

Calculco Quick Reference

by calculco@univ-littoral.fr - version 2021-07-16

identifiants Pôle Calcul

Création : saisir le formulaire :

<https://www-calculco.univ-littoral.fr/>

login,password (et mail) autorisent l'accès aux services suivants :

- **calculco.univ-littoral.fr** : accès calculs (frontal du cluster)
- **pcldata.univ-littoral.fr** : espaces de données du cluster (à privilégier pour les échanges de données conséquents depuis l'extérieur)
- **web** : accès aux pages privées (monitoring, téléchargement Matlab ...)
- **services annexes** : forge logiciel ([gogs/git](#)) et service « Dropbox like » [pcsbox/Seafile](#) (mail et password)

accès plateforme Calculco

Outils : Protocol ssh

Windows : [bitvise](#) (ou ssh , Win 10)

OS X / Linux : [ssh](#) (terminal ou console)

connexion : `ssh login@calculco.univ-littoral.fr`

Configuration de bitvise :

- Cadre Server de l'onglet Login

Host : `calculco.univ-littoral.fr`

Port : 22

- Cadre Authentication de l'onglet Login

Username : `monlogin`

(sauver une fois pour toute le *profile*)

Clé ssh (OS X / Linux) :

- créer la clé : `ssh-keygen -t ed25519 -o -a 64`

- la copier : `ssh-copy-id -i ~/.ssh/id_ed25519.pub monlogin@calculco.univ-littoral.fr`

- ajouter ou créer votre fichier `~/.ssh/config`

Host `calculco` (libre choix 1)

HostName `calculco.univ-littoral.fr`

User `monlogin`

IdentityFile `~/.ssh/id_ed25519.pub`

Host `pcldata` (libre choix 2)

HostName `pcldata.univ-littoral.fr`

User `monlogin`

IdentityFile `~/.ssh/id_ed25519.pub`

- désormais la connexion se résume à :

`ssh calculco` (selon libre choix 1)

- (... et un transfert de données :) :

`rsync -azuv ~/projet pcldata:` (selon libre choix 2)

Espaces disques

<code>\$HOME</code>	100 GB «monté»(NFS) sur tous les nœuds. Répertoire personnel. Code source et résultats (si non volumineux). Espace sauvegardé.
<code>\$DATA</code>	<code>/nfs/data/unprotected/labo.</code> Espace sans quota (la limite est l'espace labo), pour données et résultats volumineux. Non sauvegardé. <code>/nfs/data/protected/labo.</code> Idem mais plus petit (environ 10 % de l'espace <code>unprotected</code>). Sauvegardé.
<code>/scratch</code>	Espace local à chaque nœuds. Indispensable pour les travaux à écritures et lectures intensifs. Accessible (post-traitement) par NFS sur calculco (<code>/scratch/orvalXX/labo/login</code>). Purgé régulièrement.
<code>/workdir</code>	Espace BeeGFS de 30 To sans quota. Même usage (I/O intensifs), temporaire (purgé régulièrement).

Transfert de fichiers

2021 (new) : les échanges de fichiers **conséquents** se font directement sur `pcldata.univ-littoral.fr`

Commandes : Linux, MacOS (,Win10). Avec la configuration *clé ssh* et `~/.ssh/config` :

- `rsync -azuv project pcldata:`
- `rsync -azuv pcldata:project/results .`
- `scp -rp smallest calculco:`
- `scp newfile calculco:smallest/`

Outils graphiques :

— **Windows** : [bitvise](#) ou [Filezilla](#) ou ...

— **OS X/Linux** : [Filezilla](#) ou ...

Configuration bitvise :

Ouvrir l'onglet SFTP (configuration déjà effectuée cf. cadre «accès plateforme Calculco»)

Mieux : dupliquer le *profile* et remplacer `calculco` par `pcldata`

Configuration de Filezilla :

Host : `sftp://pcldata.univ-littoral.fr .`

User : `yourlogin`

Port : 22 .

Gestionnaire de ressources & tâches

Plateforme multi-utilisateurs, calculs concurrents, un *batch scheduler* est indispensable : [OAR](#)

commandes basiques :

<code>oarsub -I</code>	mode Interactif (mini test)
<code>oarsub -S ./script</code>	lancement par Script,
<code>oarsub -C JobID</code>	se Connecter (console) au <i>JobID</i>
<code>oardel JobID</code>	supprimer le <i>JobID</i>
<code>oardel -array arID</code>	suppr. le tableau de jobs
<code>oarstat -sj JobID</code>	status du <i>JobID</i> (running, waiting ...)
<code>oarstat -fj JobID</code>	full status du <i>JobID</i>
<code>oarstat -u monlogin</code>	status de mes jobs

principales options :

<code>-l, --resource <LIST></code>	spécifie les ressources
<code>-n, --name=<txt></code>	donner un nom au job
<code>-q, --queue <QUEUE></code>	choix de la queue
<code>-p, --property "<LIST>"</code>	spécifier certaines propr.
<code>-t, --type <TYPE></code>	spécifie le type de job
<code>--notify <TXT></code>	au choix : mail ou script
<code>-O <FILE></code>	nom fich. sortie standard
<code>-E <FILE></code>	nom fich. erreur standard

hiérarchie des ressources & propriétés (exemples)

<code>-l core=1,walltime=2:00:00</code>	un cœur pour 2 heures (temps par défaut)
<code>-l nodes=1/cpu=1/core=8</code>	8 cœurs sur 1 cpu
<code>-l nodes=1/cpu=2/core=4</code>	8 cœurs sur 2 cpu (noter la multiplication)

ajouts de propriétés ou de type :

