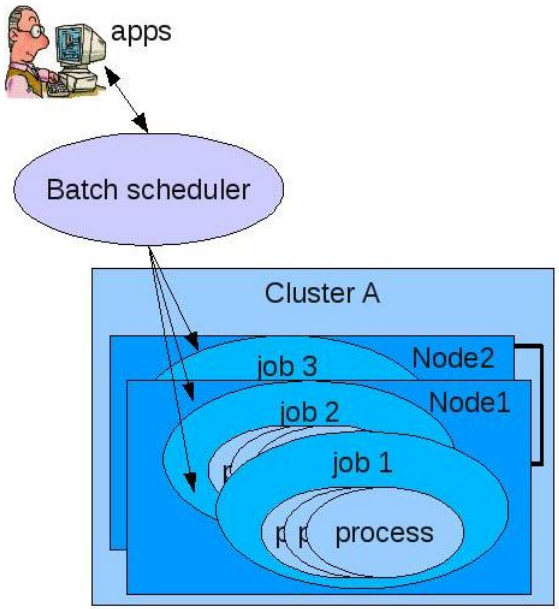
Dans notre contexte (HPC), le *gestionnaire de tâche et de ressources* (ou Resource and Job Management System ou RJMS ou Batch Scheduler) est un logiciel qui est responsable de la distribution de la puissance de calcul aux jobs utilisateurs au sein d'une infrastructure de calcul parallèle.



Grappe de calcul

Soumission de job

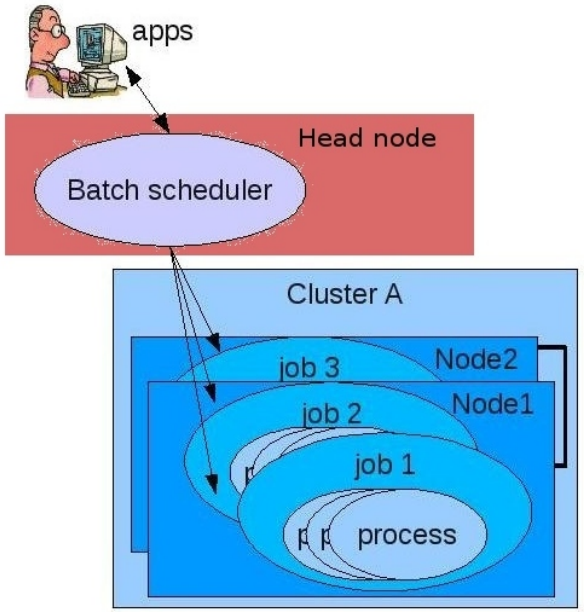
Les utilisateurs soumettent leurs jobs au gestionnaire de tâches et de ressources, qui en retour les informe sur l'état de leurs jobs.



Grappe de calcul

Frontale de soumission

La machine depuis laquelle les jobs sont soumis est appellée **frontale de soumission**, ou **noeud maître**, ou encore **head node**. En général, cette machine est accessible par les utilisateurs (souvent via *ssh*) et possède les mêmes répertoires personnels que les noeuds de calcul. Elle offre un environnement de mise au point des scripts et fourni des outils de soumission et de suivi des jobs. Parfois, elle permet aussi la compilation des programmes, mais il est en général rigoureusement **INTERDIT** de lancer des processus de calcul sur cette machine !



Outline

1. Généralités
2. Du processus à la grille, et même au delà
   - Grappe de calcul
   - Grille de calcul
   - Informatique dans le nuage
   - Grappe, Grille, Cloud : récapitulons
3. Les RJMS
   - Caractéristiques
   - Quelques RJMS
4. Fonctionnement
   - Les jobs
   - Les ressources
   - Politiques d'ordonnancement
5. Visualisation

Le concept de grille

- Vient du concept de grille de transmission de l'électricité
- Dans un réseau electrique, il y a des sources d'énergie et des consommateurs finaux qui ne savent pas forcément d'où vient l'énergie qu'ils consomment.
- Dans une grille de calcul, il y a des calculateurs et des utilisateurs finaux qui ne savent pas forcément où s'éxécutent leurs calculs.

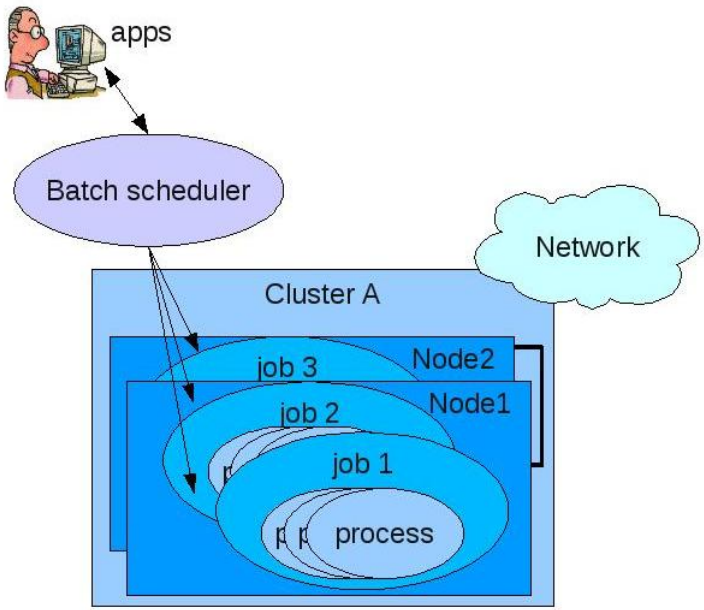
Le concept de grille

- C'est bien beau, mais...
- Les tâches de calcul peuvent être plus compliquées qu'un simple flux électrique
  - Dépendance du code de l'application
  - Dépendance avec les données d'entrée/sortie
  - Volume de données d'entrée/sortie
  - Durée
  - Type de code : parallèle/sequentiel
  - ...

Grille de calcul

Réseau public

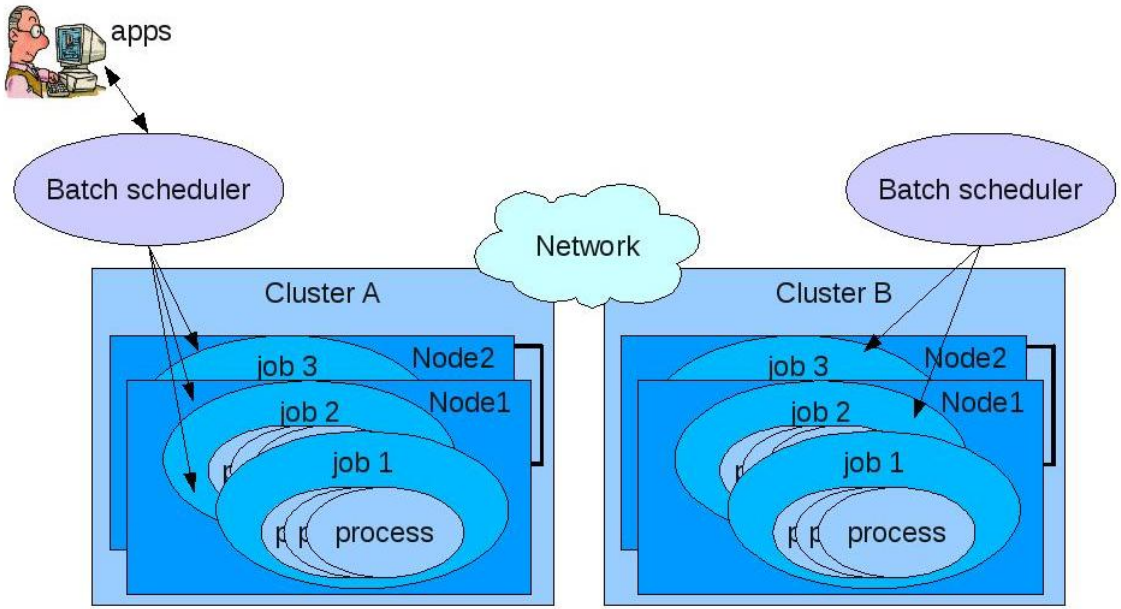
La frontale d'une grappe peut être connectée à un réseau public, en général pas le même réseau que le réseau de calcul qui, lui, est souvent privé.



Grille de calcul

Public network

On peut ainsi avoir plusieurs grappes de calcul qui sont interconnectées via leur frontale de soumission.

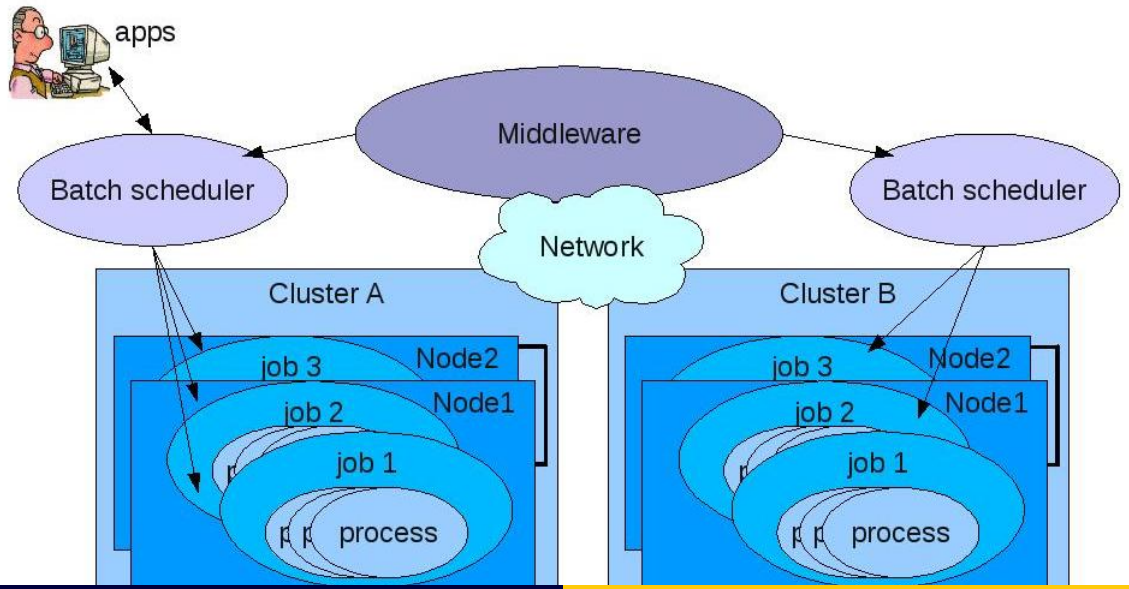


Grille de calcul

Grilles de calcul

Les grilles de calcul peuvent être composées de grappes faiblement couplées et géographiquement dispersées avec parfois des règles d'administration différentes.

