A codesign effort to get ECMWF's IFS model to an extreme O(100) OpenMP threads per MPI task for the [Peta,Exa]Scale

George Mozdzynski (ECMWF), Mark Bull (University of Edinburgh) and Harvey Richardson (CRAY UK)

George.Mozdzynski@ecmwf.int

17th Workshop on High Performance Computing in Meteorology, 25 October 2016, Reading, UK

ECMWF

esiwace
CENTRE OF EXCELLENCE IN SIMULATION OF WEATHER AND CLIMATE IN EUROPE

Acknowledgements

The ESiWACE project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 675191

The CRESTA project 3Q/2012 - 4Q2014 has received funding from the EU Seventh Framework Programme (ICT-2011.9.13), http://www.cresta-project.eu/

An award of computer time was provided by the Innovative and Novel Computational Impact on Theory and Experiment (INCITE13+INCITE14) programs. This research used resources of the Argonne Leadership Computing Facility, which is a DOE Office of Science User Facility supported under Contract DE-AC02-06CH11357.

ECMWF

Outline

- Getting IFS to sustained [Peta,Exa] Scale performance
- A little bit about IFS today
- Motivational reasearch in the CRESTA project (2012-2014)
- comodels an evolutionary OpenMP approach to scaling an IFS model

Getting a future IFS model to sustained [Peta, Exa] Scale performance is dependent on all major developments in progress at ECMWF that are being presented at this workshop (and elsewhere) – a huge team effort

Sadly, there is no magic bullet!

But we are still looking just in case

Technology applied at ECMWF for the last 30 years …

- A spectral transform, semi-Lagrangian, semi-implicit (compressible) hydrostatic model
- How long can ECMWF continue to run such a model?
- IFS data assimilation and model must EACH run in under ONE HOUR for a 10 day global forecast

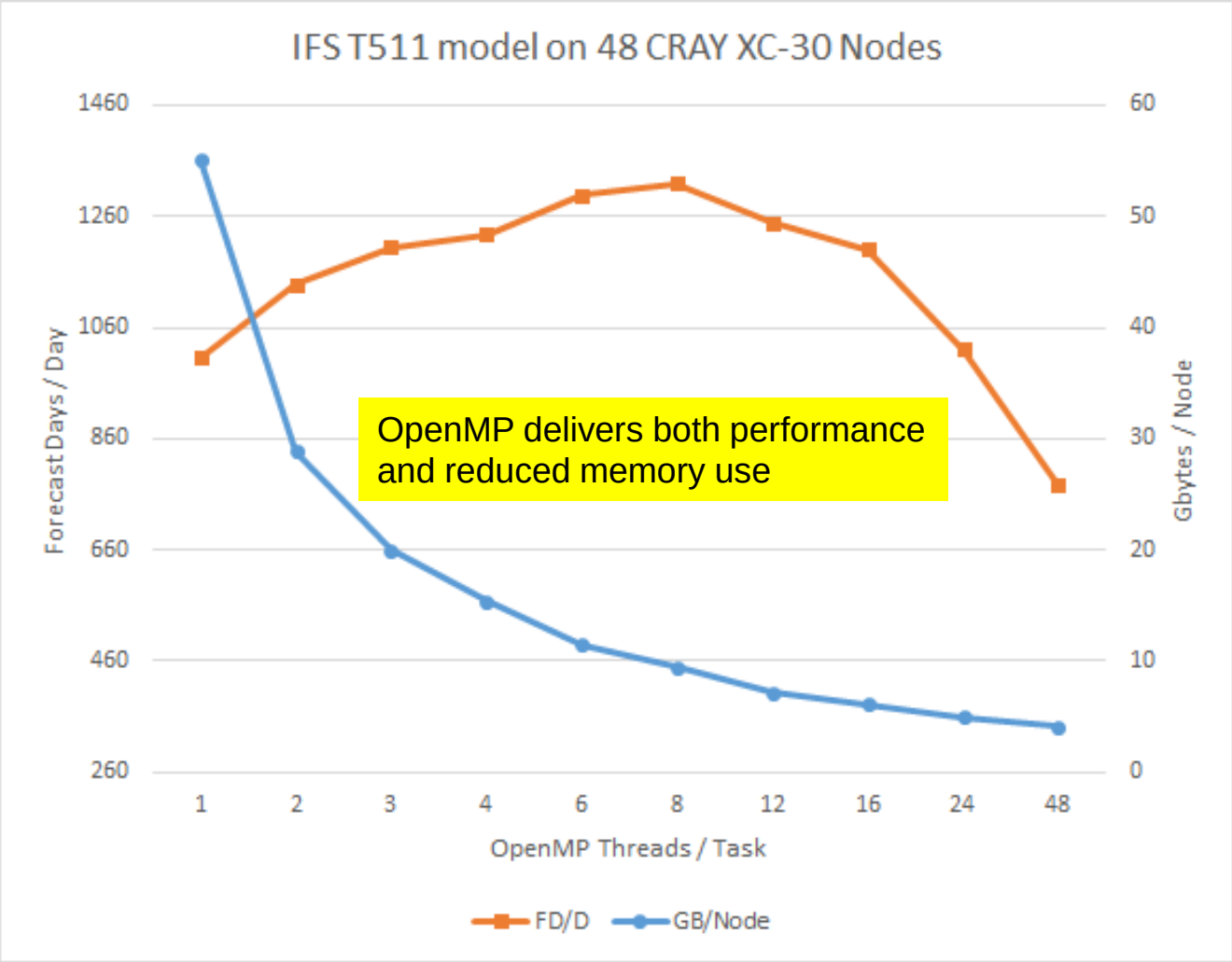
IFS ‘today’ (MPI + OpenMP parallel)

IFS = Integrated Forecasting System

Graphic shows the TCo1279 (9km global) IFS grid space domain decomposition with 1,600 MPI tasks (12 threads / task) on 400 XC-30 Ivybridge nodes implemented operationally in April 2016. All nodes now XC-40 Broadwell.