<code>-t token:matlab=1</code>	réserve un jeton matlab
<code>-t besteffort</code>	réserve en mode besteffort
<code>-p 'host="orval02"'</code>	précise le nœud
<code>-p 'nodemodel="Dell_R630"'</code>	précise le type de nœud

[tableau synthétique des équipements](#)

Politique de réservation : queues (option -q)

- **besteffort** : aucune contrainte mais priorité la plus basse, donc aucune garantie d'arriver à terme (cf. `-t idempotent`)
- **default** : 3000 heures*cœurs de calcul sur au plus 128 cœurs
- **long** : au plus 12 cœurs à répartir sur au plus 12 jobs sans limite de durée.
- **projet-lambda** réservée aux membres du *projet-lambda* (sur le matériel dédié)

Exemples de séquence



Exemple 1 transférer monprojet et lancer le calcul
monPc\$rsync -azuv ~/monprojet calculco:acalculer
monPc\$ ssh calculco # (si vous avez édité votre ~/.ssh/config)
calculco\$ cd acalculer
calculco\$ oarsub -S ./launch.oar
OAR_JOB_ID=123456
oarstat -fj 123456 # (contrôler éventuellement l'état du job)
où le script launch.oar pourra contenir (a minima) :
#/bin/bash
#OAR -l /core=4/cpu=2,walltime=10:00:00
#OAR -q besteffort
./moncalcul

Exemple 2 se connecter au job précédemment lancé .
monPc\$ ssh calculco
calculco\$ oarsub -C 123456
par ex. vérifier usage CPU et mémoire de moncalcul :
orval02:~/acalculer/\$ htop -u monlogin

Exemple 3 session interactive matlab .
monPc\$ ssh -X calculco
calculco\$ nedit test.m
calculco\$ oarsub -I
orval01\$ ml matlab (ou module load matlab)
orval01\$ matlab -nodesktop -nodisplay < test.m

Outils-Web de supervision



www-calculco.univ-littoral.fr/monitoring (identification obligatoire)

- **drawgantt** : diagramme de Gantt de tous les jobs en cours d'exécution.
- **monika** : status du cluster, synthèse de l'usage de tous les cœurs, noeud par noeud. **NB** : utiliser le formulaire pour sélectionner les *propriétés* (quel orvalXX ? cpu ? etc.)
- **ganglia** : détail des nœuds (usage : cpu, mémoire, réseau etc.)
- **zabbix** : autres tableaux de bord et usage de jetons logiciels (Matlab).

Question, support



Q : Où trouver la documentation du cluster ?

R : www-calculco.univ-littoral.fr

Q : J'ai besoin d'aide, que faire ?

R : tout d'abord, lire les tutoriaux sur www-calculco.univ-littoral.fr, ou envoyer un mail sur la liste des utilisateurs calculco-users@univ-littoral.fr.
et finalement envoyer un mail aux administrateurs calculco@univ-littoral.fr.

Q : Y a-t-il un guide de référence pour les commandes Linux ?

R : Oui : tinyurl.com/unixquickref-pdf

Q : Je ne peux plus me connecter sur le frontal ?

R : Votre adresse IP a probablement été bannie à la suite d'erreurs répétées de **login** et/ou **password**. Le bannissement est momentané (quelques heures à 3 jours si récidive) : communiquez-nous votre « vraie » adresse IP (par ex. : <http://ip4.me/>)


Q : J'ai des besoins importants de mémoire. Que faire ?

R : cf. www-calculco.univ-littoral.fr/utilisation/memoire-recommandations

Modules & environnement logiciel



l'OS du cluster est une **debian** qui dispose de nombreux **packages**, accessibles dans le **path** par défaut du système d'exploitation. D'autres logiciels / bibliothèques sont installés dans des répertoires non standard du système. Inexistants (sous forme de package) ou recompilés (avec d'autres bibliothèques pour des raisons de versions, de performances, etc.). L'utilisateur modifie ses variables d'environnement pour sélectionner les applications de son choix avec le *gestionnaire de modules* **lmod** (projet **Univ. Texas -TACC-**)

commande	description
module help (ou ml help)	lister les sous commandes
module avail (ou ml av)	lister les modules disponibles
module spider (ou ml spider)	informations supplémentaires
module spider Matlab	infos suppl. sur Matlab
module load Matlab (ou ml Matlab)	charger Matlab
module list (ou ml)	lister les modules chargés dans
module unload Matlab (ou ml -Matlab)	enlever Matlab
module purge (ou ml purge)	enlever tous les modules
astuce	
la complétion fonctionne	(touche )
sauvegarder restaurer son environnement	
module save (ou ml save)	sauvegarde des modules
module save env1 (ou ml save env1)	sauve dans ~/.lmod.d/env1
module savelist (ou ml savelist)	liste les sauvegardes
module restore env1	restaure les mod. définis dans ~/.lmod.d/env1
astuce	filtrage grep
ml av & grep -i opencv	...rend l'usage du grep!

module et scripts OAR



Les commandes modules sont accessibles en mode interactif. En mode batch, il **faut ajouter** l'instruction suivante dans le script de lancement OAR :

```
source /nfs/opt/env/env.sh (ou . /nfs/opt/env/env.sh)
```

Exemple de script OAR

```
#!/bin/bash  
#OAR -l core=4,walltime=0:30:00  
#OAR -q default  
#OAR -n moncalcul  
#OAR --notify mail:prenom.nom@univ-littoral.fr  
#OAR -O masortie.%jobid%.out  
#OAR -E masortie.%jobid%.err  
#OAR -t token:matlab=1  
source /nfs/opt/env/env.sh  
module load matlab  
PROG=testcalcul.m  
matlab -nodesktop -nodisplay -nosplash < $PROG > matlab.out
```