- Un logiciel (ou un ensemble de logiciels) appellé *intergiciel de grille (grid middleware)*, est utilisé pour la surveillance, la découverte et la gestion des ressources pour permettre l'éxécution des applications au niveau de la grille
- A ce niveau, une collaboration entre les RJMS des grappes locales et l'intergiciel de grille est nécessaire.



Grilles de calcul

Intergiciel (Grid middleware)

L'intergiciel est donc le composant qui agit entre les différentes ressources de la grille et les applications des utilisateurs

- Il peut être très complexe et composé d'éléments très spécifiques à un type de grille donné
- L'intergiciel de grille peut donner un accès uniforme à des ressources hétérogènes
- Il gère et alloue les ressources de la grille à un niveau global (disponibilité des grappes, charge et propriétés, topologie du stockage,...)
- Il gère des problématiques d'authentification et de confidentialité
- Il peut offir des outils de visualisation et de surveillance
- Exemples : Globus, UNICORE, gLite, CiGri...

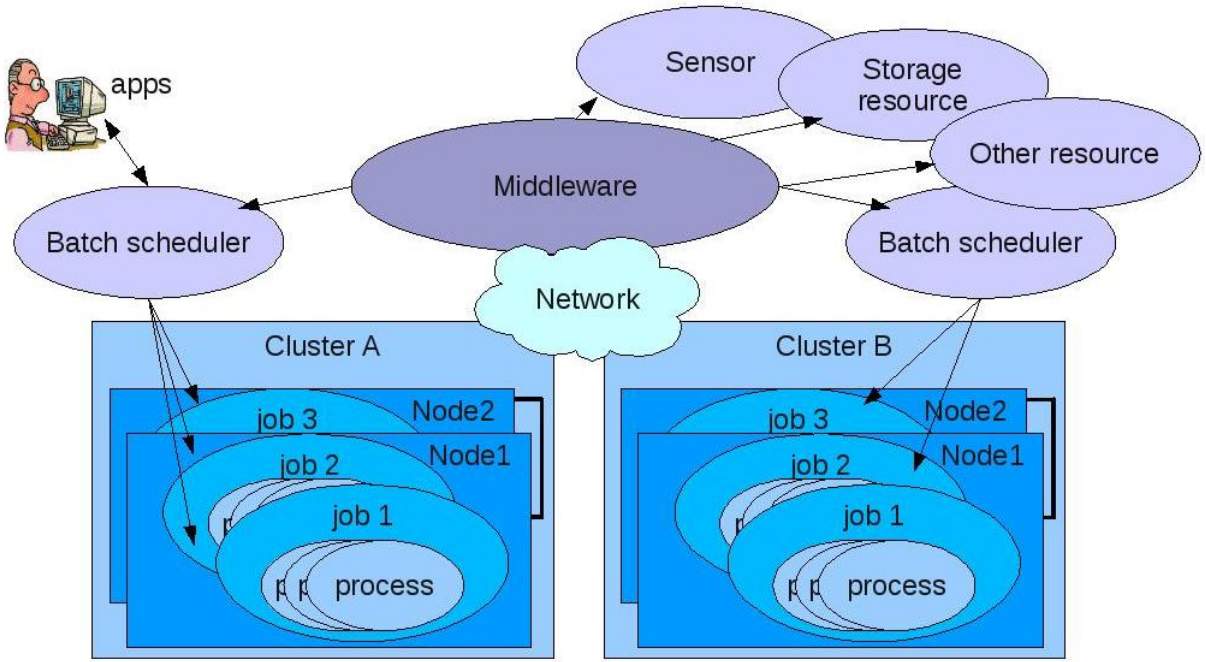
Un intergiciel de grille n'est pas un gestionnaire de tâches et de ressources !

... pour les raisons évoquées ci-dessus, mais aussi parcequ'il n'a pas forcément une vision fine de chaque ressource de calcul de la grille, mais plutôt une vision agrégée, et il s'appuie sur les RJMS pour la gestion des jobs au niveau local.

Grilles de calcul

Intergiciel (Grid middleware)

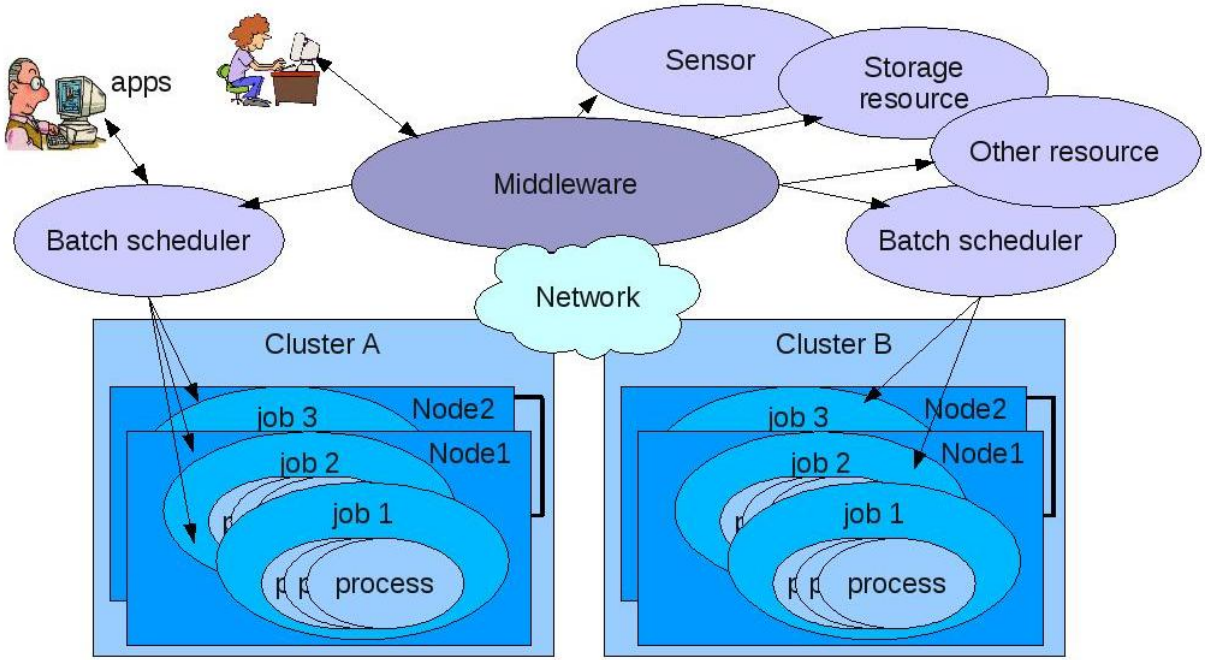
L'intergiciel peut aussi gérer la communication avec d'autres éléments, comme des capteurs ou des systèmes de stockage.



Grilles de calcul

Soumission des jobs grille

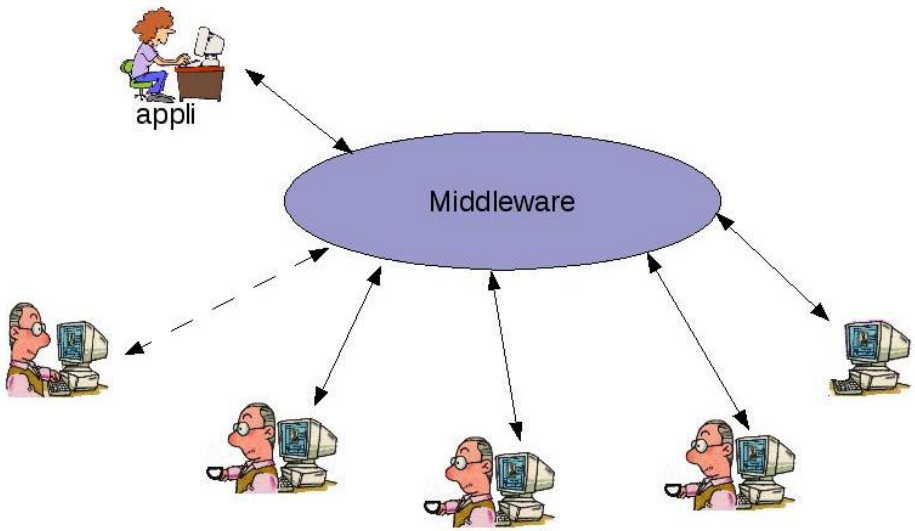
Un utilisateur de la grille interragit avec l'intergiciel, en particulier pour soumettre ses jobs.



Grilles de calcul alternatives

Desktop/volonteer computing

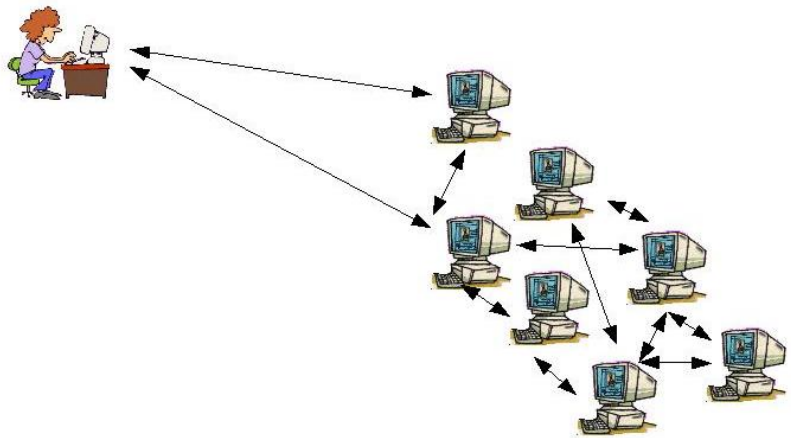
Une grille peut aussi ressembler à cela...



Grilles de calcul alternatives

Peer-to-peer grid

...ou à cela...



