Science & Technology Facilities Council

Multi-core and NEMO

Andrew Porter, Stephen Pickles & Mike Ashworth

Scientific Computing Department,
STFC Daresbury Laboratory, UK

http://www.stfc.ac.uk/scd

Contents

- Introduction to NEMO
- This Work (OpenMP)
- Performance Results
- Conclusions and next steps
- Acknowledgements

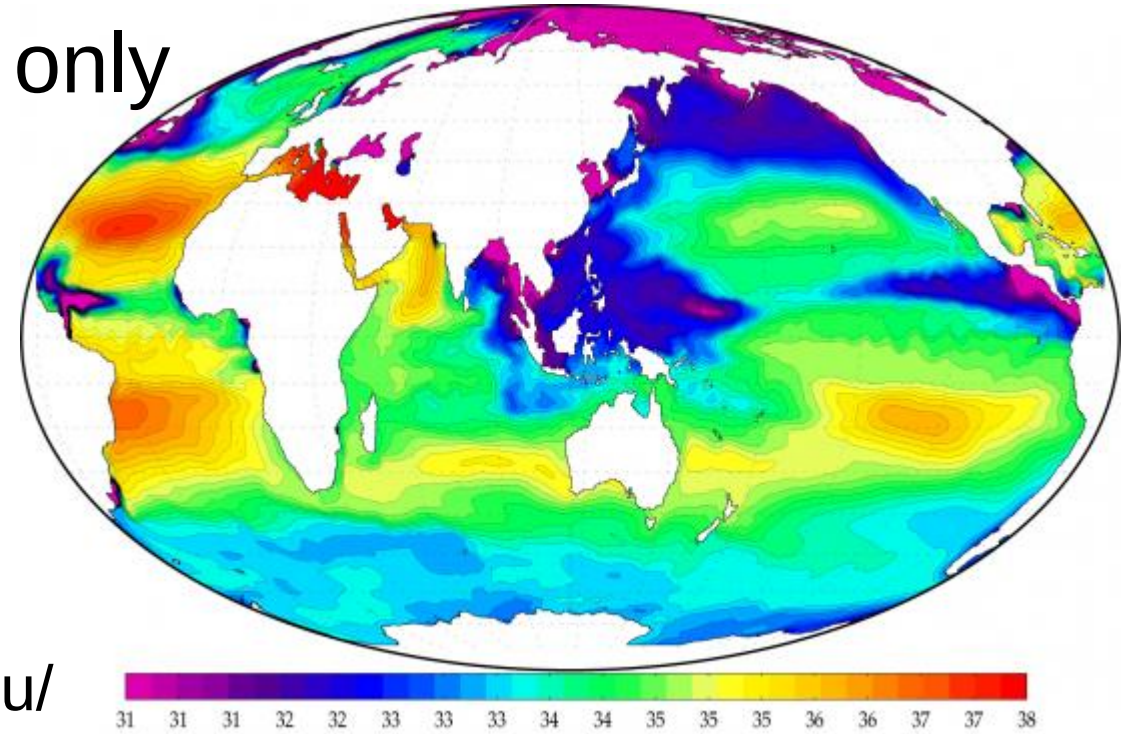
Science & Technology Facilities Council

Contents

- Introduction to NEMO
- This Work (OpenMP)
- Performance Results
- Conclusions and next steps
- Acknowledgements

Introduction to NEMO

- Widely-used European ocean model
- Finite-difference on tri-polar ORCA grid
- Memory-bandwidth bound
- Fortran90, MPI only
- Highly portable
- ~20 years of development



http://www.nemo-ocean.eu/

Contents

- Introduction to NEMO
- This Work (OpenMP)
- Performance Results
- Conclusions and next steps
- Acknowledgements

