Arctic Region Supercomputing Center

Benchmarking and Evaluation of the Weather Research and Forecasting (WRF) Model on the Cray XT5

Don Morton, Oralee Nudson and Craig Stephenson

Arctic Region Supercomputing Center

University of Alaska Fairbanks

HPC Modernization Program

UAF University of Alaska Fairbanks

Overview and Outline

- ***Introduction of the ARSC WRF Benchmark Suite***
- ***Initial testing of WRF on the ARSC Cray XT5, pingo***
- ***Attempts to push the limits with huge WRF domains (billion grid points)***

Acknowledgements

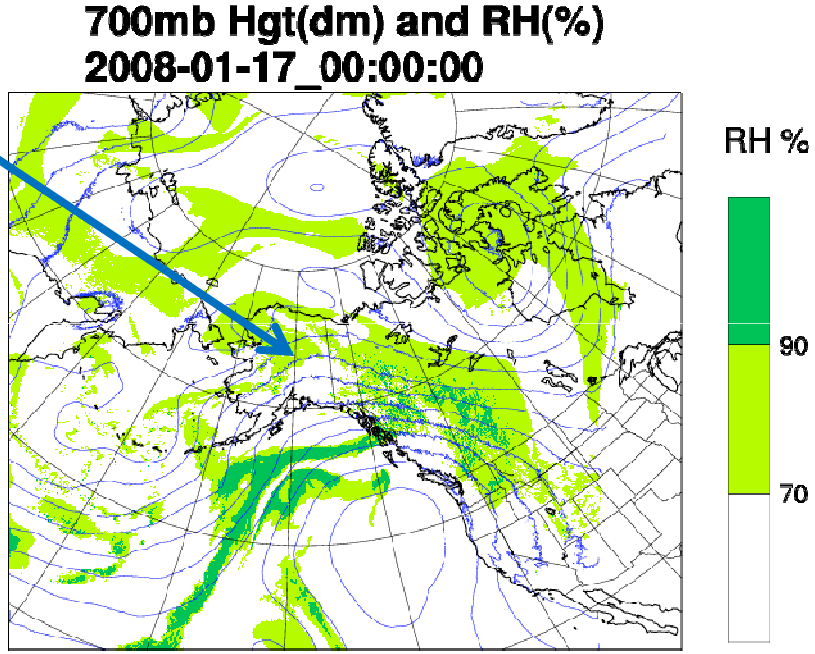
- ***Arctic Region Supercomputing Center***
- ***Peter Johnsen, Cray, Inc.***
- ***John Michalakes, WRF architect, NCAR***

The ARSC WRF Benchmarking Suite

- **Motivated by Michalakes' WRF V3 Parallel Benchmark Page**
- **Goals**
  - *Support testing of WRF on all architectures from single-CPU, to novel architectures, to the largest HPC systems*
  - *Provide the data and tools to make WRF benchmarking easy*

ARSC WRF Benchmark Domain

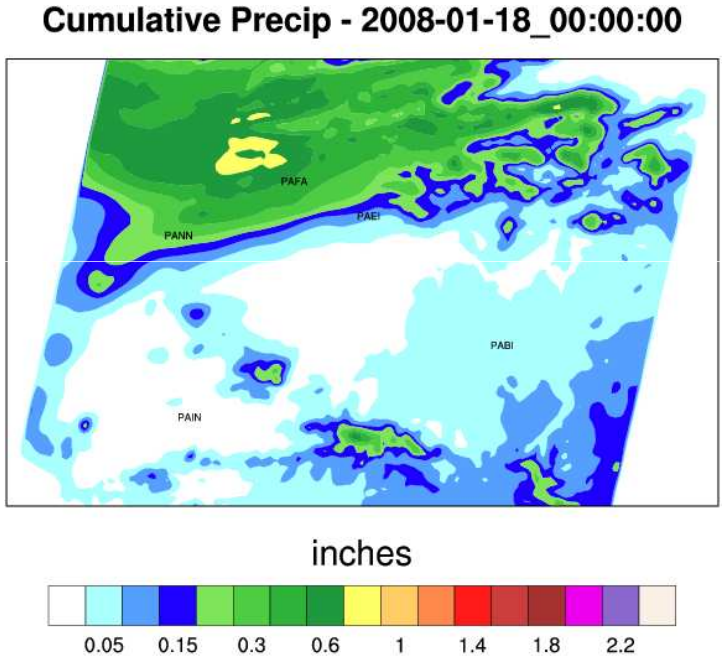
- ***Standard 6075x6075km region, centered on Frank Williams' office***
- ***Multiple resolutions to support full range of benchmarking needs***