Outline

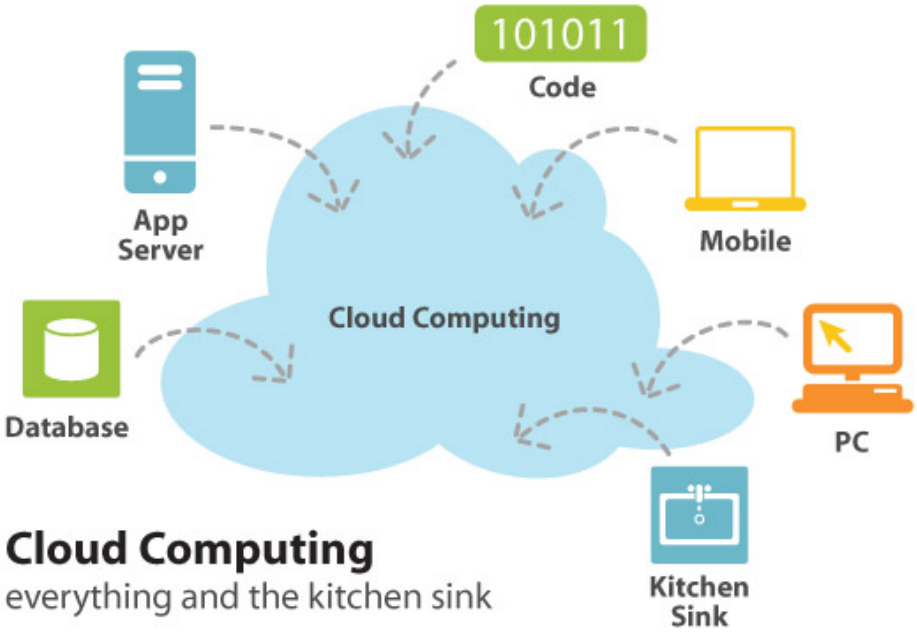
1. Généralités
2. Du processus à la grille, et même au delà
   - Grappe de calcul
   - Grille de calcul
   - Informatique dans le nuage
   - Grappe, Grille, Cloud : récapitulons
3. Les RJMS
   - Caractéristiques
   - Quelques RJMS
4. Fonctionnement
   - Les jobs
   - Les ressources
   - Politiques d'ordonnancement
5. Visualisation

Definition : ”cloud”

Cloud computing

C’est un terme générique pour désigner tout ce qui délivre des services hébergés sur Internet. D’après la définition du NIST, les services sont identifiés dans 3 catégories :

- Infrastructure-as-a-Service (IaaS)
- Platform-as-a-Service (PaaS)
- Software-as-a-Service (SaaS)



Definition : ”cloud”

Concept

Un service ”cloud” a 3 caractéristiques distinctes :

- Il est vendu **à la demande** (à la minute ou à l'heure,...)
- Il est **élastique** – un utilisateur peut avoir plus ou moins de ce service à certains moments (intérêt de la mutualisation)
- et le service est entièrement **géré par le fournisseur** (le consommateur a juste besoin d'un ordinateur personnel avec un accès à internet)

Cloud computing

- L'idée est que vous pouvez utiliser une application ou gérer des données à travers de services sans savoir où elles se trouvent (quelquepart dans le nuage)
- Il y a un lien très fort avec un modèle économique où les clients paient pour un service sans se soucier de l'infrastructure.
- Le cloud est directement lié à la grille et à la virtualisation (vous pouvez louer un système d'exploitation qui tourne quelquepart dans le nuage)
- La notion de flexibilité est aussi très importante : l'infrastructure peut s'adapter très rapidement à ce dont vous avez besoin (un jour 2 serveurs, le lendemain 10)

Outline

1. Généralités
2. Du processus à la grille, et même au delà
   - Grappe de calcul
   - Grille de calcul
   - Informatique dans le nuage
   - Grappe, Grille, Cloud : récapitulons
3. Les RJMS
   - Caractéristiques
   - Quelques RJMS
4. Fonctionnement
   - Les jobs
   - Les ressources
   - Politiques d'ordonnancement
5. Visualisation

Grappe, Grille, Cloud : récapitulons

- **Grappe** : des ressources (CPU cores), des tâches (jobs) – Un *RJMS*
- **Grille** : des ressources (grappes, pc, stockage), des tâches ou ensembles de tâches – Un *Intergiciel de grille*
- **Cloud** : des fournisseurs, des services – Un *Intergiciel de cloud*

Dans la suite, nous ne parlerons que de **Grappes de calcul**.

Outline

1 Généralités

2 Du processus à la grille, et même au delà
- Grappe de calcul
- Grille de calcul
- Informatique dans le nuage
- Grappe, Grille, Cloud : récapitulons

3 Les RJMS
- Caractéristiques
- Quelques RJMS

4 Fonctionnement
- Les jobs
- Les ressources
- Politiques d'ordonnancement

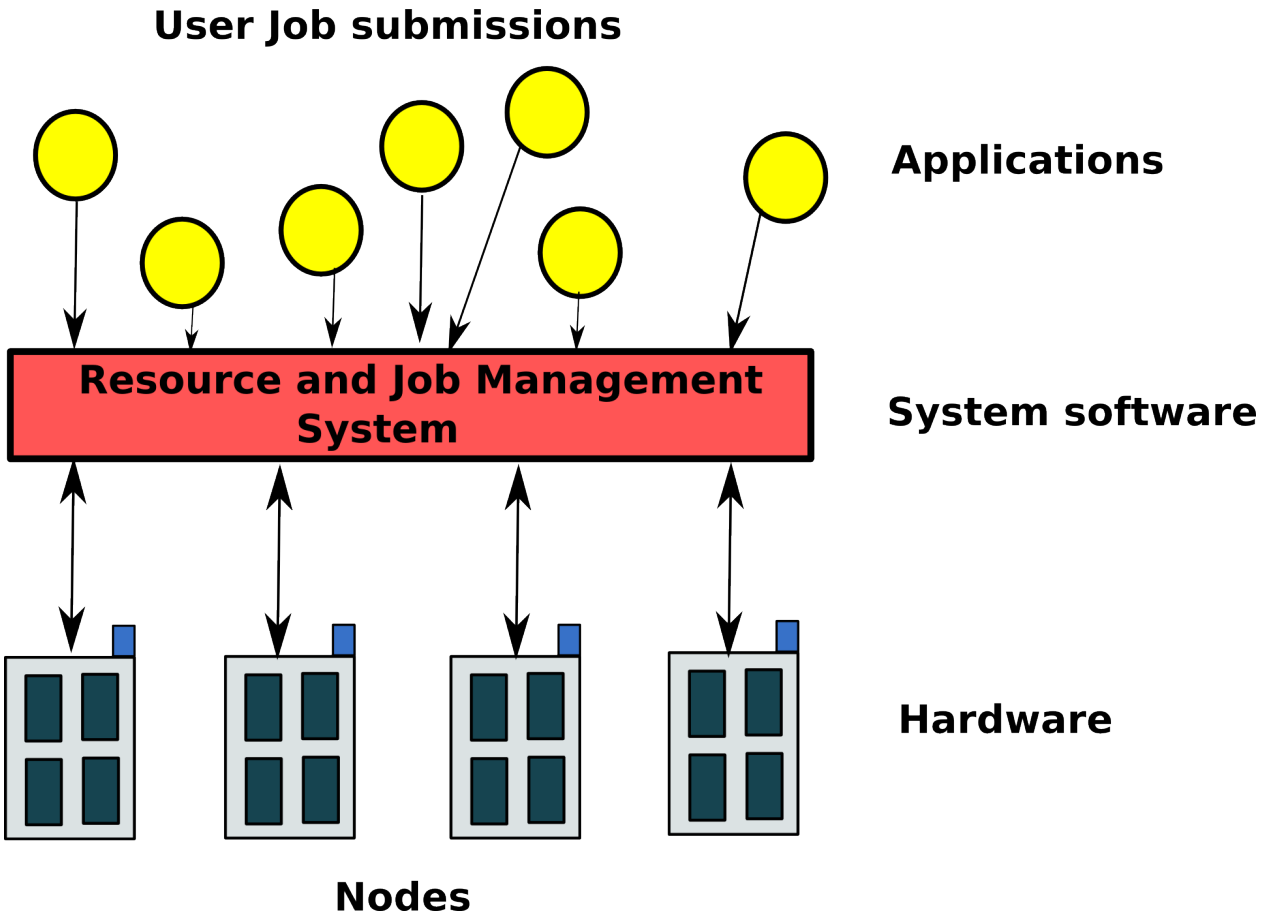
5 Visualisation

Outline

1. Généralités
2. Du processus à la grille, et même au delà
   - Grappe de calcul
   - Grille de calcul
   - Informatique dans le nuage
   - Grappe, Grille, Cloud : récapitulons
3. Les RJMS
   - Caractéristiques
   - Quelques RJMS
4. Fonctionnement
   - Les jobs
   - Les ressources
   - Politiques d'ordonnancement
5. Visualisation

Gestionnaires de tâches et de ressources : but

Le but d'un gestionnaire de tâches et de resources (RJMS) est de satisfaire la demande en calcul des utilisateurs et d'assigner les jobs aux ressources de calcul de manière efficace.



RJMS Importance

Strategic position but complex internals :

- **Direct and constant knowledge** of resources and jobs
- **Multifacet procedures** with complex internal functions

RJMS : concepts

L'assignation de ressources à un job implique 3 niveaux d'abstraction :

- la déclaration d'un job avec ses caractéristiques et contraintes sur les ressources
- l'ordonnancement du job par rapport aux ressources
- et le lancement et placement des instances du job sur les ressources de caclcul, ainsi que les éléments de controle d'éxécution

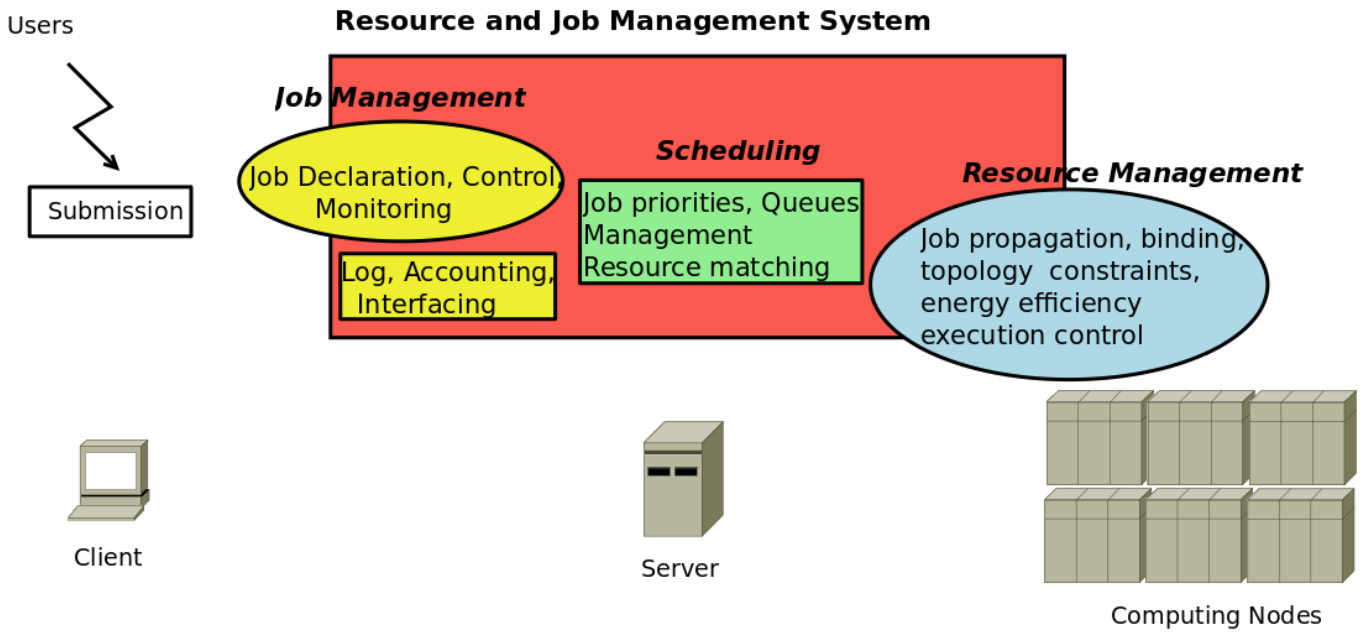
Dans ce sens, le travail d'un gestionnaire de tâches et de ressources peut se décomposer en 3 sous-systèmes : **Gestion des tâches**, **Ordonnancement** et **Gestion des ressources**.