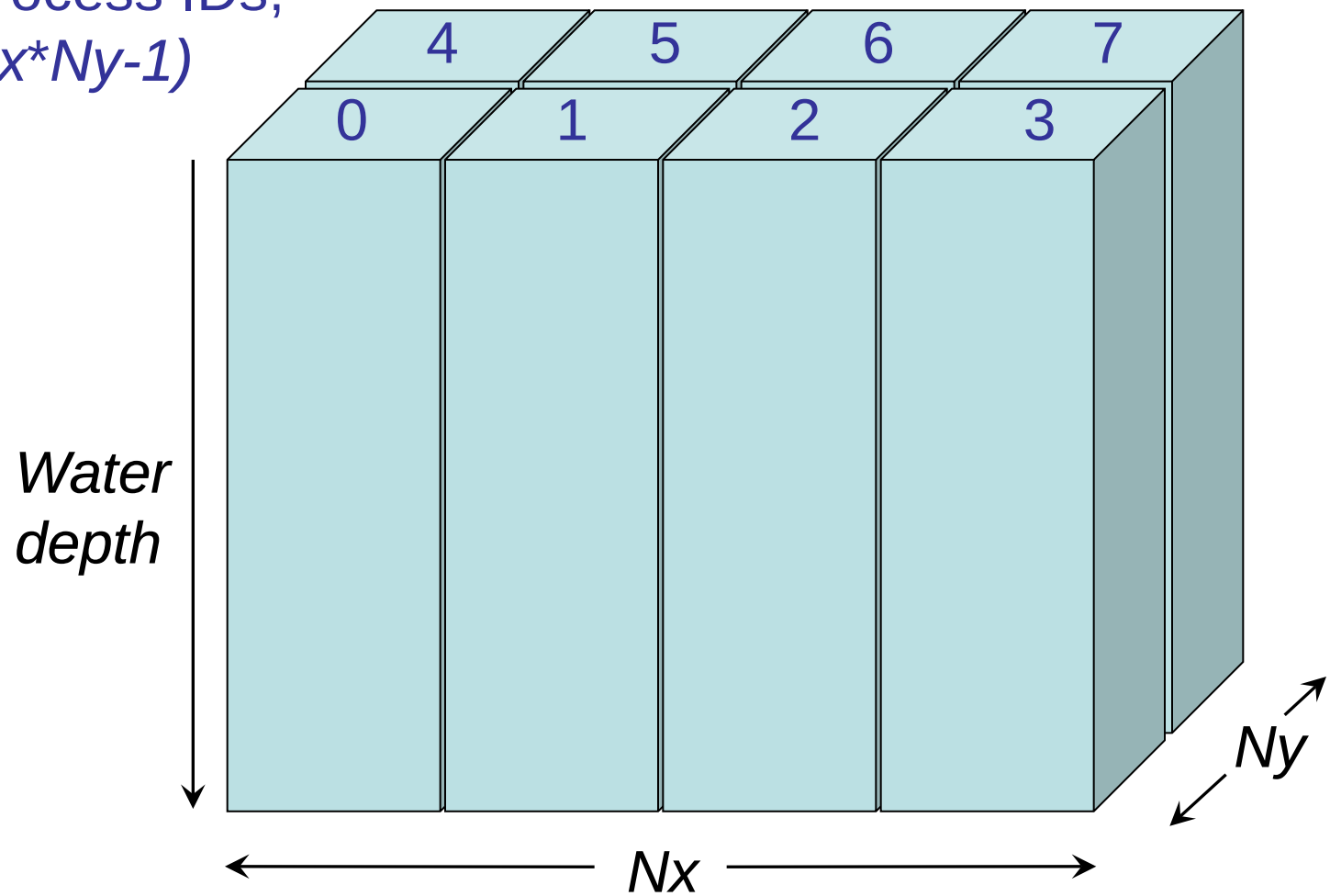
Continue under PRACE 2iP

- STFC is spending 6 PM's in PRACE 2iP looking at different OpenMP strategies in NEMO
  - Effect of array index ordering
  - Loop-level versus coarse-graining/tiling
- Collaboration with Giovanni Aloisio, Italo Epicoco and Silvia Mocavero (European Mediterranean Centre for Climate Change, CMCC)

MPI Domain Decomposition in NEMO

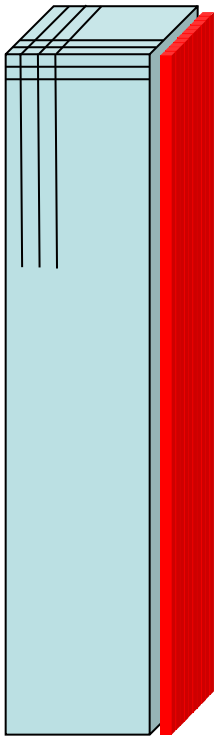
Domain decomposed on regular grid in *x-y* (lat-lon) plane:

MPI process IDs;
0 – (*Nx*Ny-1)*



Array Index Ordering

- Arrays in NEMO are set-up with the *x* index leading and the depth index last
- Limits performance since *x*-dimension of sub-domains shrinks as no. of MPI procs increases
  - This is the dimension of the **vectorisable** inner loop
- Also, pack/unpack of halos involves vertical slabs and thus strides through memory



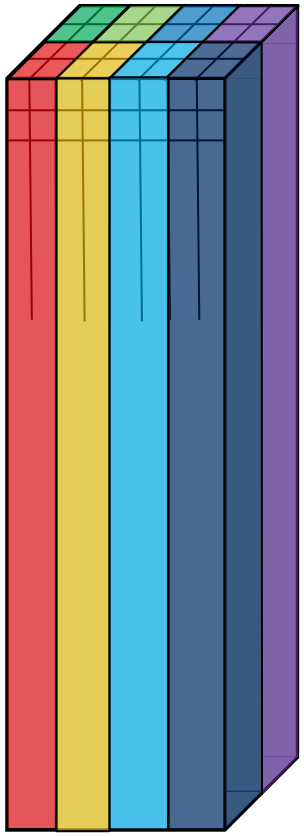
Array Index Re-ordering

- We recently completed a project looking at effects of changing the array index ordering in NEMO
  - Choose *z*-last/*z*-first order at compile time
- Working for two standard NEMO configurations (GYRE and AMM12)
- Not yet clear whether performance benefits out-weigh re-coding pain...
- ...does use of OpenMP inform this?

Collaboration with CMCC

- Silvia Mocavero & Italo Epicolo have implemented loop-level and coarse-grained OpenMP in NEMO
  - Depth index parallelised over threads
- Re-ordering array indices enables us to experiment with decomposing the MPI sub-domains as 2D 'tiles' *c.f.* approach used in WRF
  - Also, retain full depth dimension for **vectorisation**

Single MPI domain decomposed over threads



Contents

- Introduction to NEMO
- This Work (OpenMP)
- Performance Results
- Conclusions and next steps
- Acknowledgements