Horiz Res	Grid points		
81km	75x75x28	=	157,500
27km	225x225x28	=	1,417,500
9km	675x675x28	=	12,757,500
3km	2025x2025x28	=	114,817,500
1km	6075x6075x28	=	1,033,357,500

ARSC WRF Benchmark Case

- ***Based on a highly localized FAI snowstorm, captured well by WRF***
- ***Benchmark case begins 48 hours into the forecast and lasts for 3 forecast hours***



Running a Benchmark Case

- ***Install WRF on target machine***
- ***Download following from benchmark site***
  - WRF Restart file (full variable dump of benchmark case at Forecast Hour 48)
  - WRF lateral boundary condition file
  - WRF namelist.input (run time parameters for the benchmark case)

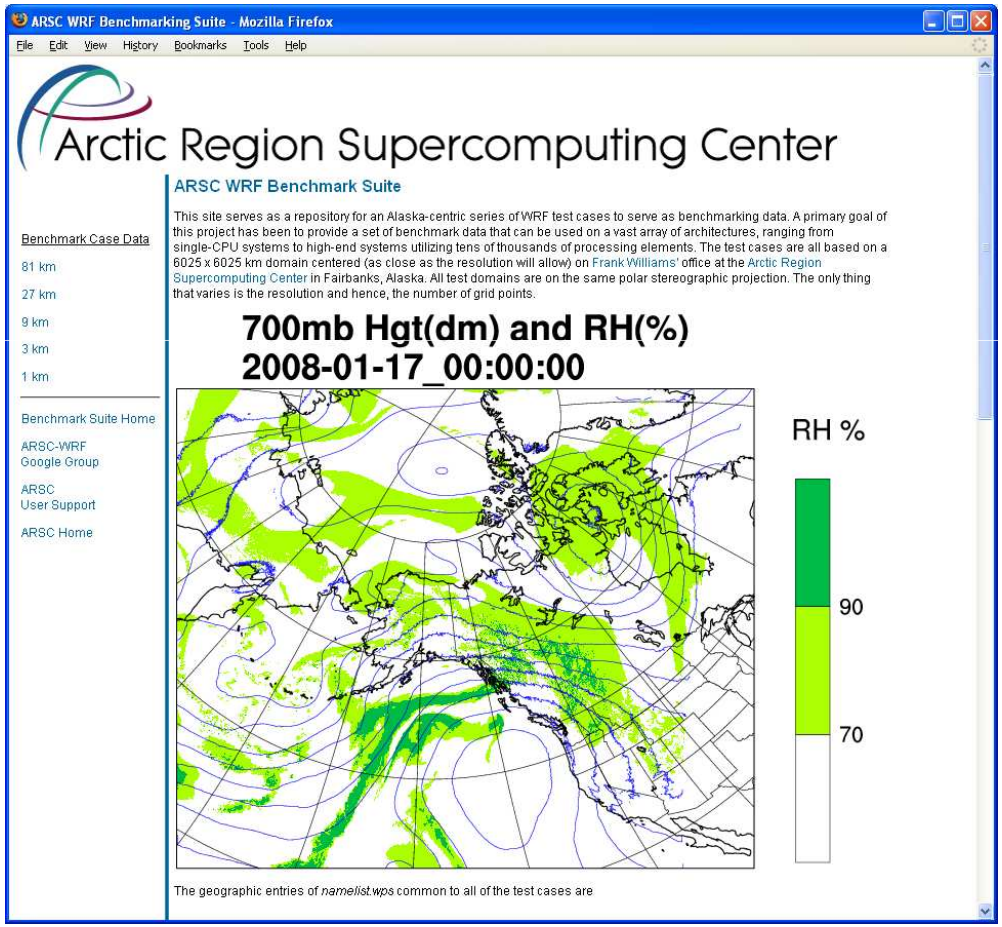
Running a Benchmark Case

- ***Use job-launching procedures on target machine to run the executable***
- ***After 3 forecast hours, the simulation produces***
  - A file with timing information
  - A WRF output file of the major output variables

ARSC WRF Benchmark Suite

http://weather.arsc.edu/BenchmarkSuite/

- **Currently available with test case resolutions of 81km, 27km, 9km and 3km**
- **The 1km case is still being worked on**