RJMS : caractéristiques principales

sous-système	Caractéristiques génériques	Caractéristiques avancées
Gestion des jobs	-Déclaration de job (types, characteristiques,...) -Contrôle des jobs (signaling, reprioritizing,...) -Monitoring (reporting, visualisation,..)	- Authentification (limitations, securité,..) - QOS (checkpoint, suspend, accounting,. - Interfaçage (MPI libs, debuggers, APIs,.
Ordonnancement	-Algorithmes d'ordonnancement (builtin, externe,..) -Gestion des queues (priorités, classement,..)	- Reservation à l'avance - Licences logicielles - Mécanismes d'équité
Gestion des ressources	-Définition des ressources (hierarchie, partitions,..) -Lancement des jobs, propagation, contrôle éxéc -Placement des tâches (topologie, contraintes,...)	- Haute disponibilité - Gestion de l'énergie - Placement automatique en fonction de la topologie

RJMS : Organisation générale

- Un serveur central
- Programmes clients (ligne de commande à minima) pour l'interaction avec les utilisateurs
- De nombreux paramètres de configuration ! !



RJMS : Fonctionnalités (1/2)

liste non-exhaustive

- Tâche (*soumission*) Interactive (shell) / Batch
- Tâche séquentielle et parallèle
- Walltime (temps limite). (**important pour l'ordonnancement)**
- Accès exclusif / non-exclusif aux ressources
- Appariement de ressources
- Scripts Epilogue/Prologue (exécuter avant/après les tâches)
- Suivi (*monitoring* des tâches (consommation des ressources)
- Dépendance entre tâches (*workflow*)
- Logging et accounting
- Suspension/reprise des tâches

RJMS : Fonctionnalités (2/2)

liste non-exhaustive

- Dépendance entre jobs
- Tableaux de tâches
- First-Fit (Conservative Backfilling,)
- Fairsharing
- ...

Outline

1. Généralités
2. Du processus à la grille, et même au delà
   - Grappe de calcul
   - Grille de calcul
   - Informatique dans le nuage
   - Grappe, Grille, Cloud : récapitulons
3. Les RJMS
   - Caractéristiques
   - Quelques RJMS
4. Fonctionnement
   - Les jobs
   - Les ressources
   - Politiques d'ordonnancement
5. Visualisation

Quelques gestionnaires de tâches et de ressources

Open Source RJMS

- SLURM
- TORQUE
- MAUI
- OAR
- CONDOR
- SGE (before Oracle)

Commercial RJMS

- Loadleveler
- LSF
- MOAB
- PBSPro
- OGE (Oracle Grid Engine)

Quelques gestionnaires de tâches et de ressources

Open Source RJMS

- SLURM
- TORQUE
- MAUI
- OAR
- CONDOR
- SGE (before Oracle)

Commercial RJMS

- Loadleveler
- LSF
- MOAB
- PBSPro
- OGE (Oracle Grid Engine)

Etude comparative

- ”Quantifiable Functionalities Evaluation of opensource and commercial RJMS” Yiannis Georgiou (05/11/2010 Phd Thesis Bull / UJF)

RJMS Quantifiable Functionalities Comparison

Quantifying Functionalities support by RJMS (Yiannis Georgiou)

- Resource Management
  - Resources Treatment, Job Launching, Task Placement, High Availability,...
- Job Management
  - Job declaration, Job Control, Monitoring, Interfacing, Quality of Services,...
- Scheduling
  - Scheduling Algorithms, Queues Management, Advanced Reservations,...

Overall Evaluation / RJMS Software	SLURM	CONDOR	TORQUE	OAR	MAUI	LSF
Resource Management (/10)	7.1	5.2	5.2	6.9	1.9	6.9
Job Management (/10)	5.1	6.5	5.1	5.5	3.1	6.8
Scheduling (/10)	6	5.3	3	5.7	5.5	5.7
Overall Evaluation Points (/10)	**6.2**	5.7	5.1	**6**	3.4	**6.4**

RJMS Quantifiable Functionalities Comparison

Quantifying Functionalities support by RJMS (Yiannis Georgiou)

- Resource Management
  - Resources Treatment, Job Launching, Task Placement, High Availability,...
- Job Management
  - Job declaration, Job Control, Monitoring, Interfacing, Quality of Services,...
- Scheduling
  - Scheduling Algorithms, Queues Management, Advanced Reservations,...

Overall Evaluation / RJMS Software	SLURM	CONDOR	TORQUE	OAR	MAUI	LSF
Resource Management (/10)	7.1	5.2	5.2	6.9	1.9	6.9
Job Management (/10)	5.1	6.5	5.1	5.5	3.1	6.8
Scheduling (/10)	6	5.3	3	5.7	5.5	5.7
Overall Evaluation Points (/10)	**6.2**	5.7	5.1	**6**	3.4	**6.4**

RJMS : Etude comparative

Que conclure de cette étude ?

- Qu'elle n'est pas exhaustive !
- Que SLURM est un très bon RJMS open-source
- Que LSF est un très bon RJMS commercial
- Que OAR est un bon RJMS open-source :-)
- Que c'est peut-être sur des caractéristiques très spécifiques que va se faire votre choix...

RJMS : Choix

Exemples de critère de choix

- Budget : vous avez les moyens de payer un support de haut niveau – LSF
- Beaucoup de ressources et de jobs, recherche de scalabilité – SLURM
- Besoin de beaucoup de personnalisation – OAR
- Besoin d'interopérabilité avec une grille qui ne supporte que PBS-like – Torque/MAUI
- Besoin d'une API simple et performante – OAR (et son API-REST)

Attention

Mais les choses bougent !

Outline

1 Généralités

2 Du processus à la grille, et même au delà
- Grappe de calcul
- Grille de calcul
- Informatique dans le nuage
- Grappe, Grille, Cloud : récapitulons

3 Les RJMS
- Caractéristiques
- Quelques RJMS

4 Fonctionnement
- Les jobs
- Les ressources
- Politiques d'ordonnancement

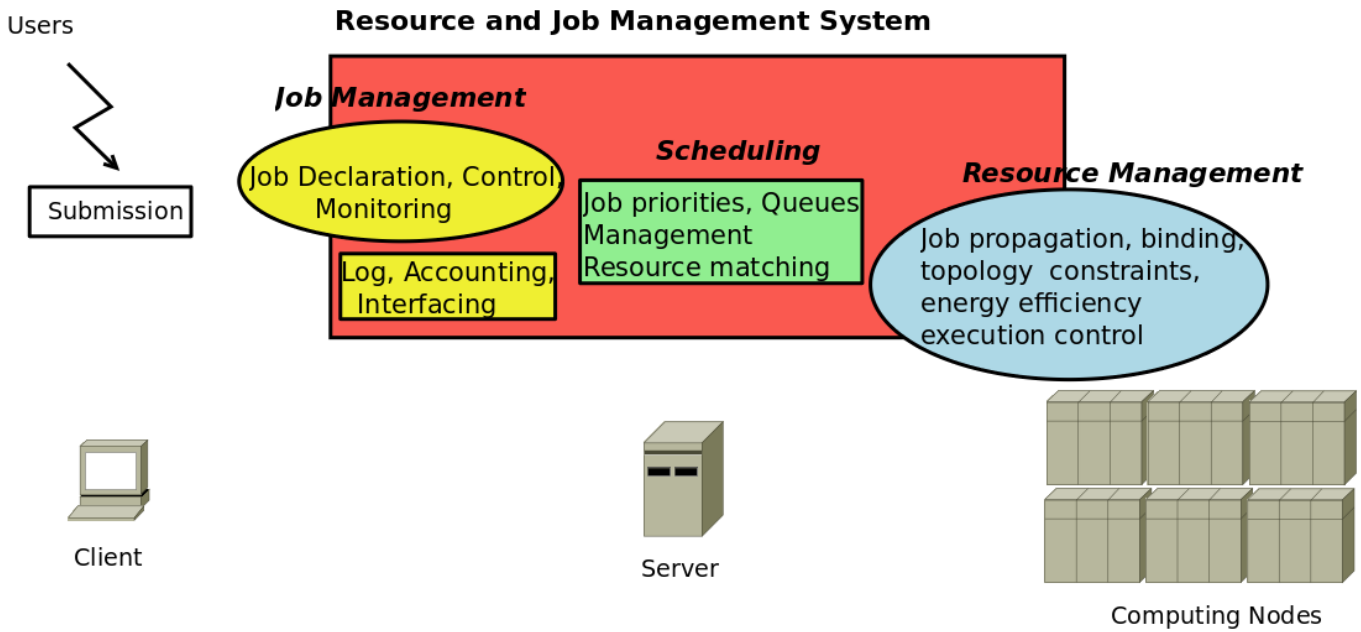
5 Visualisation

Outline

1. Généralités
2. Du processus à la grille, et même au delà
   - Grappe de calcul
   - Grille de calcul
   - Informatique dans le nuage
   - Grappe, Grille, Cloud : récapitulons
3. Les RJMS
   - Caractéristiques
   - Quelques RJMS
4. Fonctionnement
   - Les jobs
   - Les ressources
   - Politiques d'ordonnancement
5. Visualisation

RJMS : Fonctionnement

- Un serveur central
- Programmes clients (ligne de commande à minima) pour l'interaction avec les utilisateurs
- De nombreux paramètres de configuration ! !