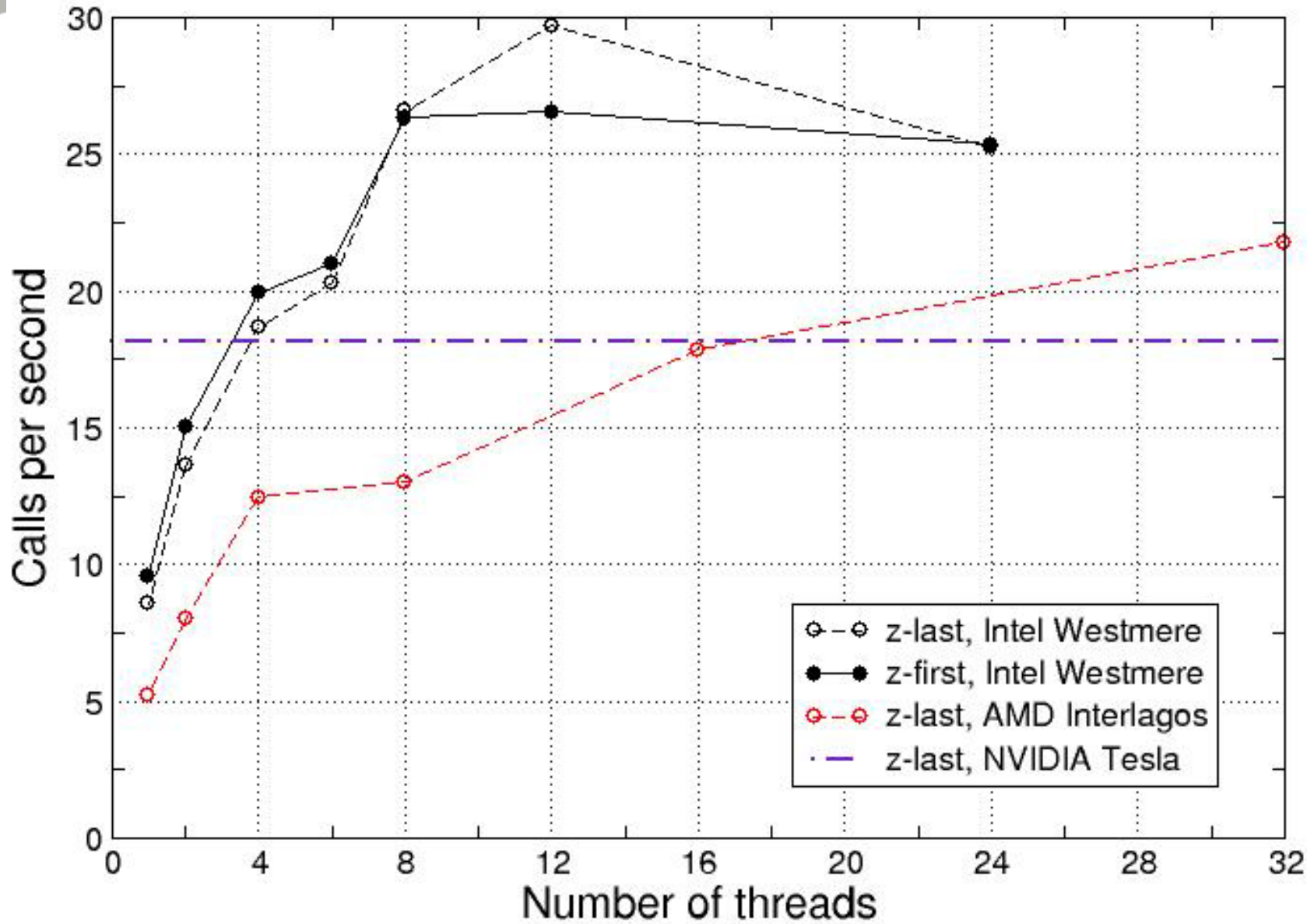
OpenMP Work

- CMCC concentrating on *tracer advection* in 'MUSCL' scheme
- We have z-last and z-first versions of equivalent routine that uses 'TVD' scheme
- Both routines characterised by many triply-nested loops interspersed with halo swaps
- Test system contains two Intel Xeon X5680 (Westmere), each with 6 physical cores
  - Supports 24 hyperthreads in total

Test case details

- ORCA2 grid – 182x149 with 31 levels
- Loop-level OpenMP
  - Outer 2 loops COLLAPSE'd where poss.
- OMP_SCHEDULE=guided eases imbalance caused by halo-swap calls
- No 'first-touch' (ensuring good memory locality) because of above
- Thread->Core binding controlled using KMP_AFFINITY on Intel systems

