

# vol de travail et communications

You

October 14, 2015

objectif : réaliser un simulateur

- il prend en entrée deux fichiers
  - un fichier qui décrit la plateforme
    - \* à faire plus tard ; pour le moment juste un entier indiquant le nombre de processeurs
  - un fichier qui décrit le graphe de tâches
    - \* chaque ligne contient la description d'une tâche : son nom, le travail, le volume du fichier généré, la liste des dépendances
    - \* il existe des tâches spéciales, de durée 0 : elles doivent être exécutées obligatoirement sur le processeur qui les rend prêtes.
- il exécute un algo d'ordonnancement centralisé
  - quand un processeur n'a plus rien à faire, il prend la première tâche prête dans la file. il récupère tous les fichiers dont il a besoin, qu'il n'a pas encore (à simuler) puis exécute la tâche.
  - quand une tâche termine, il faut regarder qui dépend d'elle et si c'était la dernière dépendance en attente alors marquer la tâche comme prête et l'ajouter dans la file.
  - événements : fin de tâche, fin de transfert de fichier
  - quand une tâche nécessite plusieurs fichiers, elle les récupère un par un. Chaque récupération prend un certain temps.
- il génère une trace visualisable dans un outil de visualisation (ex: viva <https://github.com/schnorr/viva/>)
- il affiche le temps d'exécution ; le volume de données transférées ; les communications sur le chemin critique ???

exemple de fichier de tâches:

```
t1 34 234234 t2
t2 23423 54354 t3 t4
t3 0 23432.5235 t5
t4 9813.34134 0 t5
t5 3983 39843
```

pour git: le workflow est le suivant:

- tout au début, faire un *git clone https://github.com/wagnerf42/ws-simulator*
- avant de travailler, récupérer les modifs distantes : *git pull*
- travail local
- validation du travail local: *git add fichier1 fichier2 fichier3* où fichier1,2,3 sont les fichiers modifiés
- sauvegarde : *git commit*
- upload sur le site : *git push*

pour la suite:

- passer à un ordo décentralisé
- générer automatiquement les fichiers de tâches
- analyse théorique
- passage à un vrai outil -i, vrais tests au lieu de simulations