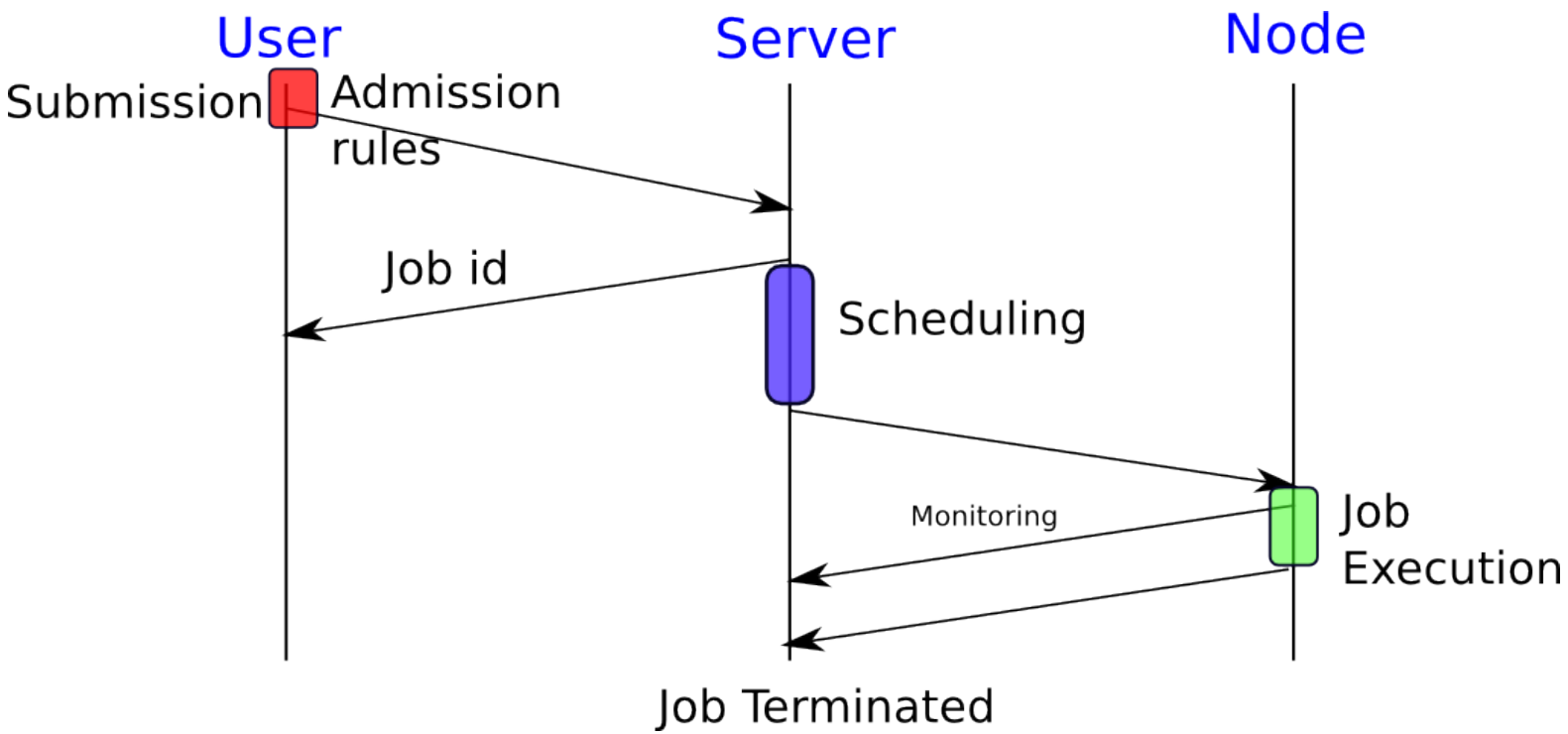


Jobs : Files d'attente (queues)

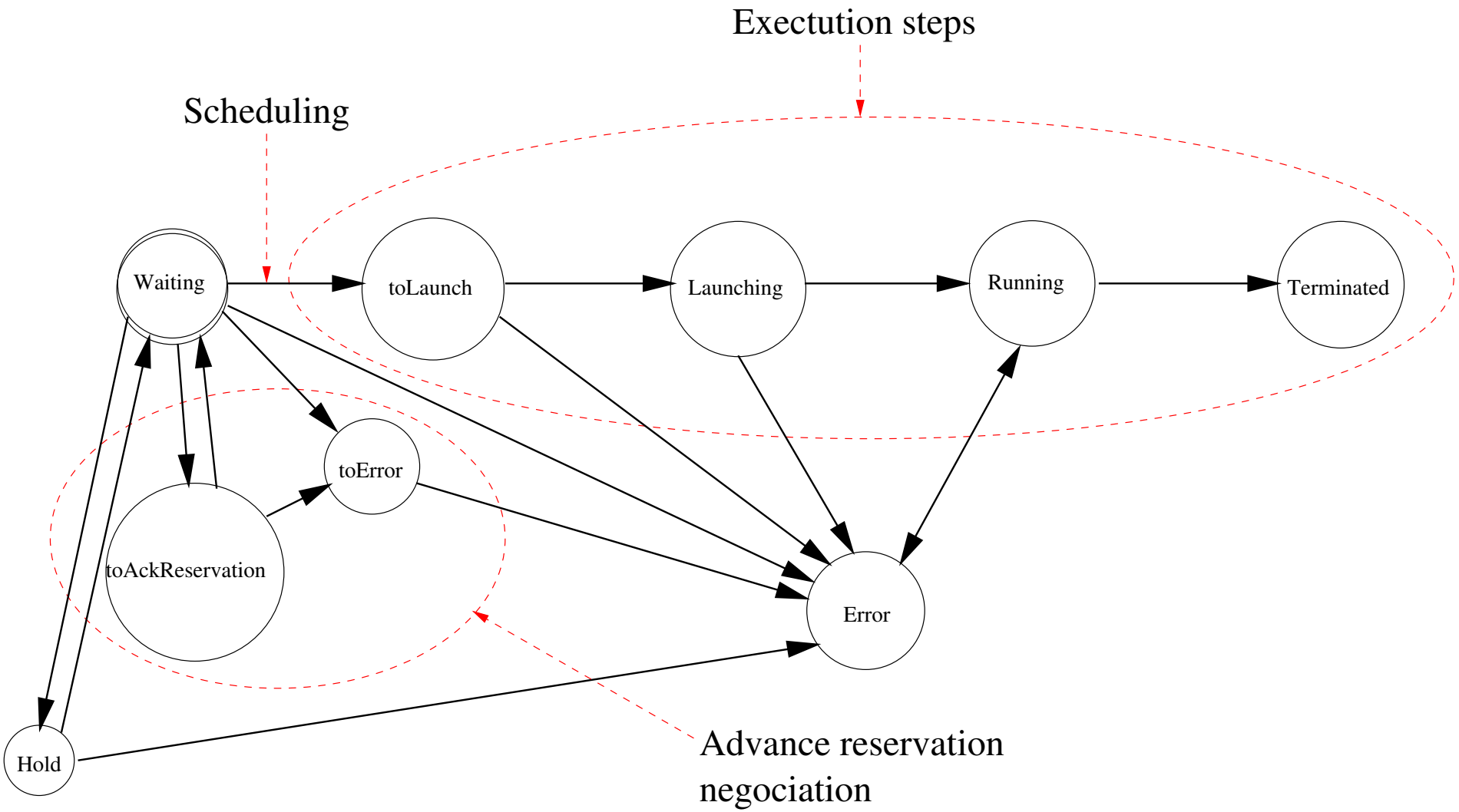
Queues

- La plupart du temps, les jobs soumis sont placés dans des files d'attente
- Ces "queues" permettent de classifier les jobs juste après leur soumission, en fonction de critères définit par l'administrateur (job long, job court, job appartenant à tel groupe d'utilisateur, etc...)
- Les queues sont traitées différemment par l'ordonnanceur (priorité, mode de fonctionnement ou d'ordonnancement particulier)
- Lorsqu'un job est en queue, il est en attente d'éxécution
- Un job en queue, peut être "schedulé" ou pas. Lorsqu'un job en queue est schédulé, l'ordonnanceur a déja prévu les ressources sur lesquelles le job va tourner, mais est en attente de libération de celles-ci.

Jobs : cycle de vie



Jobs : diagramme d'états d'un job (exemple du système OAR)



Exemples de soumission de job (exemple du système OAR)

Job interactif : [1]

- **oarsub -I nodes=4 -I**

Soumission en *Batch* (avec un *walltime* et choix de la file d'attente (queue)) :

- **oarsub -q default -l walltime=2 :00,nodes=10 /home/toto/script**

Connexion à un noeud d'un job en cours d'éxécution, en utilisant l'id :

- **oarsub -C 154**

1. **Note :** Chaque soumission retourne un numéro de job : **id**.

Exemples de suivi et d'action sur un job (exemple du système OAR)

Vérification de l'état d'un job

- **oarstat -j 51409**

```
Job id     Name           User           Submission Date     S Que
---------- -------------- -------------- ------------------- - ---
51409                     bzizou         2011-09-13 10:08:30 R def
```

Suppression d'un job

- **oardel 51409**

Démonstration de soumissions

Outline

1. Généralités
2. Du processus à la grille, et même au delà
   - Grappe de calcul
   - Grille de calcul
   - Informatique dans le nuage
   - Grappe, Grille, Cloud : récapitulons
3. Les RJMS
   - Caractéristiques
   - Quelques RJMS
4. Fonctionnement
   - Les jobs
   - Les ressources
   - Politiques d'ordonnancement
5. Visualisation

Ressources

La ressource

- La notion de ressource peut différer d'un RJMS à un autre
- En fait, elle peut même différer d'une instance d'un RJMS à une autre instance
- La ressource est en fait le plus petit sous-ensemble d'une grappe que l'on peut allouer à un job
- Ca peut-être un coeur, une socket ou un noeud tout entier
- Souvent, les ressources sont structurées : noeud/cpu/core

Appariement de ressources

- Classement des ressources
- Filtrage des ressources
- Spécification de besoins particulier (mémoire, architecture, OS, niveau de charge,...
- En général, pour que cet appariement soit possible, on défini des **propriétés** à chaque ressource

Spécificités du job à la soumission

Par exemple : mes jobs utilisent 4 noeuds et ont besoin d'un **minimum de 16Go de RAM par noeuds**

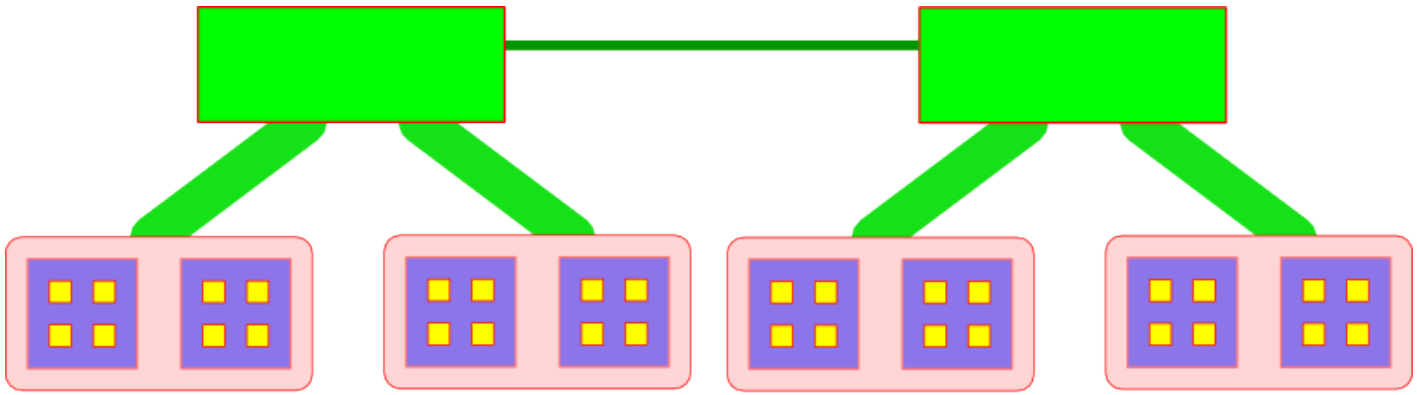
- **oarsub -l nodes=4 -p "memnode > 16" -l**