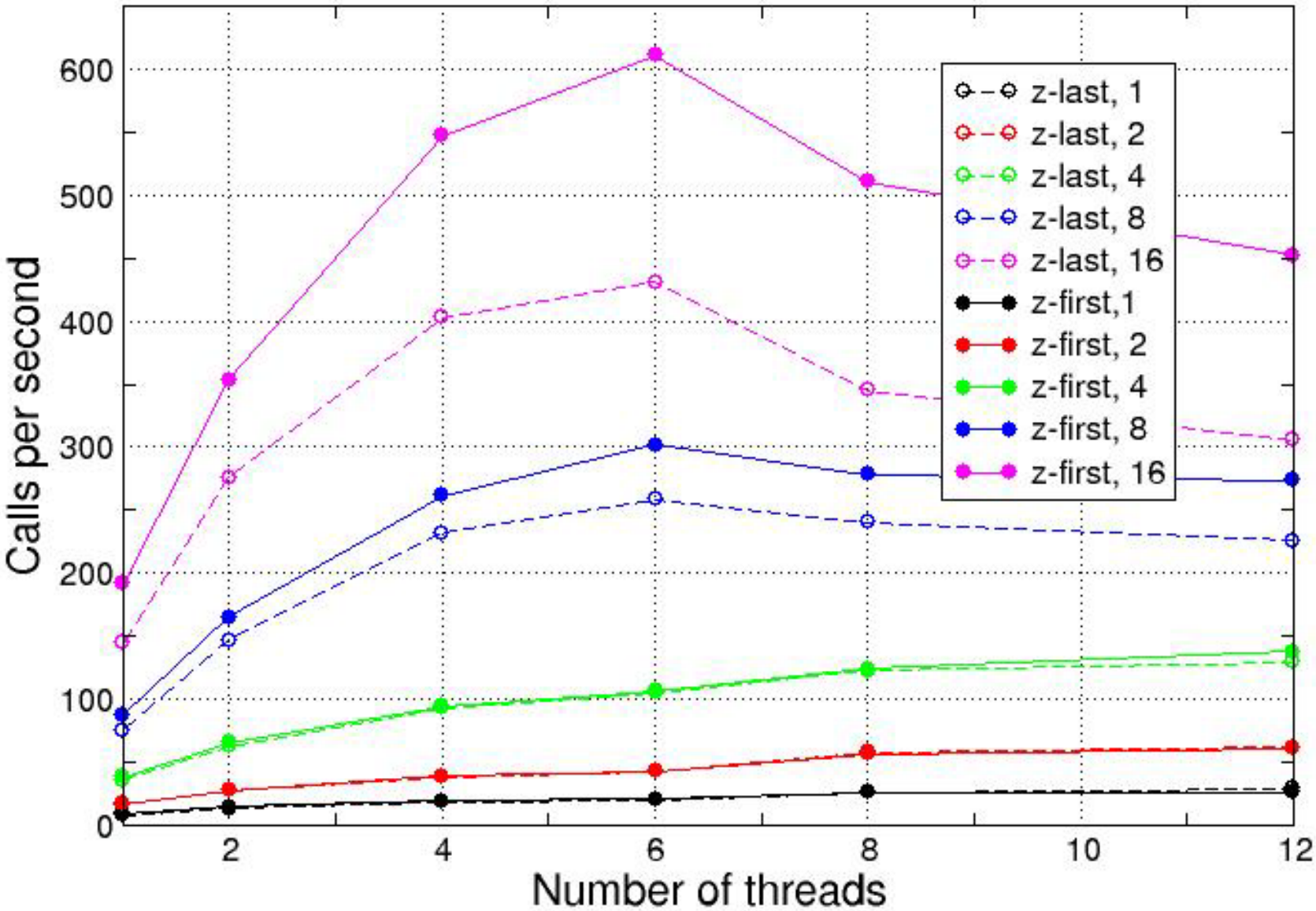
Scaling for ORCA2 case



Notes on results I

- z-first/last versions perform similarly except on largest no. of threads
- One Intel Westmere socket outperforms the NVIDIA GPU
- What happens as we shrink the domain (mimic MPI decomposition)?

Results with fake MPI decomposition in *x*



Notes on results II

- z-first out-performs z-last when x-dimension split over 8 or more ‘procs’
- Performance gain is ~50% for 16 ‘procs’
- Negative effect of running on more than 6 cores (1 socket) clear for 8 or more ‘MPI procs’
  - Memory locality issue?

Science & Technology Facilities Council

Technical report

Google “pickles porter NEMO”

http://epubs.cclrc.ac.uk/work-details?w=63488

Science & Technology Facilities Council

ePublication Archive

Log in | Search | (Advanced Search)

ePubs Home | About ePubs | ePubs Help | ePubs News | ePubs Statistics | Contact ePubs

ePubs Policies | STFC Deposit Guide | STFC Libraries | SHERPA RoMEO | OpenDOAR | Google Scholar

Browse

- by Year
- by Author
- by Affiliation
- by Journal
- by Report Series
- by Department
- by Collaboration
- by Type
- by Title (with fulltext)

Developing NEMO for large multi-core scalar systems: final report of the dCSE NEMO project

Work

Work Id	63488
OAI Item Identifier	oai:epubs.cclrc.ac.uk:work/ 63488
Title	Developing NEMO for large multi-core scalar systems: final report of the dCSE NEMO project
Contributions	Au. *SM Pickles* (STFC Daresbury Lab.), Au. *AR Porter* (STFC Daresbury Lab.)
Departments	

Science & Technology Facilities Council

Contents

- Introduction to NEMO
- This Work (OpenMP)
- Performance Results
- Conclusions and next steps
- Acknowledgements

Conclusions & next steps

- z-first array index ordering can boost OpenMP performance as MPI sub-domains shrink
- Handling halo swaps efficiently in a threaded implementation is a challenge
  - taking explicit control of threading using *e.g*. tiles may help (reserve one thread for MPI)
- Implement tiling rather than loop-level OpenMP

Science & Technology Facilities Council

Acknowledgments

- EPSRC for funding dCSE NEMO Project
- NEMO Systems Team
- UK Met Office for AMM configuration
- DisCo Group at Daresbury for systems

EPSRC
Engineering and Physical Sciences Research Council

NATURAL ENVIRONMENT RESEARCH COUNCIL

If you have been …

… thank you for listening

Mike Ashworth

mike.ashworth@stfc.ac.uk

http://www.stfc.ac.uk/scd