Within the EU funded CRESTA project we have run a TC3999 2.5km global IFS model on TITAN with up to 28,672 MPI tasks each with 8 cores (OpenMP threads), for a total of 229,376 cores.

ECMWF

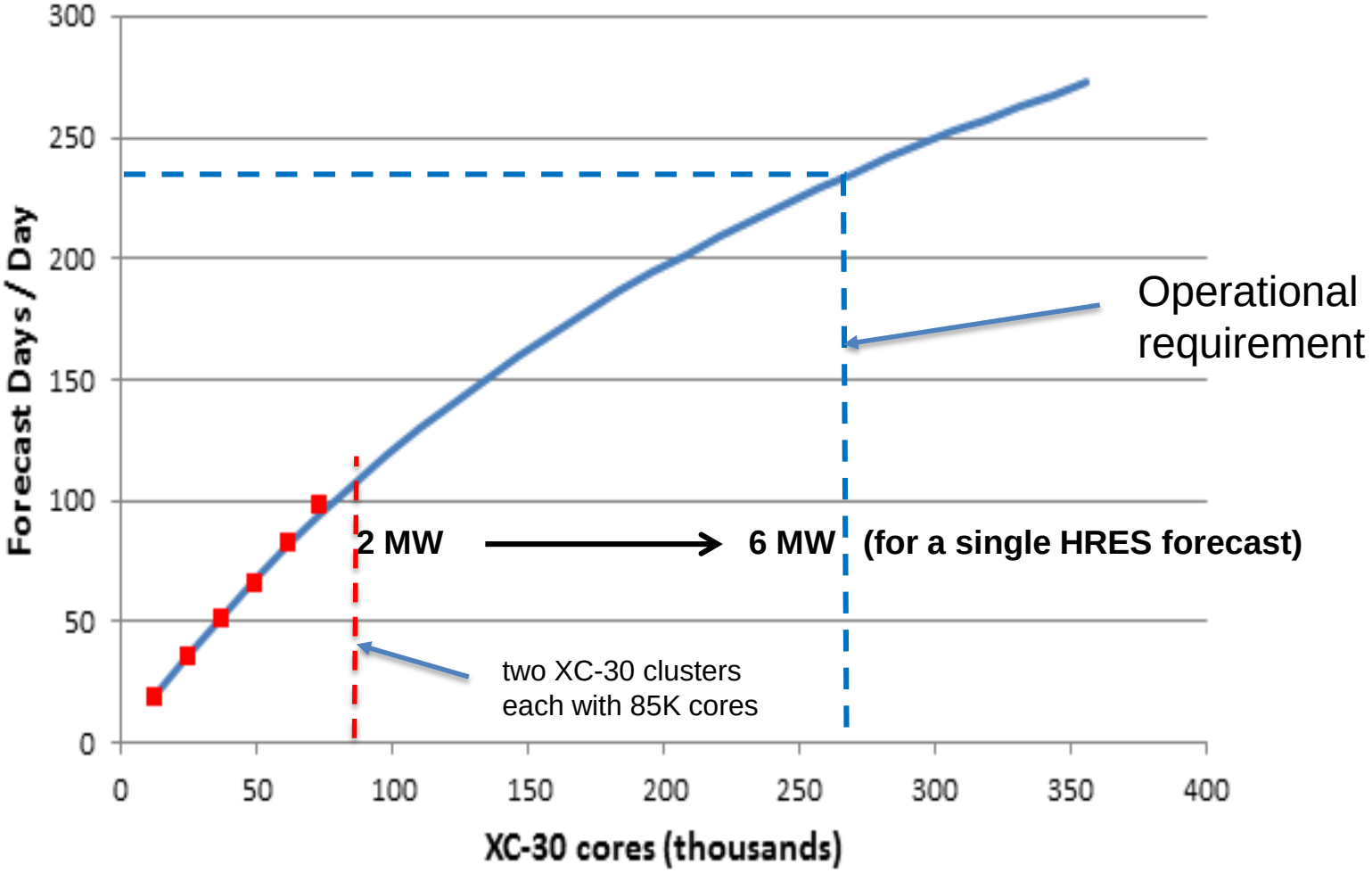


Performance and memory utilisation of an IFS TL511 model on 48 CRAY XC-30 nodes. The blue curve denotes Gigabytes of required memory per node whereas the orange curve shows simulated forecast days per wall clock time day.