Contraintes Topologiques

Evolution du matériel : architectures non uniformes

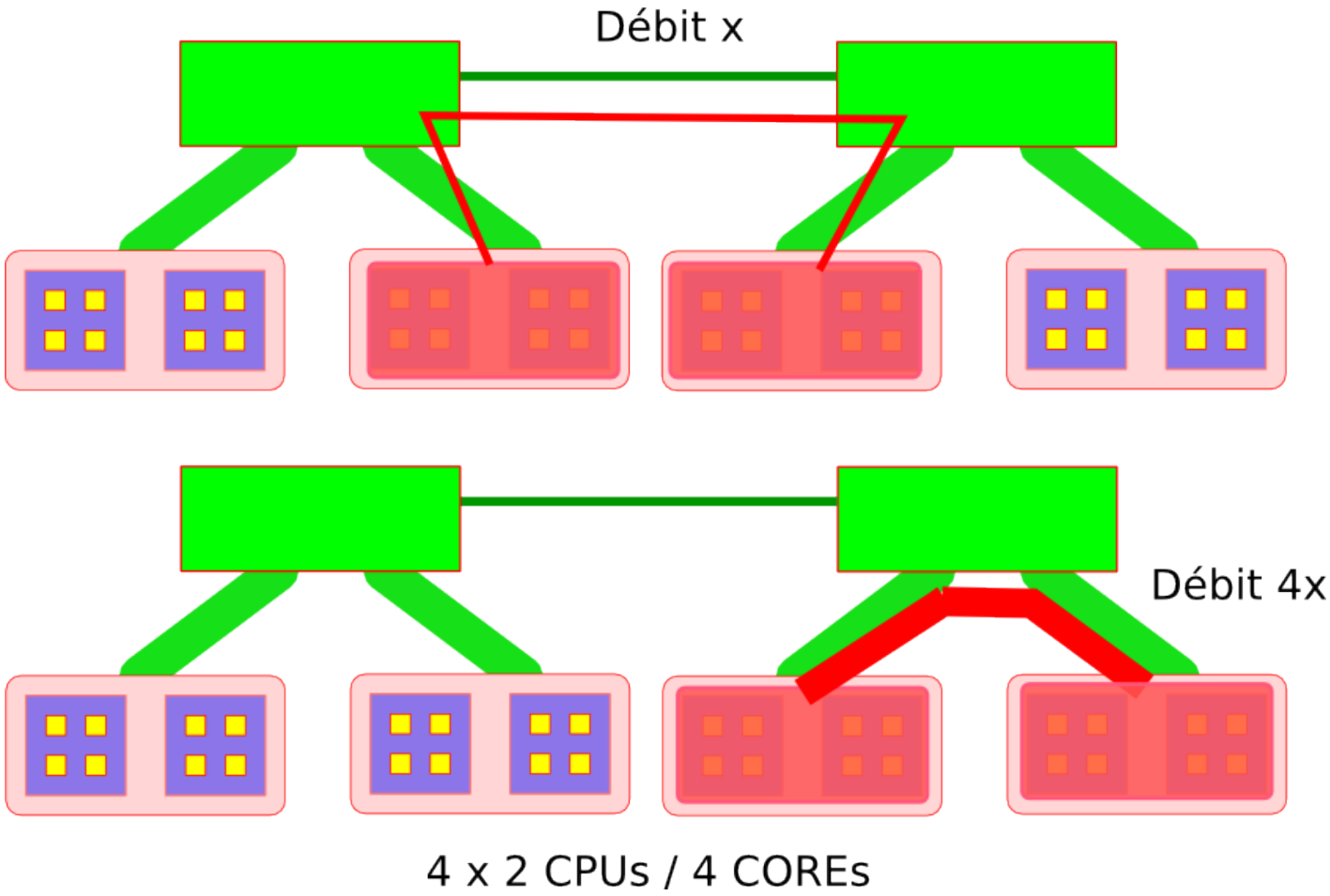
- switch/noeud/cpu/core : Architecture Hierarchique
- machine NUMA / machine BlueGene : Architecture en grille 2D, 3D ou hybride

▸ ...



Contraintes Topologiques hiérarchiques

Problème avec les applications parallèles sensible au débit communication.



Application parallèle et affinité processeur

Note : CPUSET ensemble de coeurs et/ou CPU sur un noeud.

1. L'attribution CPUSET/core pour application parallèle peut ne pas suffire
2. Problème de l'ordonnanceur de l'OS (ici souvent Linux), le processus change de coeur à l'intérieur des CPUSET
3. Il faut utiliser les capacités de verrouillage sur coeur (*Processor Affinity, par exemple "taskset -c 0,1 programme"*)

Gestion de l'énergie

- Certains RJMS ont des fonctionnalités de gestion de l'énergie.
- Il est vivement recommandé de les utiliser lorsque vous avez une charge qui n'est pas constante, avec des pics d'utilisation et des périodes creuses.
- Dans ce cas, le RJMS peut éteindre et allumer les noeuds à la demande, évitant ainsi une consommation d'énergie en l'absence de jobs.
- On peut même avoir des contraintes environnementales qui vont demander une diminution des noeuds actifs à certaines périodes (cas d'un système en freecooling total par exemple)

Gestion de l'énergie

Peut être assez complexe

- Au bout de combien de temps sans jobs éteindre un noeud ?
- Anticiper l'arrivée de nouveaux jobs, par exemple en gardant quelques noeuds toujours allumés
- Gérer le temps d'allumage des noeuds
- Que faire d'un noeud qui ne se rallume pas ?
- Offrir la possibilité de baisser la fréquence pendant les jobs i/o

Outline

1. Généralités
2. Du processus à la grille, et même au delà
   - Grappe de calcul
   - Grille de calcul
   - Informatique dans le nuage
   - Grappe, Grille, Cloud : récapitulons
3. Les RJMS
   - Caractéristiques
   - Quelques RJMS
4. Fonctionnement
   - Les jobs
   - Les ressources
   - Politiques d'ordonnancement
5. Visualisation

L'ordonnancement (scheduling)

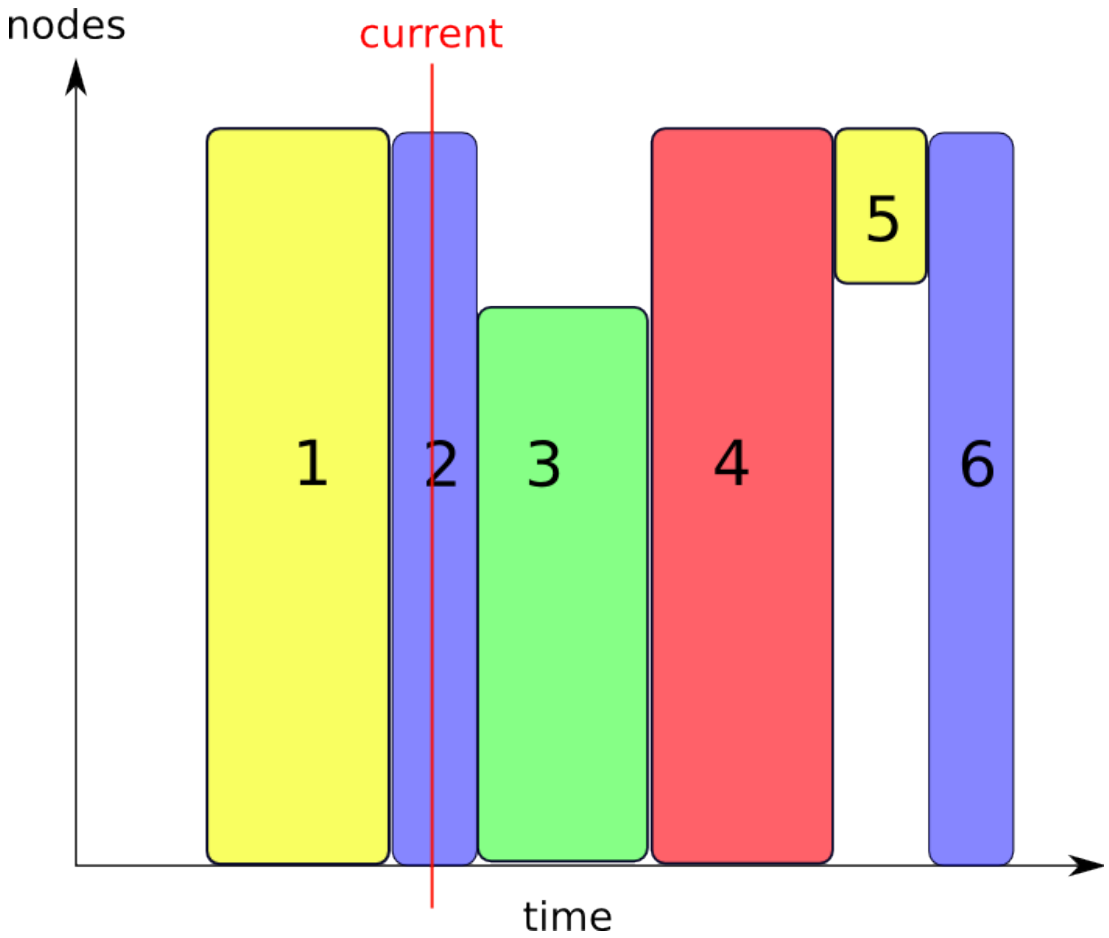
L'ordonnancement est l'étape [2] où le système choisi les **ressources à attribuées** aux tâches et **les dates de lancement**.

L'ordonnancement est défini suivant une **politique** qui se traduit par l'utilisation **d'algorithmes d'ordonnancement**.

De plus de nombreux **critères et paramètres** sont utilisés pour guider et cadrer les allocations et les priorités.