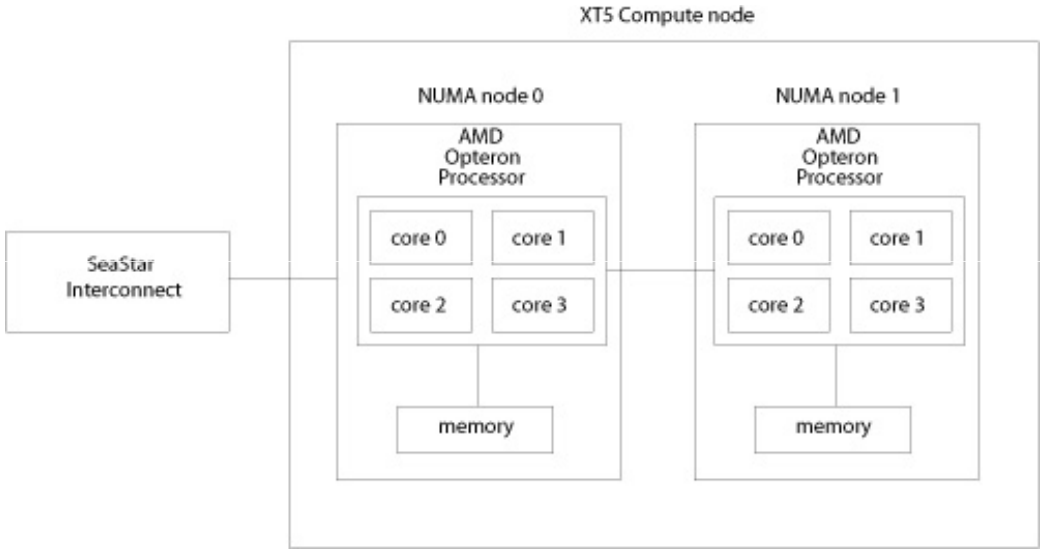


Preliminary Benchmarking on Cray XT5

- ***WRF V3.0.1.1, compiled with PGI and gcc for a Cray XT/CNL environment***
  - Distributed MPI executable
  - Hybrid MPI/OpenMP executable
  - Used default WRF configure options

The ARSC Cray XT5, pingo

- ***432 compute nodes, each with***
  - 32 GBytes of shared memory
  - 2 quad core 2.3 GHz AMD Opteron processors, connected through Cray Seastar 2+ interconnect interface
- ***Total of 3,456 cores***
- ***Supported by 150 TB Lustre scalable file system***



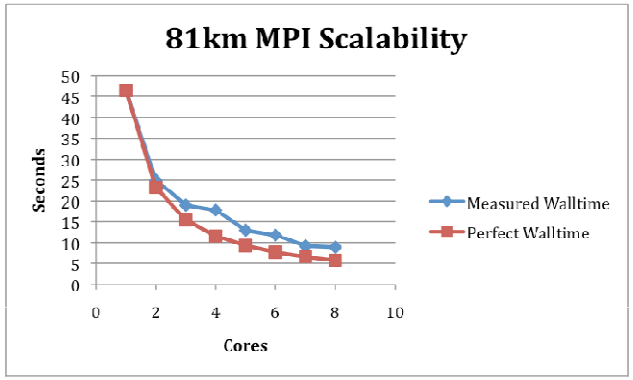
From *Louhi User's Guide*, ©CSC – IT Center for Science Ltd.

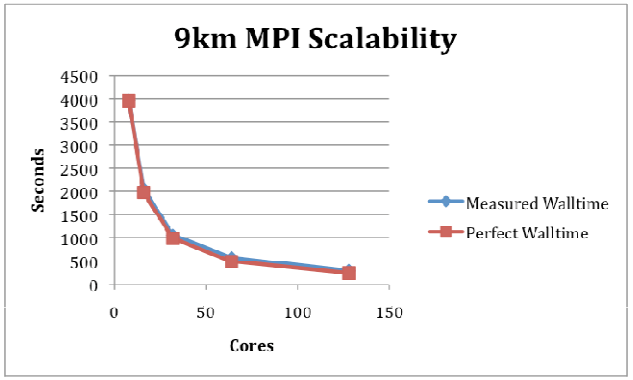
pingo

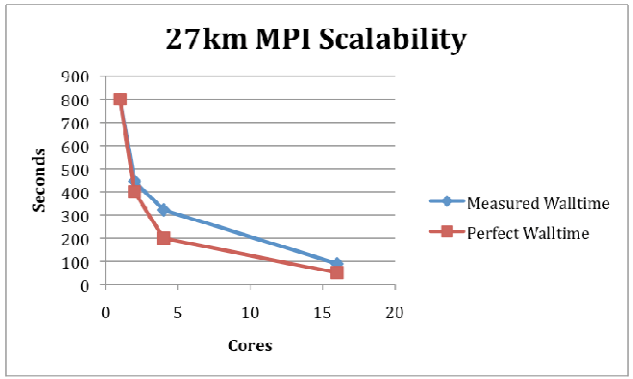
- ***A pingo is an earth-covered ice hill formed by the upward expansion of underground ice. Pingos tend to form in permafrost environments and can reach heights of up to 230 feet.***