ECMWF

For models >= T1279 **12** threads / task gives optimal performance

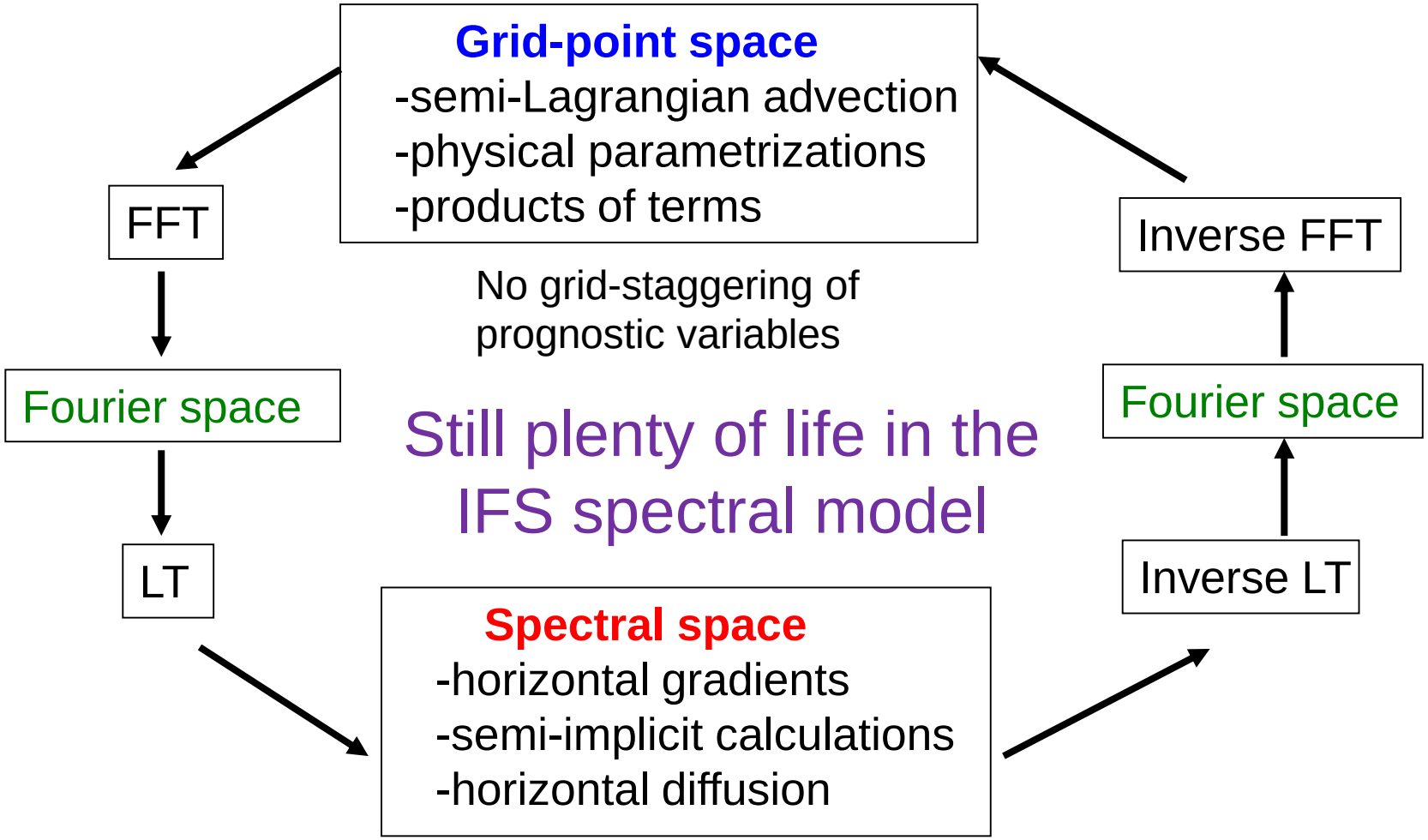
Predicted 2.5 km model scaling on a XC-30



ECMWF require system capacity to run simultaneously 10 to 20 HRES forecasts

ECMWF

Schematic description of the spectral transform method in the ECMWF IFS model (single time-step)

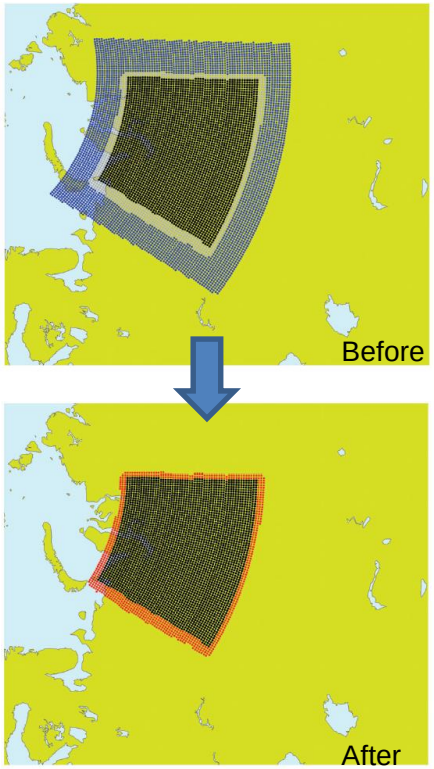


FFT: Fast Fourier Transform, LT: Legendre Transform

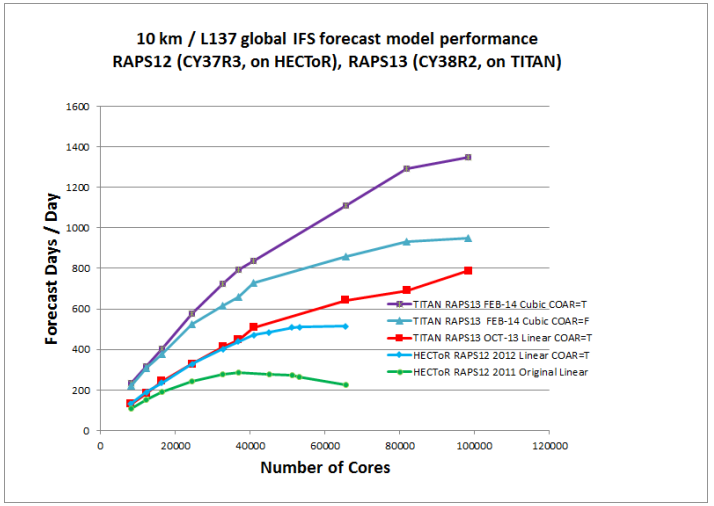
Incremental developments / research in the CRESTA project (2012-2014)