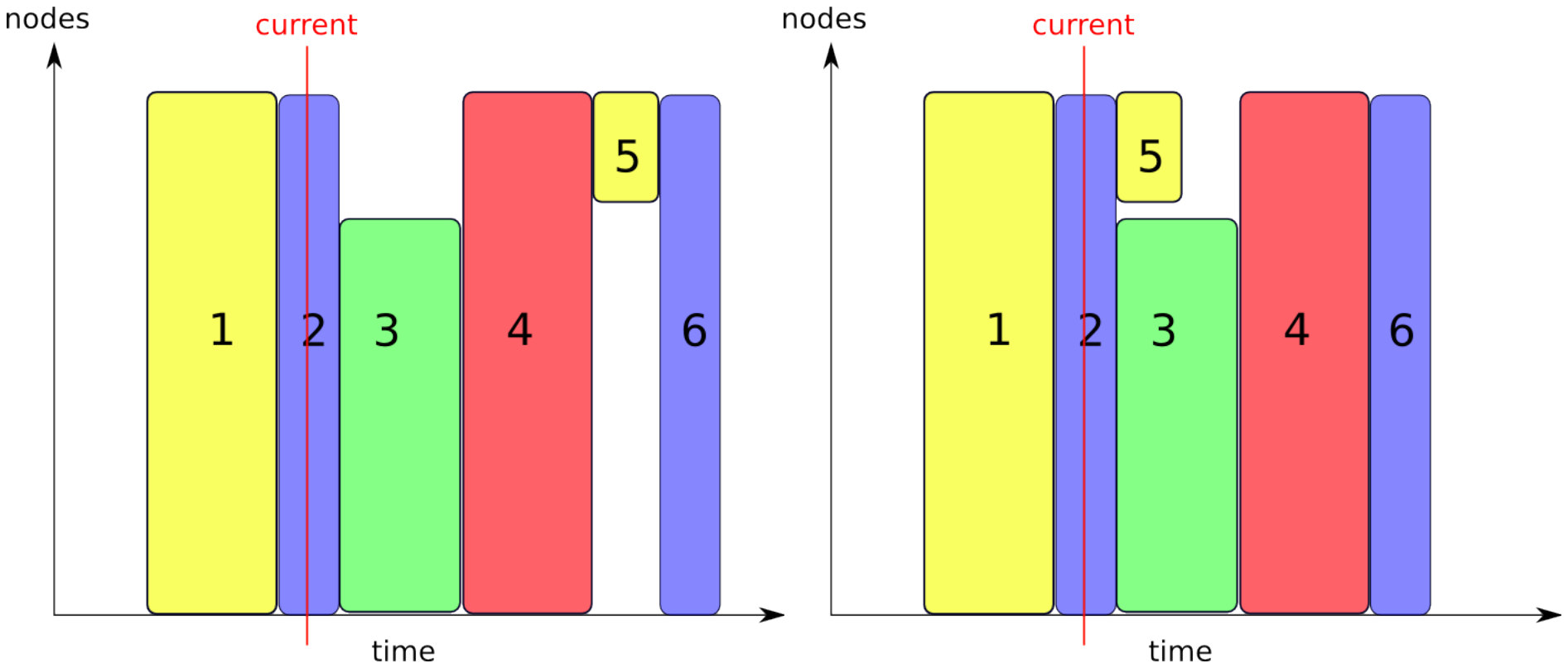
2. **Note :** l'ordonnancement est recalculé à chaque changement d'état (majeur) d'une tâche.

FIFO : Fisrt-In First-Out



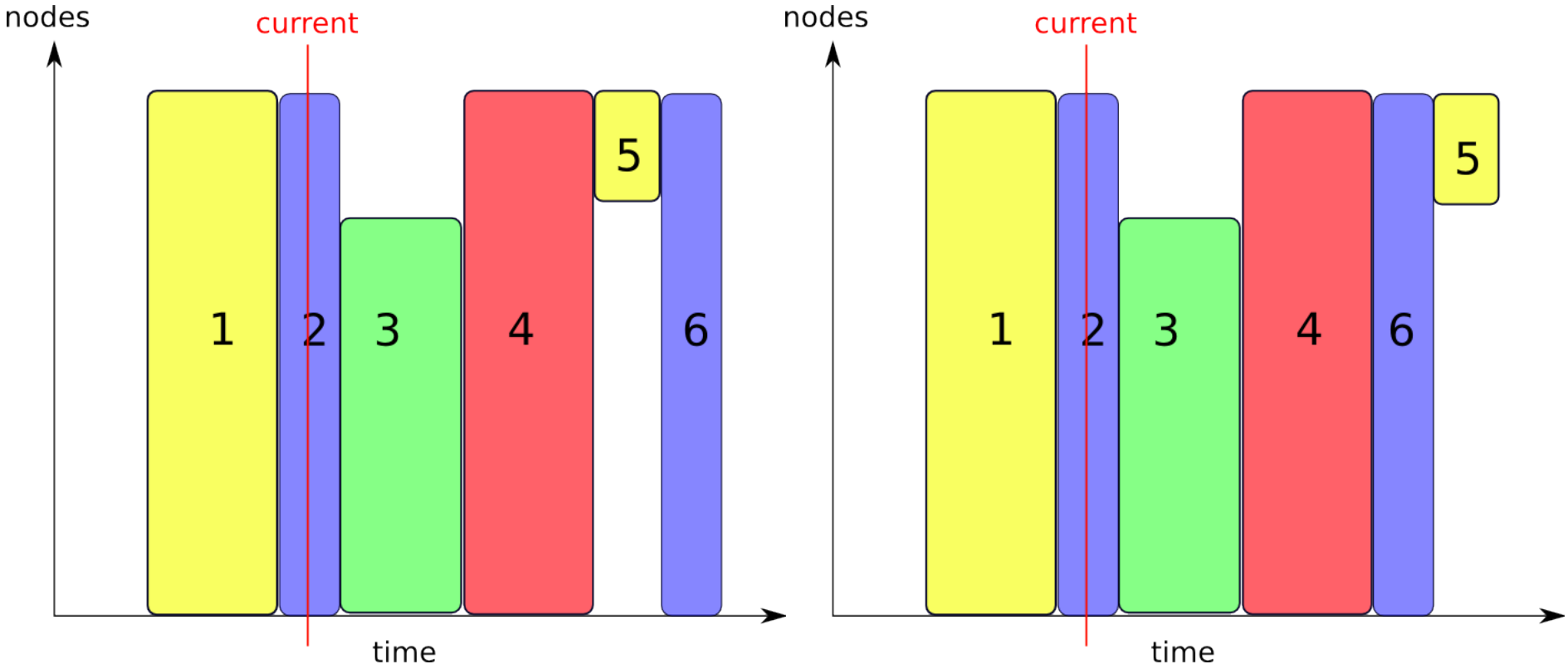
First-Fit (Backfilling)

Remplissage des trous si l'ordre des tâches soumises antérieurement n'est pas modifié



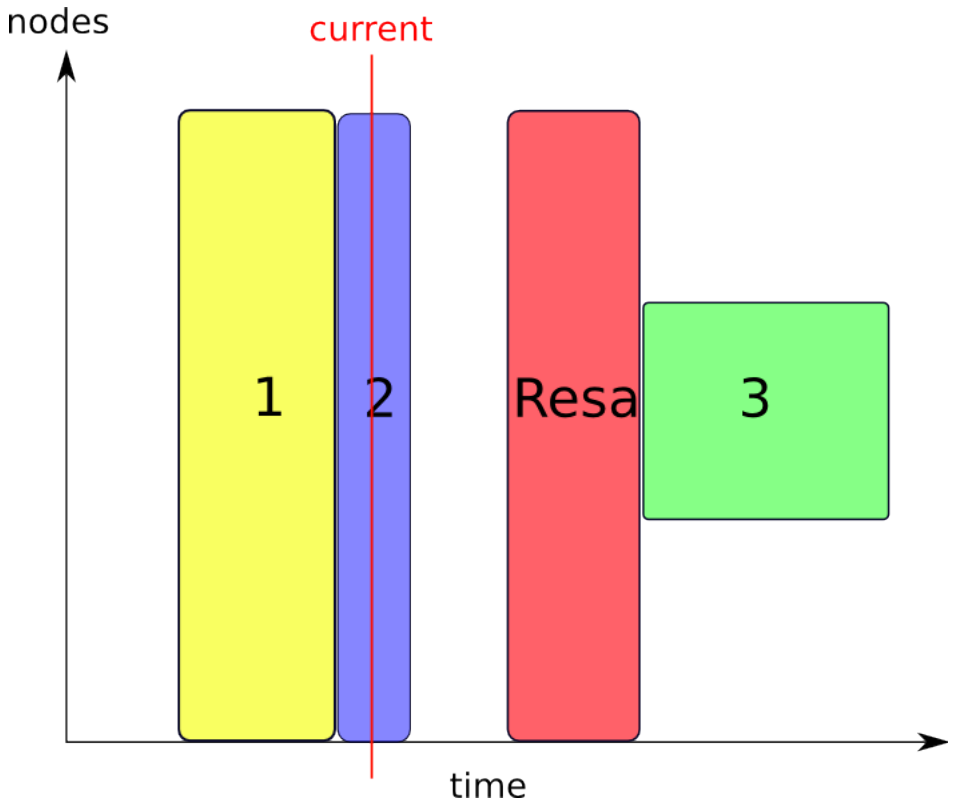
FairSharing (partage équitable)

L'ordre est calculé suivant ce qui a été consommé (on favorise les utilisateurs peu gourmands). Définition d'une fenêtre et paramètres de pondération.



