Foreword

First Workshop on Co-Scheduling of HPC Applications (COSH 2016)

Prague, Czech Republic, January 19, 2016
Co-located with HiPEAC 2016

Welcome from the Organisers

Welcome to COSH 2016, the first workshop on Co-Scheduling of HPC Applications! The workshop is held in conjunction with the HiPEAC 2016 conference in the wonderful medieval city of Prague, Czech Republic. Holding the workshop for the first time, we received nine submissions from four different countries. Out of these, the programme committee selected six high quality papers for publication. In the process, each paper received three reviews. In addition, we very much appreciate a keynote by Christopher Dahnken of Intel Corporation on recent hardware support and latest developments in microprocessor technology within the co-scheduling context.
We are grateful for the outstanding job of our programme committee which managed to return all reviews within the very tight time constraints. We hope this workshop will lead to new insights and fruitful discussions around its relatively novel topic within the context of High Performance Computing.

Munich, December 2015

Carsten Trinitis, Josef Weidendorfer
Workshop Co-Chairs

Workshop Description

The task of a high performance computing system is to carry out its calculations (mainly scientific applications) with maximum performance and energy efficiency. Up until now, this goal could only be achieved by exclusively assigning an appropriate number of cores/nodes to parallel applications. As a consequence, applications had to be highly optimised in order to achieve even only a fraction of a supercomputer's peak performance which required huge efforts on the programmer side.
This problem is expected to become more serious on future exascale systems with millions of compute cores. Many of today's highly scalable applications will not be able to utilise an exascale system's extreme parallelism due to node specific limitations like e.g. I/O bandwidth. Therefore, to be able to efficiently use future supercomputers, it will be necessary to simultaneously run more than one application on a node. To be able to efficiently perform co-scheduling, applications must not slow down each other, i.e. candidates for co-scheduling could e.g. be a memory-bound and a compute bound application.
Within this context, it might also be necessary to dynamically migrate applications between nodes if e.g. a new application is scheduled to the system. In order to be able to monitor performance and energy efficiency during operation, additional sensors are required. These need to be correlated to running applications to deliver values for key performance indicators.

Main topics

Exascale architectures, supercomputers, scheduling, performance sensors, energy efficiency, task migration