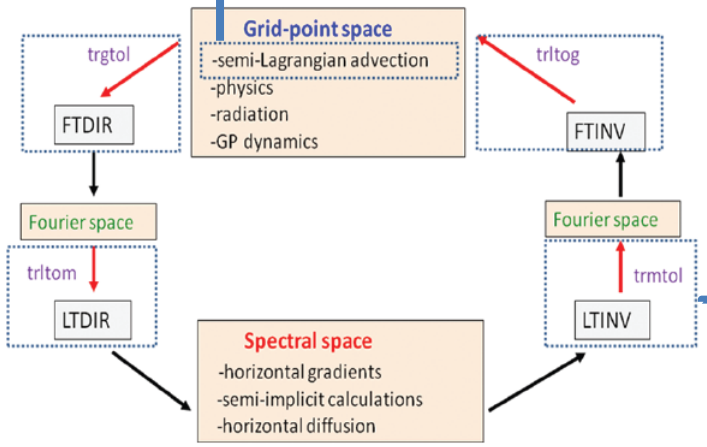
- Overlapping communication and computation using Fortran 2008 coarrays
- Radiation in parallel (separate model and radiation tasks)
- Scaling IFS to 229K cores on TITAN (5km and 2.5km global)
- Application of Fast Legendre Transform
- Initial development of the Atlas library to support an alternative dynamical core option
- Spectral Transform test on TITAN GPUs

CRESTA: Incremental model upgrades



MPI halos defined during runtime





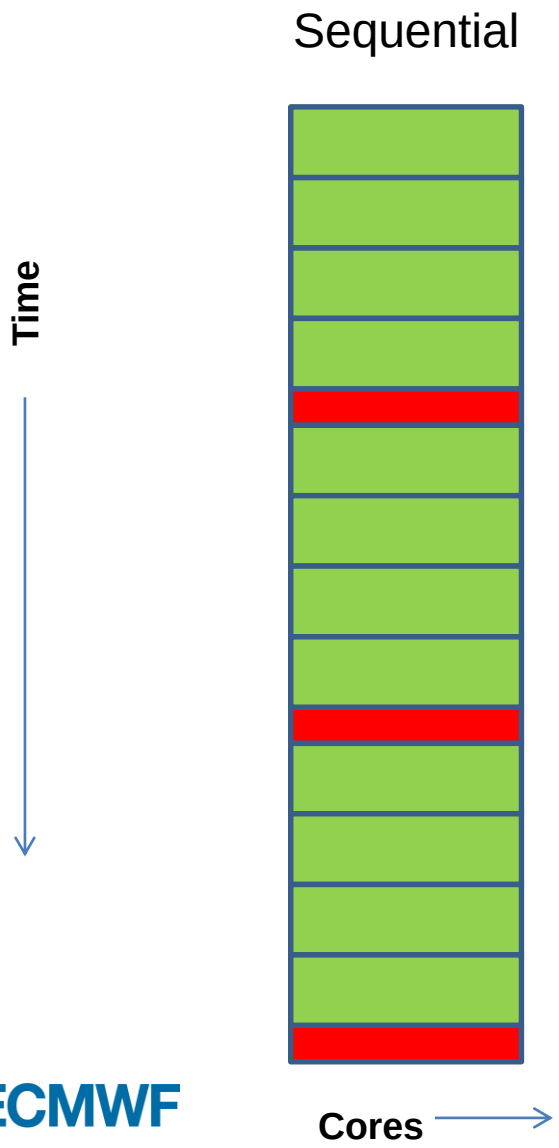
Overlap of computation and communication (the dotted boxes) using Coarrays in Fortran 2008

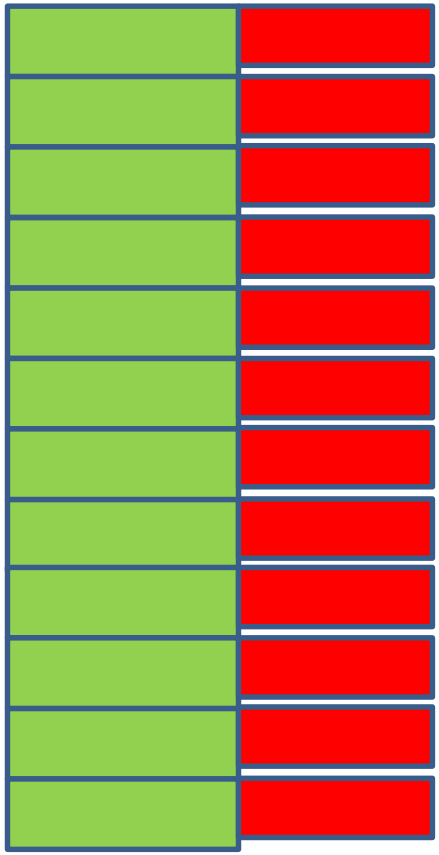
[Mozdzynski et al. 2015: A PGAS implementation of the ECMWF IFS. Int. J. High-Perf. Comp. App.]