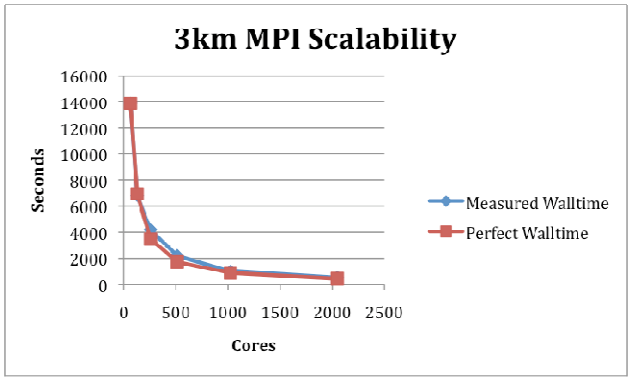
Pingo image courtesy of Emma Pike, Wikimedia Commons

Basic Scalability with MPI

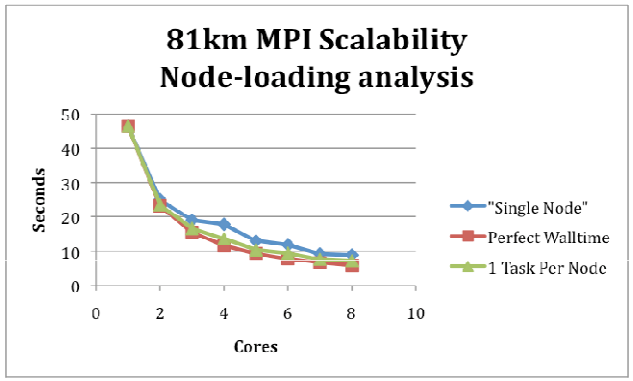


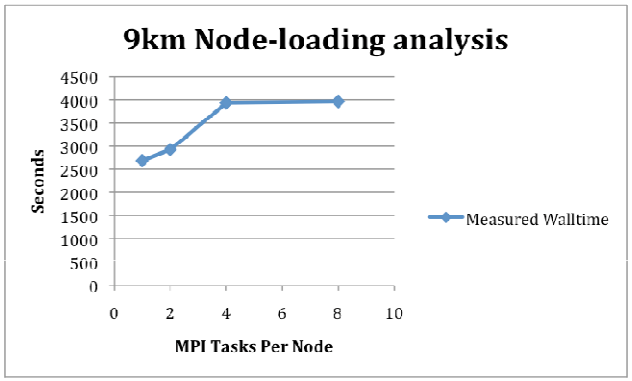


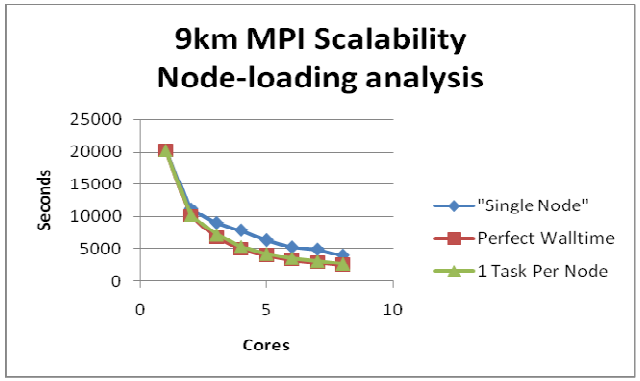




Node-loading Analysis

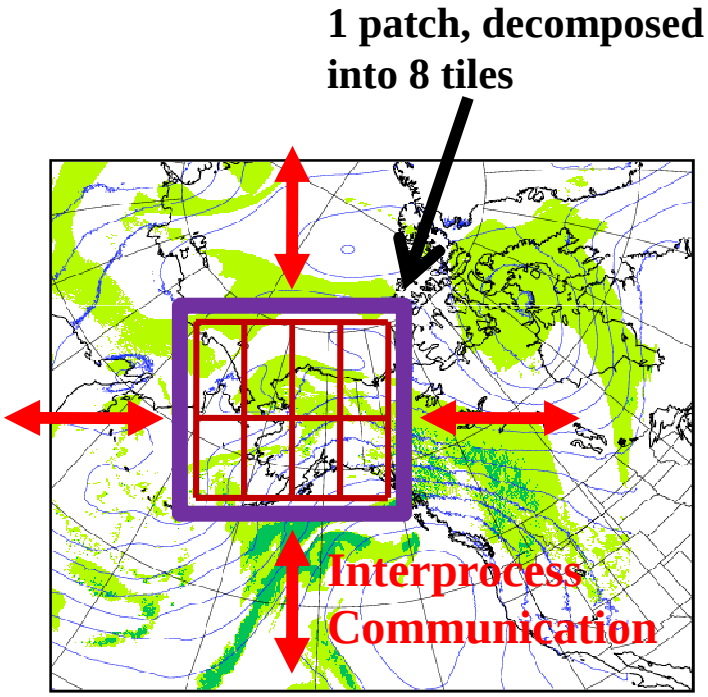






Hybrid WRF MPI/OpenMP

- ***Support for hybrid distributed and shared memory computations***