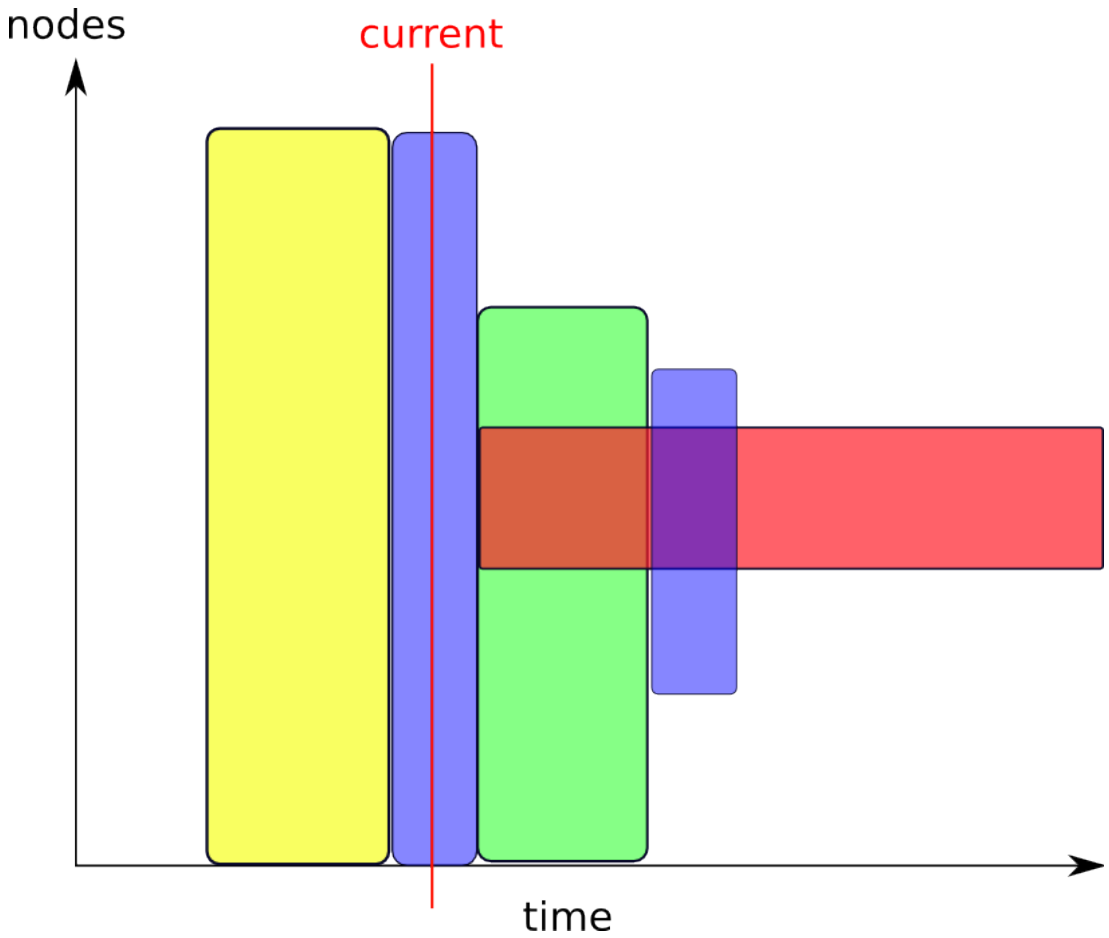
Réservation (*Advance Reservation*)

- **Très pratique** pour démo, planification, tâche multi-site ou de type grille...
- **Mais**
  - Contraignant pour l'ordonnanceur (attention au niveau d'utilisation)
  - Les ressources sont rarement utilisée sur toute la durée (gaspillage)



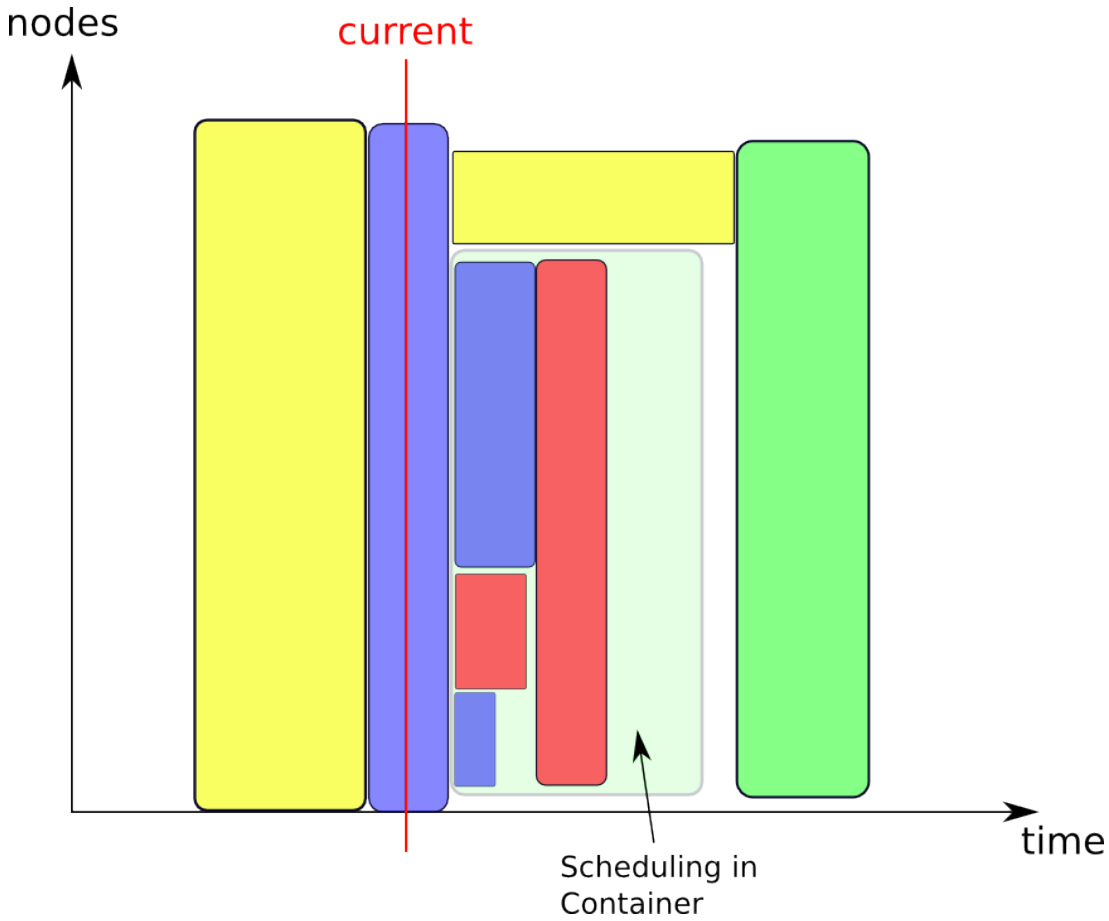
oarsub -r "2008-04-27 11 :00" -l nodes=12

TimeSharing



Récursivité

Faire de l'ordonnancement dans une allocation/réservation. Intéressant pour formation, démo, partage de ressource plus flexible par groupe d'utilisateurs / projet.
Tâche de type container.



Outline

1 Généralités

2 Du processus à la grille, et même au delà
- Grappe de calcul
- Grille de calcul
- Informatique dans le nuage
- Grappe, Grille, Cloud : récapitulons

3 Les RJMS
- Caractéristiques
- Quelques RJMS

4 Fonctionnement
- Les jobs
- Les ressources
- Politiques d'ordonnancement

5 Visualisation

Etat instantané des ressources (exemple de OAR

OAR Cluster nodes

default summary

	Free	Busy	Total
network_address	4	15	32
resource_id	32	120	256

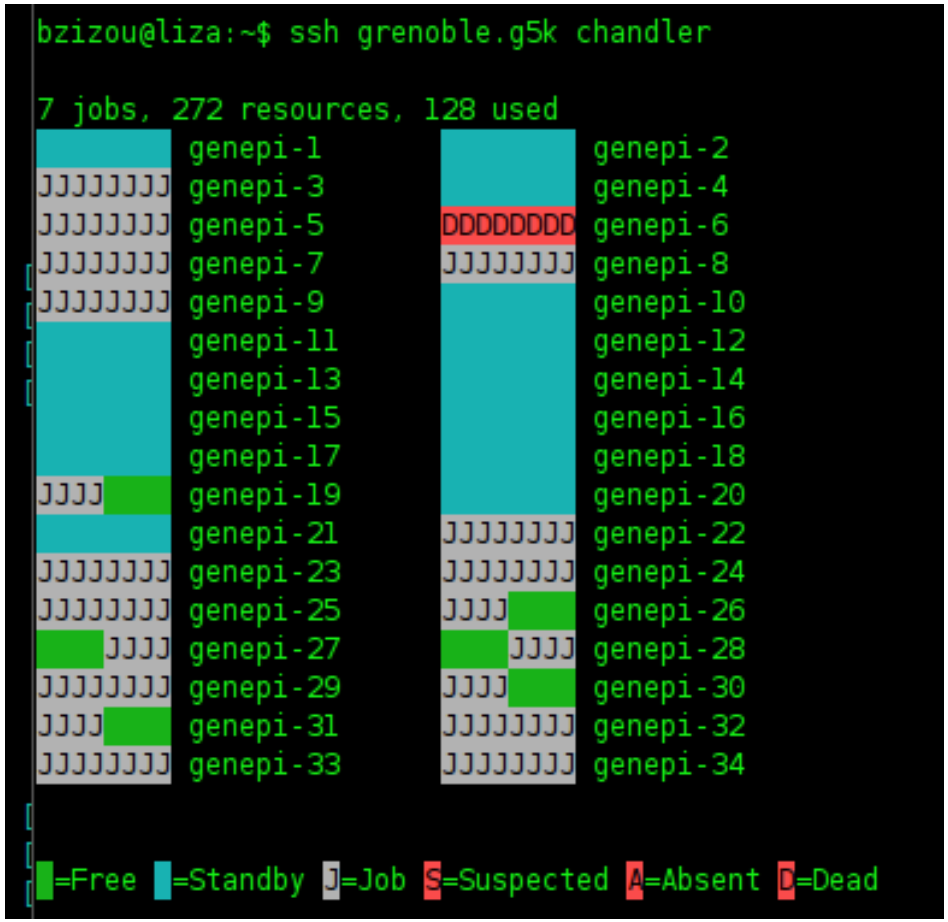
Reservations:

Reservations for property iru=0:

rli0n0	182270	182270	182270	182270	182270	182270	182270	182270
rli0n1	182270	182270	182270	182270	182270	182270	182270	182270
rli0n2	Free	Free	Free	Free	Free	Free	Free	Free
rli0n3	Free	Free	Free	Free	Free	Free	Free	Free
rli0n4	182267	182267	182267	182267	182267	182267	182267	182267
rli0n5	Free	Free	Free	Free	Free	Free	Free	Free
rli0n6	182271	182271	182271	182271	182271	182271	182271	182271
rli0n7	182271	182271	182271	182271	182271	182271	182271	182271
rli0n8	182271	182271	182271	182271	182271	182271	182271	182271
rli0n9	StandBy	StandBy	StandBy	StandBy	StandBy	StandBy	StandBy	StandBy
rli0n10	StandBy	StandBy	StandBy	StandBy	StandBy	StandBy	StandBy	StandBy
rli0n11	182294	182294	182294	182294	182294	182294	182294	182294
rli0n12	182282	182282	182282	182282	182282	182282	182282	182282

Diagramme de Gantt (exemple de OAR)

Visualisation rapide du cluster en mode texte (exemple de OAR)



APIS

Certains RJMS offrent des APIS pour que les utilisateurs et administrateurs puissent facilement interagir de manière programmatique. Les APIS peuvent être RESTfull ou compatible RDMAA,... Cela peut permettre la réalisation de simple outils de visualisation, tout comme de véritables portails de gestion de jobs (soumission, suivi, etc...)., ou encore peut aider aux interractions avec un middleware de grille (ex : grid5000)

Merci

Bruno.Bzeznik@imag.fr