ECMWF

CRESTA : Radiation in Parallel (using separate MPI tasks, 2013-2014)

(Radiation computations, model)





http://www.ecmwf.int/sites/default/files/elibrary/2014/11300-reorganization-radiation-transfer-calculations-ecmwf-ifs.pdf

ECMWF

Papers/Technical Memoranda

https://www.researchgate.net/publication/285711190_The_modelling_infrastructure_of_the_Integrated_Forecasting_System_Recent_advances_and_future_challenges

Mozdzynski G, Morcrette J-J, ECMWF Technical Memorandum 721, http://www.ecmwf.int/sites/default/files/elibrary/2014/11300-reorganization-radiation-transfer-calculations-ecmwf-ifs.pdf

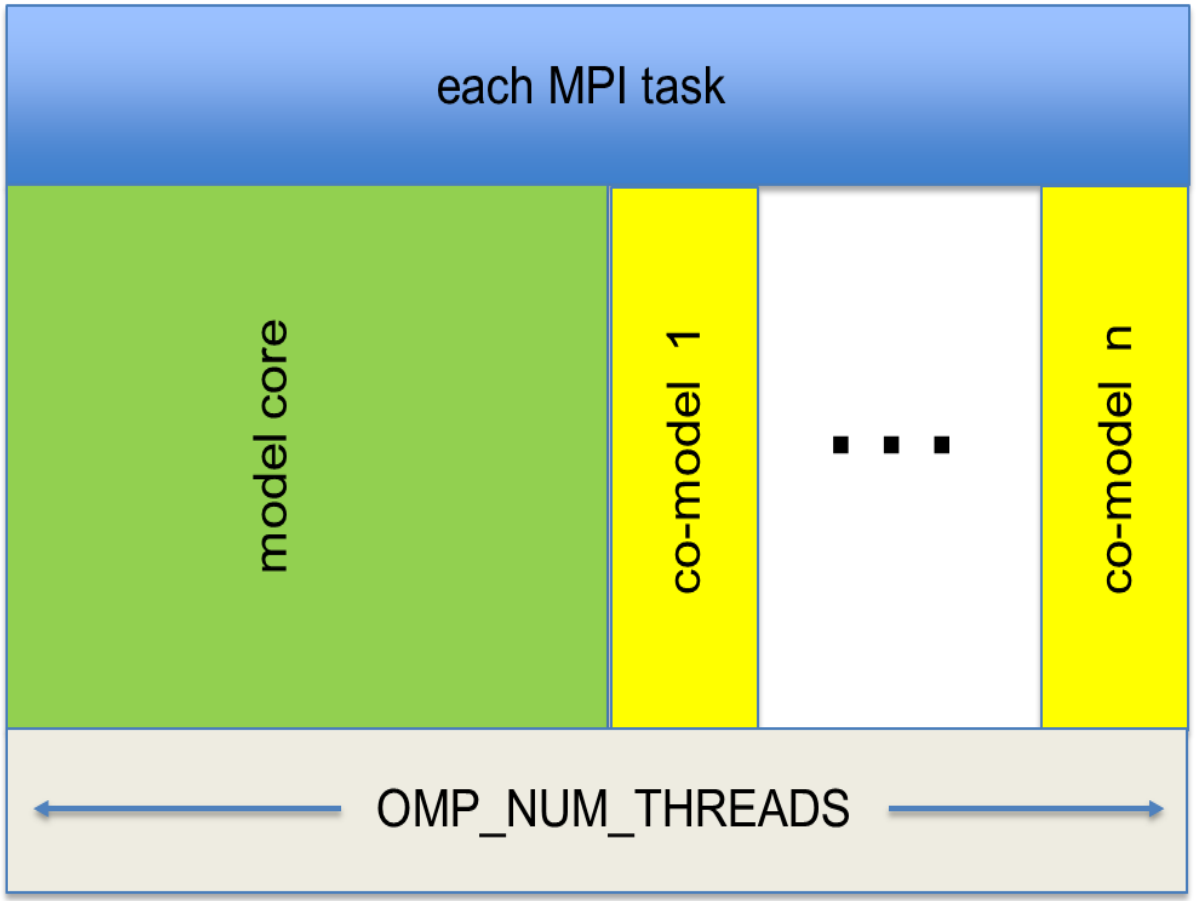
Mozdzynski, G., M. Hamrud, N.P. Wedi (2015), A Partitioned Global Address Space implementation of the European Centre for Medium Range Weather Forecasts Integrated Forecasting System, Int. J. High Perform. Comput. Appl., 29(3), 261-273

Mozdzynski G., A new partitioning approach for ECMWF's integrated forecasting system (IFS), p 148-166 in Proceedings of the Twelfth ECMWF Workshop: Use of High Performance Computing in Meteorology, 30 October – 3 November, 2006, Reading, UK, World Scientific (2007) 273 pp.