- ***Domain decomposed into <u>patches</u> assigned to MPI processes (message passing)***
- ***Patches further decomposed into <u>tiles</u> assigned to OpenMP threads (shared memory)***

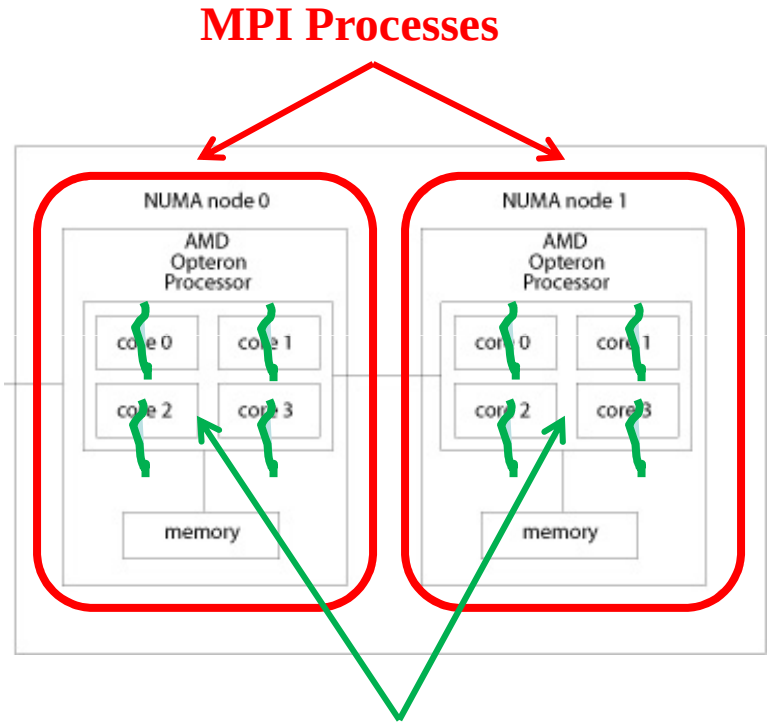


Running Hybrid WRF on the XT5

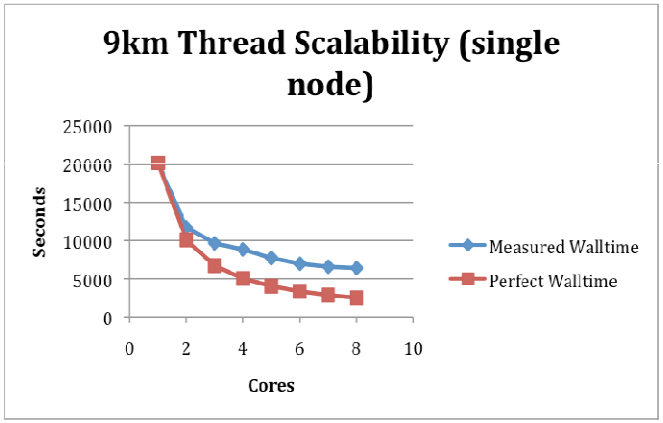
- ***With PBSPro, allocate MPI tasks and threads. For example – to run 8 MPI processes, two on each node, with 4 threads assigned to each MPI task:***

