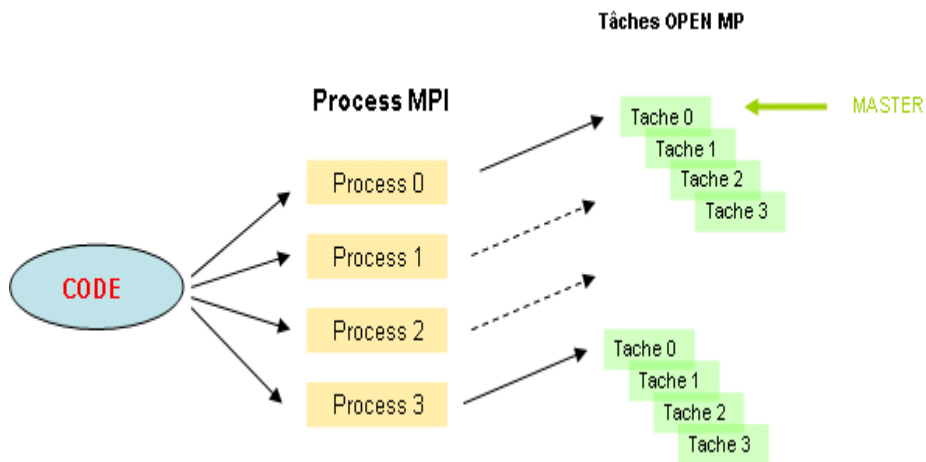


## Intégration OpenMP en standard dans IPSLCM5A (et autres configurations)

**Objectif :** Utiliser au maximum les machines de type SMP (vargas, titane, platine, jade) en s'affranchissant de la limitation du nombre de processus MPI (3 bandes de latitudes par process MPI) et des problèmes éventuels de mémoire intra-nœud en particulier lorsqu'on monte en résolution.

**Solution :** Utilisation de la parallélisation mixte MPI-OpenMP (MPI inter-nœud, OpenMP intra-nœud).



Ce qu'on sait déjà faire :

1) extraire LMDZ (trunk version), extraire ORCHIDEE (tag 1\_9\_2\_OpenMP), compiler en activant « à la main » les options qu'il faut dans AA\_make de la configuration (IPSLCM5A, LMDZOR\_v3):

- ./makelmdz\_fcm -parallel mpi\_omp (sans la cle cpp ORCHIDEE\_NOOPENMP) pour LMDZ
- ./configure\_orchidee -p mpi\_omp pour ORCHIDEE (possible avec 1\_9\_2\_OpenMP)

2) créer un job « à la main » avec les bons paramètres et lancer dans un RUN\_DIR (sans libIGCM). Par exemple pour jade :

- #PBS -l select=147:ncpus=8
- export OMP\_NUM\_THREADS=8
- répartir les process et tâches sur les cœurs. (-f mpirun-param)

Des runs forcés LMDZOR MPI-OpenMP ont tourné sur vargas, platine, titane et jade : rapports CICLE (performances)

Des runs couplés IPSLCM4\_v2 MPI-OpenMP ont tourné sur platine et jade : rapports CICLE (performances) et Grand Challenge CINES. Sur vargas et titane, le mode de lancement est encore à trouver.

Ce à quoi on veut arriver : extraire IPSLCM5A (ou LMDZOR de reference ou ...) sur titane (ou vargas ou jade ou NEC ...) le compiler (via modipsl), lancer une simulation (avec libIGCM) avec un certain nombre de processus MPI et de tâches OpenMP (définis par défaut à l'optimal ou modifiables).

Ce qu'il manque :

- Modèles : ORCHIDEE MPI-OpenMP en phase avec Orchidee\_1\_9\_5 (version CMIP5)
- Compilation : gestion du mode MPI, MPI-OpenMP, OpenMP via modipsl : cles cpp (ORCHIDEE\_NOOPEMP pour LMDZ), options du makegcm\_fcm et configure\_orchidee gérées, lien avec exécution (nombre de process MPI pour NEMO, cohérence entre compilation et exécution,...)
- Exécution :
  - lancer une configuration optimale MPI-OpenMP définie par défaut. Possibilité de choisir un nombre de process MPI et tâches OpenMP différents de la configuration par défaut.
  - lien avec la compilation
  - tourner une configuration couplée mixte MPI-OpenMP sur vargas et titane.

Etapes :

Point de départ : IPSLCM5A figé.

**Etapes préliminaires** à faire en parallèle (car indépendantes les unes des autres) :

Cahier des charges du point de vue d'un utilisateur : qu'est ce qu'on veut pouvoir faire exactement avec les outils modipsl et libIGCM pour différentes configurations ? Par ex : définition dans config.card du nombre de processus MPI et nombre de tâches OpenMP par composantes, compilation choisie facilement (gmake LMDZ9695 MPI-OpenMP,...)  
LMDZINCA est une bonne configuration de travail.

Noms : Anne, Olivier, Martial, Yann, Arnaud, Marie-Alice

Version CMIP5 d'Orchidee avec OpenMP disponible, compatible

Noms : Martial, Yann

Configuration couplée sur titane et vargas : lancement d'un code MPMD avec un composant mixte MPI-OpenMP.

Noms : Yann, Arnaud, Marie-Alice

**Point de rencontre** : on sait ce qu'on veut, on connaît les contraintes des machines, les modèles sont à jour : création d'un prototype IPSLCM5A\_OMP.

**Point final** : validation, analyse de performances du prototype, documentation.

**Important** : un point sur les avancées OpenMP à chaque réunion ESCI (comme CMIP5).