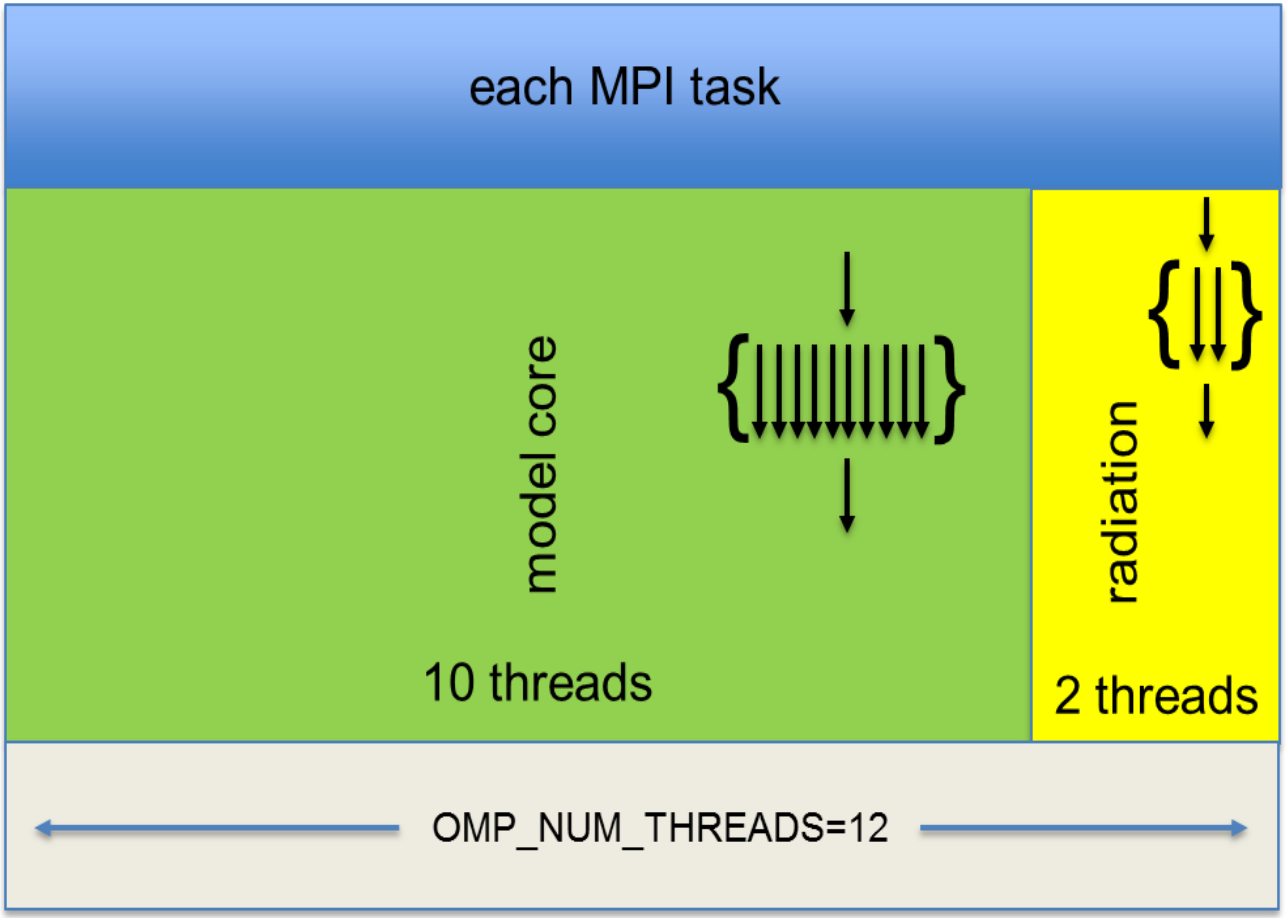
comodels – a potential future ESM architecture to exploit extreme parallelism with O(100) threads per MPI task

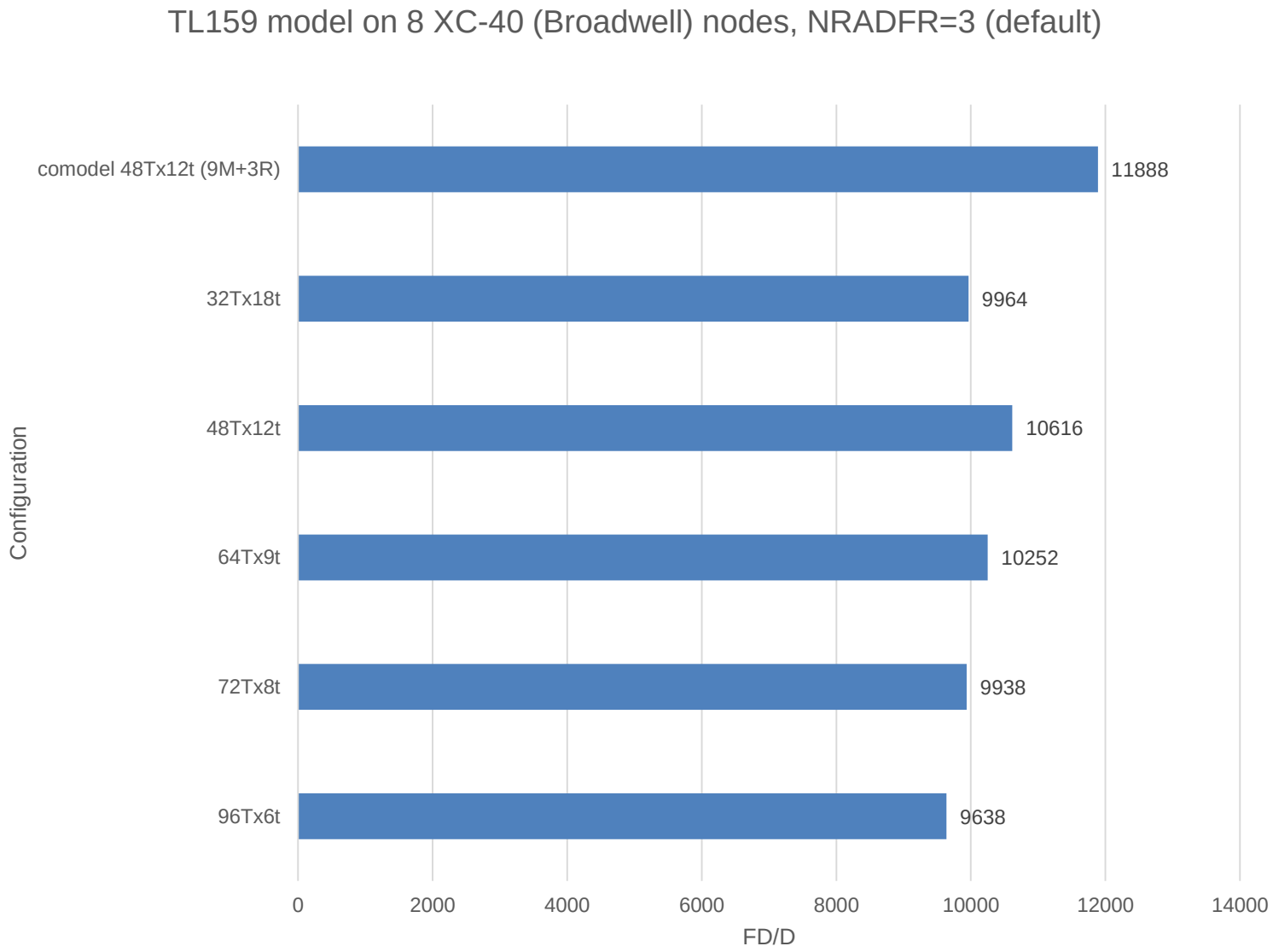
- Example comodels (Dynamics, Physics, Wave, Radiation, Ocean, Land Surface, Ice, etc.)
- comodels execute in parallel with other comodels using data from previous time step
- Implemented using existing OpenMP 4.0 capabilities
- comodels all run on the same set of MPI tasks
- No additional MPI communication required
- Individual comodels can still have message passing internal to their comodel and OpenMP
- Number of threads to be used in each comodel a practical choice
- Xeon Phi, GPUs and conventional servers can all be supported
- ECMWF “Radiation in parallel implementation” explored this concept (lagged model/radiation scheme but with separate MPI tasks for model and radiation), this required additional MPI communication, was bad for performance, bad for code maintenance, inflexible, overall a bad approach