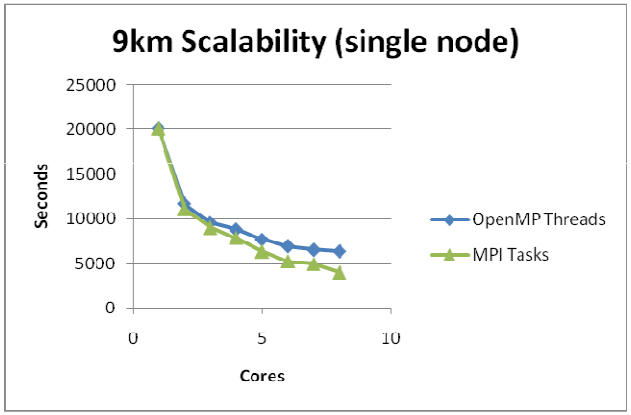
```
export OMP_NUM_THREADS=4
#PBS  -l  mppwidth=8
#PBS  -l  mppnppn=2
#PBS  -l  mppdepth=4

aprun  -n8  -N2  -d4  ./wrf-hybrid.exe
```

- ***This gives us four nodes, each with 2 MPI processes, each process running 4 threads. Total of 32 threads***

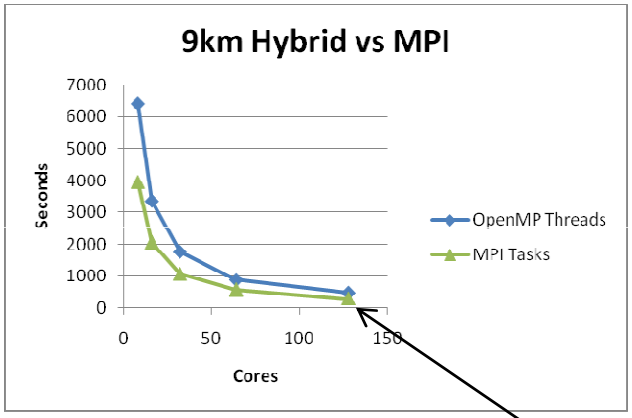


Thread Scalability on a Single Node





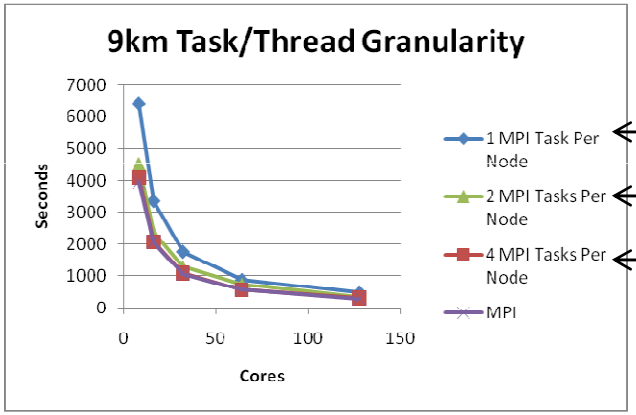
Hybrid vs MPI Performance



128 PEs

- **MPI – 294 seconds**
- **Hybrid – 490 seconds**

Task/Thread Decomposition Analysis



8 threads per MPI task

4 threads per MPI task

2 threads per MPI task

Setting Up a Large-Scale Problem

- Original intent was to try a 1km resolution case with 6075x6075x28 (over 1 billion) grid points
- We ran into a number of issues, described in the paper, and we're still working on it.
- We did manage to get a 250 million grid point case running

The Quarter-Billion Point Benchmark Case

- ***Same areal coverage as the other cases***
- ***2km resolution, using 3038x3038x28 grid points***
- ***We had problems generating a restart file, but were able to run a simulation from initial input data***
- ***For context***
  - 9km restart file – 4.2 GBytes
  - 3km restart file – 37.7 GBytes

2km Scalability

- ***Additional tests***
  - 1 MPI process assigned to each of 128 nodes, each task running 8 threads – 5,511 seconds
  - 8 MPI processes assigned to each of 128 nodes – 4,109 seconds
  - 1 MPI task assigned to each of 256 nodes, each running 4 threads (i.e. half the cores were left idle) – 3,910 seconds

Summary

Summary

- ***Primary motivation – implementation of versatile WRF benchmark suite for entire spectrum of architectures, useful to the HPCMP community and others wanting to test WRF on new systems***
- ***Preliminary benchmarking on pingo suggests that running in MPI-only mode yields significantly better (almost a factor of two) performance than hybrid MPI/OPenMP mode***
- ***We are enjoying the opportunity to push the limits with benchmark cases that seem to break things!***