Towards system-scale optimisation of HPC applications

TADaaM: Topology-Aware System-Scale Data Management for High-Performance Computing Applications

Emmanuel Jeannot

October 2016
 INTRODUCTION

Optimize application execution at system-scale

Topology

Data

Applications

Tadaam, october 2016

Emmanuel Jeannot - 2
Outline

1. Context and problematic
2. Scientific challenges
3. Software and use-cases
4. Conclusion
1 Context and Problematic
Computing is easy, accessing data is difficult

Lot of computing power.

Bringing data at the right place at the right time is the challenge.

Flops are free but bytes are expensive!
Stacking Optimized Library and Runtime Systems

- Multithreaded application
- Multithreaded Comp. Library
- Multithreaded Runtime System
- Hardware

MPI (progress threads)

Scientific app
Parallel Blas
OpenMP
Multicore+parallel

Pb: Each thread ignore the existence of the other threads!
Mapping? Priority? Scheduling?

Tadaam, october 2016
Emmanuel Jeannot - 6
Platform partitioning

Pb: message transfer not aware of other applications!
Contention, routing, message scheduling

What is missing?

A “thing” that allows for managing data by doing:

• Cross-layer optimizations
• System-wide optimizations
How application can make the best possible use of the available resources

Problematic:
- Allocate data
- Partition data
- Reserve resources
- Control affinity
- Map computation
- Manage contention
- Optimize communication
- Access storage
- Perform visualization
Our approach: An intermediate service layer for optimizing execution

Programming Model

Stateful System-wide Service Layer

Application a
Application b

Memory hierarchy
Cache size
Network topology
Allocated resources
Other applications
Storage

Hardware
Applications needs

Application can express its varying needs for:

- Memory usage
- Computation
- Network access
- Storage
- Affinity
- Model/data refinement
- etc.
2 Scientific challenges
The application within its ecosystem

- Applications
- Programming models
- Hardware
- SW stack
- Runtime systems
- Operating systems
- Libraries
- Network
- Storage

Environment model
Application need and model
Optimization algorithm
Optimized execution
Challenges

We need:

• A layer based on models and abstractions (application and environment)
• System-wide services that take into account the whole ecosystem at scale
• A stateful optimization engines
3

Software and use-case
Mesh-based High-performance computing applications

Most of the large-scale applications (at least 2/3 in last PRACE call) use meshes:

- domain decomposition
- stencil
- unstructured
- hierarchical
- etc.

Ex: aerodynamic, climate, electromagnetism, seismology, plasma, etc.
Software suite: use-case example

Mesh/graph partitioning (Scotch)
Platform model (Hwloc)
Topology-aware locality mechanisms (TreeMatch)
Parallel mesh adaptation (Pampa)
Communication optimization (New Madeline)
4 Conclusion
System-wide topology-aware data management

Machines are more complex and applications require to be executed at large-scale.

Need for cross-layer and system-wide optimizations

Target mesh-based applications.

Design, implement, deploy a stateful, system-wide service layer to:

• Optimize application execution
• According to its needs
The TADaaM Team

Emmanuel Jeannot, senior research scientist (DR2), Inria, Team leader;
Guillaume Aupy, Research scientist (CR2), Inria
Alexandre Denis, experienced research scientist (CR1), Inria;
Brice Goglin, experienced research scientist (CR1), Inria;
Guillaume Mercier, assistant professor, Bordeaux Institute of Technology;
François Pellegrini, professor, University of Bordeaux;
Raphaël Blanchard, PhD student, CIFRE Onera;
Cyril Bordage, Postdoc, COLOC, Inria;
Remi Barat, PhD student, CIFRE, CEA;
Nicolas Denoyelle, research engineer, COLOC, Inria;
Clément Foyer, Engineer, ELCI, Inria;
Cédric Lachat, post-doc, ELCI, Inria;
Benjamin Lorendeau, PhD student, CIFRE, EDF;
Farouk Mansouri, Post-doc, Inria,
Adèle Villiermet, PhD student, COLOC, Inria. ;
Hugo Taboada, PhD student, CEA;

Cécile Boutors, Team assistant.
Thanks!