comodels generic structure



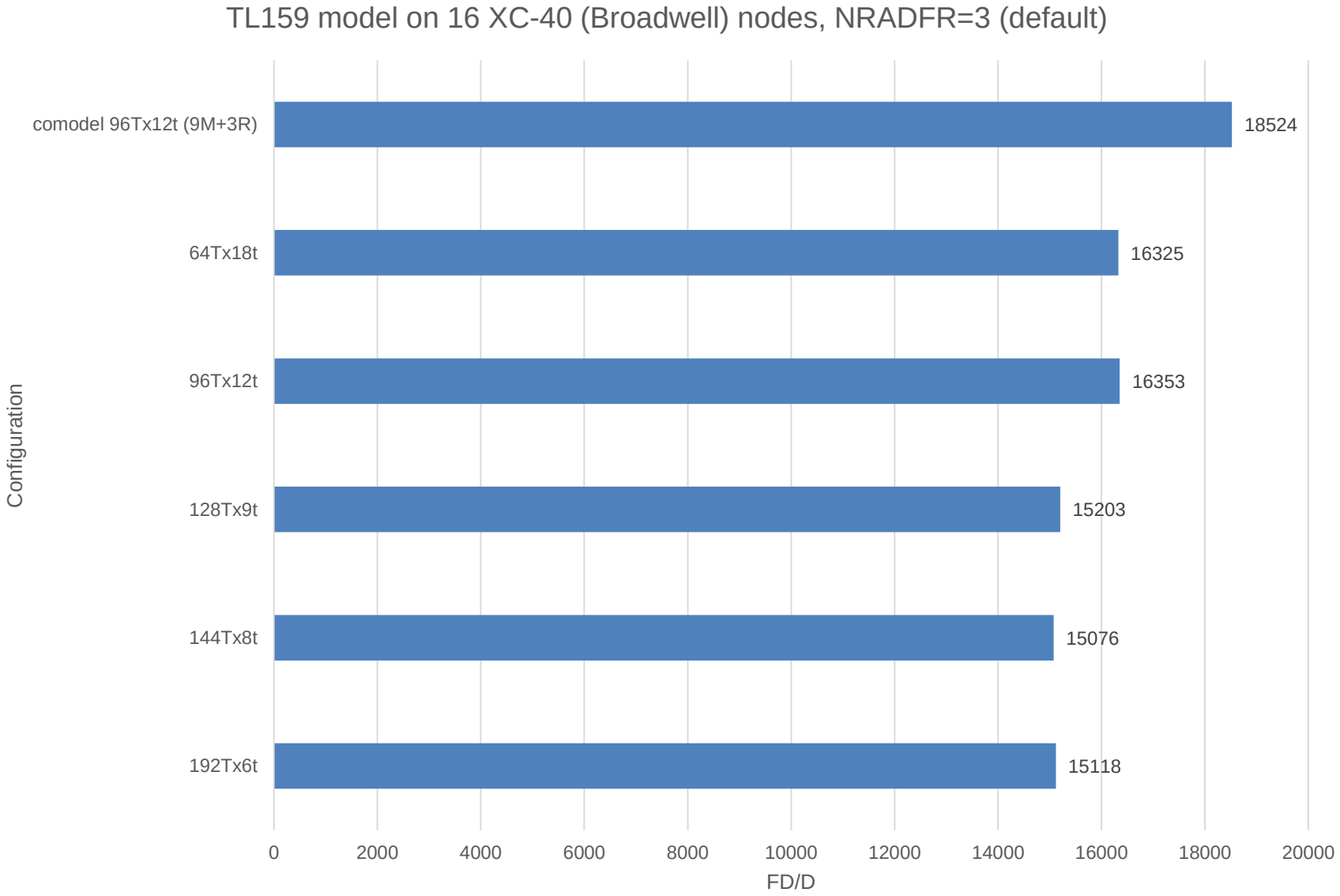
comodels initial prototype example (model + radiation)





