D7.5 Deliverable contribution guidelines for Subtask 7.5.D

Chapter D: "Multi-core/many-core systems"

10-15 pages total; Contributors: all

Contribution from SNIC-LIU :

Modern multi-core nodes posses challenge for distributed computing. In many cluster configuration each node is connected by a single interconnect, e.g., inifiniband cable. It creates a blocking configuration for MPI communication. As an example for a 32 core node when all the ranks try to communicate outside node through a single interconnect cable it creates a 1:32 blocking situation. This might impact the data transfer rate and consequently the scalability of codes. On the other hand in multi core nodes often intra-node bandwidth / latency is better than inter-node interconnect bandwidth / latency. Thus it is important to write topology aware algorithms which exploit intra-node bandwidth / latency more and less reliant on inter-node bandwidth / latency. For MPI collective operations it is possible to rewrite the collectives in a topology aware way. In the project - “Improving MPI communication latency on Euroben kernels” we have demonstrated the impact of topology aware MPI_Alltoall routine on the timing of a kernel called mod2f.

D.1 Introduction and summary

3 pages; Contributors: Antun Balaz, Vladimir Slavnic

D.2.Improving MPI communication latency on Euroben kernels

One section D.2.X per project X, 1-1.5 pages per project, Contributor: project owner

Contents:

D.2.Improving MPI communication latency on Euroben kernels.1 : Project summary

Project name: Improving MPI communication latency on Euroben kernels

Project authors and their contacts (institutions, e-mails):

Chandan Basu, Linköping University, Sweden, cbasu@nsc.liu.se

WP7 contributors and their contacts (institutions, e-mails):

Chandan Basu, Linköping University, Sweden, cbasu@nsc.liu.se

Effort (pms): 0.8

White Paper, Technical report or Scientific publications

- Title, reference, Authors
- Link (in BSCW or otherwise)

Hardware platform(s): Curie system

Programming language(s): c, MPI

Profiling tools: NA

Libraries: NA

Brief project description (one reasonable paragraph):

On modern many-core / multi-core SMP nodes, introducing network-awareness in the application program through some system dependent parameters, such as number of CPUs/node, number of nodes connected to 1st level switch etc. can reduce the MPI communication latency.

Applications area(s): Applications heavily dependent on collective operations, e.g., MPI_Alltoall

D.2.Improving MPI communication latency on Euroben kernels.2 : Description of work

- Brief description of the contribution to the project, what results are obtained
    - Description of the work should be divided into tasks (e.g. profiling (computation/communication time ratio and code scaling (speedup or execution time) in function of the number of MPI processes and number of threads as one example metric), porting to OpenMP if only MPI version of the code existed, hybridization, optimization, benchmarking and such) and estimate of the effort for each task should be given
- Main problems encountered
- Graphs illustrating the experimental/benchmarking results
- Lessons learned regarding hybridization itself, libraries, hardware platforms (if more than one is used), etc.

D.2.Petascaling enabling and support for EC-Earth3

D.2.Petascaling enabling and support for EC-Earth3.1 : Project Summary

D.2.Petascaling enabling and support for EC-Earth3.2 : Description of Work

D.3 Conclusions

Contributors: Antun Balaz, Vladimir Slavnic