



Séminaire du Jeudi 10 Septembre 2015, 10h, TGCC.

**Improving the Hybrid model MPI+Threads through Applications,
Runtimes and Performance tools**

Jeudi 10 Septembre 2015, Aurèle Mahéo, Doctorant au Laboratoire Exascale Computing Research, nous présentera ses travaux lors d'une répétition de sa soutenance de thèse.

Voici le résumé de cette présentation qui aura lieu dans l'**Amphi du TGCC** à 10h.

Improving the Hybrid model MPI+Threads through Applications, Runtimes and Performance tools.

Afin de répondre aux besoins de plus en plus importants en puissance de calcul de la part des applications numériques, les supercalculateurs ont dû évoluer, intégrant notamment des architectures dites NUMA (Non Uniform Memory Access), et sont de plus en plus compliqués à programmer. Une autre contrainte, la diminution de la mémoire disponible par coeur de calcul, doit être soulignée.

C'est ainsi que des modèles parallèles tels que MPI (Message Passing Interface) doivent être combinés avec d'autres modèles plus adaptés aux systèmes à mémoire partagée afin de permettre aux codes scientifiques d'exploiter efficacement ces machines. OpenMP, en tant que modèle standardisé, est un choix privilégié pour être combiné avec MPI. Mais mélanger deux modèles différents est une tâche compliquée et peut engendrer des goulets d'étranglement. Cette thèse a pour objectif d'aborder ces limitations et met en avant plusieurs contributions couvrant divers aspects. Notre première contribution permet de réduire le surcoût des supports exécutifs OpenMP pour les codes MPI+OpenMP en optimisant le travail d'activation et de synchronisation des threads OpenMP. Dans un second temps, nous nous focalisons sur les opérations collectives MPI et proposons une technique permettant de réutiliser des coeurs de libres à l'aide des threads OpenMP. Nous introduisons également le concept de collectives unifiées, faisant intervenir des tâches MPI et des threads OpenMP dans une même opération. Enfin, nous nous intéressons à l'instrumentation des codes MPI+OpenMP, notre dernière contribution consistant en l'implémentation et l'utilisation de l'outil OpenMP Tools API dans le support exécutif OpenMP du framework MPC.