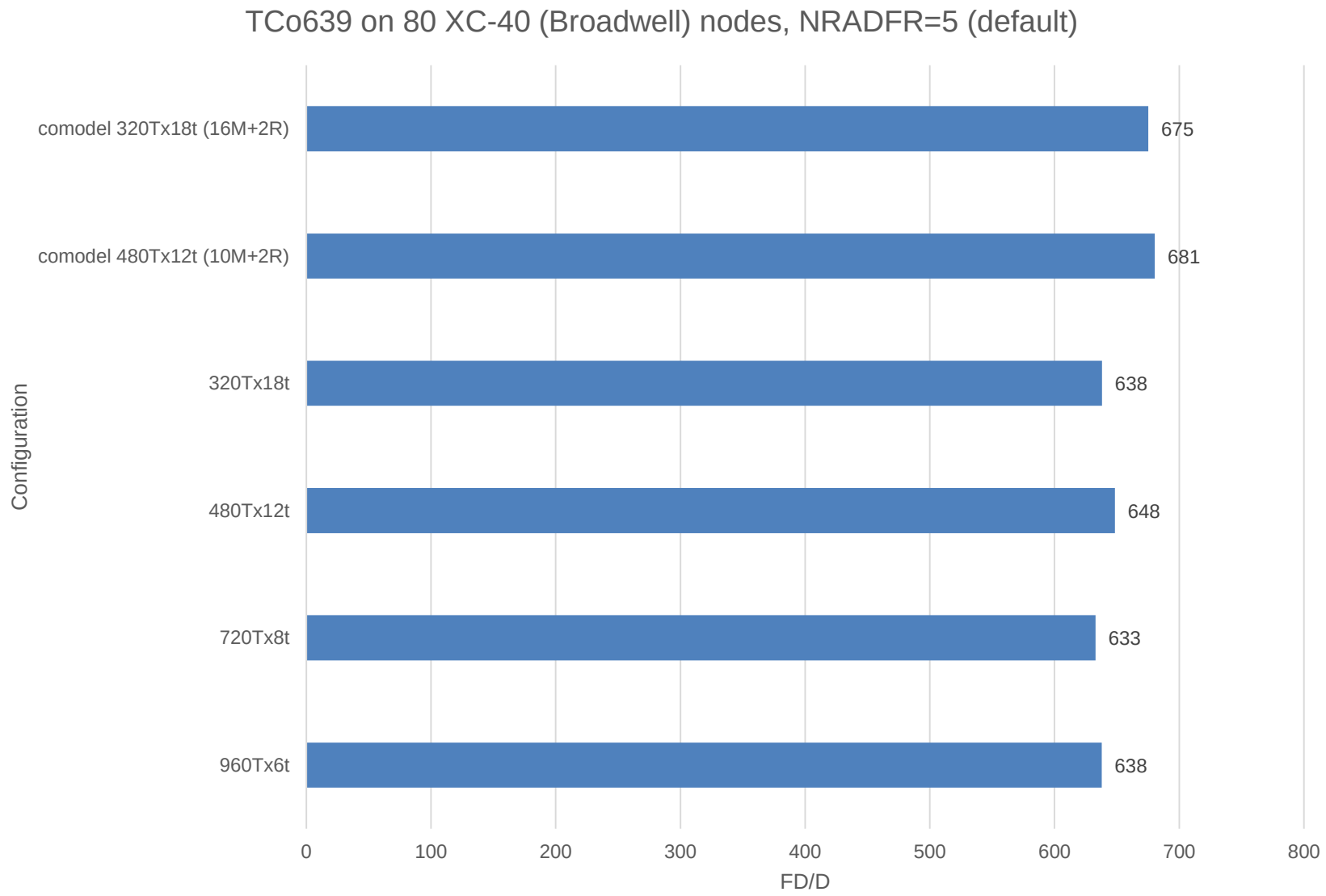
11888/10616=1.120

ECMWF



18524/16353=1.133

ECMWF



681//648=1.051

ECMWF

Some implementation details

- OMP thread synchronisation
  - between model ‘core’ master thread and each comodel master thread
  - technically challenging
  - a lot of offline testing
  - thank you to Mark Bull and Harvey Richardson for all the much needed support
- using OMP_NESTED
- comodels use data from previous model ‘core’ synchronisation time-step
  - using ‘old’ data could be problematic for low resolution cases but not expected to an issue for a high-res ESM
  - e.g. radiation computations are run every NRADFR model time-steps (e.g. a model hour or potentially 30 minutes in the future)
  - requires some careful synchronisation for the initial synchronisation time-steps

Future work

- make comodel synchronisation more modular
  - retain diagnostics to show where model ‘core’ and/or comodels have excessive synchronisation wait times
  - To help finding optimal thread distribution
- initial meteorological evaluation for radiation comodel (1Q2017)
  - expecting no surprises here as this was done before for “radiation in parallel” MPI task implementation (see TM721)
- add the IFS wave mode (WAM) as the second comodel (1H2017)
  - a first example of a comodel doing MPI communication internally
  - may need MPI ‘thread-multiple’ MPI support
- explore alternative synchronisation using OMP tasking
  - comodel thread distribution could then be (potentially) fully dynamic


Thank you for your attention
Questions?