CAM Performance on the X1E and XT3

Patrick H. Worley
Oak Ridge National Laboratory

The 48th Cray User Group Conference
May 8-11, 2006
Palazzo dei Congressi
Lugano, Switzerland

Apology

- I apologize for being unable to attend the meeting. The presenter, Trey White, has been a collaborator in previous work on this topic, and is well-prepared to present and explain the results herein.

Acknowledgements

- Research sponsored by the Atmospheric and Climate Research Division and the Office of Mathematical, Information, and Computational Sciences, Office of Science, U.S. Department of Energy under Contract No. DE-AC05-00OR22725 with UT-Battelle, LLC.
- These slides have been authored by a contractor of the U.S. Government under contract No. DE-AC05-00OR22725 with UT-Battelle, LLC. Accordingly, the U.S. Government retains a nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or allow others to do so, for U.S. Government purposes.
- This research used resources of the National Center for Computational Sciences at Oak Ridge National Laboratory, which is supported by the Office of Science of the Department of Energy under Contract DE-AC05-00OR22725, and of the National Energy Research Scientific Computing Center, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC03-76SF00098.

Acknowledgements

- Many of the results described in this talk were obtained in collaboration with
  - Art Mirin at Lawrence Livermore National Laboratory (LLNL)

  and
  - Brian Eaton and Mariana Vertenstein, both in the Community Climate System Model (CCSM) Software Engineering Group (CSEG) at the National Center for Atmospheric Research (NCAR).
- This work was funded by the Scientific Discovery through Advanced Computing (SciDAC) projects
  - Collaborative Design and Development of the Community Climate System Model for Terascale Computers

  and
  - High-End Computer System Performance: Science and Engineering.

Overview

- The Community Atmosphere Model (CAM) was previously ported and optimized on the X1 and reported on at previous CUG meetings.
- This talk is an update, focusing on new performance results for the
  - X1E,
  - XT3, and
  - a new dynamical core that will be used in production in the future.

Outline

- CAM Overview
- X1E porting and optimization experience
- XT3 porting and optimization experience
- Performance results and analysis

Community Atmosphere Model (CAM)

Atmospheric global circulation model

- Timestepping code with two primary phases per timestep
  - *Dynamics*: advances evolution equations for atmospheric flow
  - *Physics*: approximates subgrid phenomena, such as precipitation, clouds, radiation, turbulent mixing, …
- Multiple options for dynamics:
  - Spectral Eulerian (EUL) dynamical core (*dycore*)
  - Spectral semi-Lagrangian (SLD) dycore
  - Finite-Volume semi-Lagrangian (FV) dycore

  all using tensor product *latitude* x *longitude* x *vertical level* grid over the sphere, but not same grid, same placement of variables on grid, or same domain decomposition in parallel implementation
- Separate data structures for dynamics and physics and explicit data movement between them each timestep (in a "coupler")
- Developed at NCAR, with contributions from the Department of Energy (DOE) and the National Aeronautics and Space Administration (NASA).

CAM Performance Portability Goals

1) Maximize single processor performance, e.g.
   a) Optimize memory access patterns
   b) Maximize vectorization or other fine-grain parallelism
2) Minimize parallel overhead, e.g.
   a) Minimize communication costs
   b) Minimize load imbalance
   c) Minimize redundant computation

for

- a range of target systems,
- a range of problem specifications (grid size, physical processes, …)
- a range of processor counts

while preserving maintainability and extensibility.

No optimal solution for all desired (platform,problem,processor count) specifications. Approach: compile-time and runtime optimization options.

CAM Performance Optimization Options

1. Physics data structures
   - Index range, dimension declaration
2. Physics load balance
   - Variety of load balancing options, with different communication overheads
   - SMP-aware load balancing options
3. Communication options
   - MPI protocols (two-sided and one-sided)
   - Co-Array Fortran
   - SHMEM protocols

   and choice of pt-2-pt implementations or collective communication operators
4. OpenMP parallelism
   - Instead of some MPI parallelism
   - In addition to MPI parallelism
5. Aspect ratio of dynamics 2D domain decomposition (FV-only)
   - 1D is latitude-decomposed only
   - 2D is latitude/longitude-decomposed in one part of dynamics, latitude/vertical-decomposed in another part, with remaps to/from the two decompositions during each timestep.

CAM Performance Experiments

1. Spectral Eulerian dycore running on T85L26 computational grid
   - 128x256x26 (latitude by longitude by vertical) grid
   - Current production dynamical core and grid resolution in CCSM
2. Finite Volume dycore running on 1.9x2.5 degree horizontal grid with 26 vertical levels
   - 96x144x26 (latitude by longitude by vertical) grid
   - Finite volume dycore is the preferred (required among current options) dycore for atmospheric chemistry due to its conservation properties. 1.9x2.5 degree resolution is the initial CCSM production grid size.
3. Finite Volume dycore running on 0.5x0.625 degree horizontal grid ('D grid') with 26 vertical levels
   - 361x576x26 (latitude by longitude by vertical) grid
   - 15 times larger than FV production grid resolution.

Performance result for a given dycore, problem size, platform, and processor count is the optimal observed over all compile and runtime optimization options.

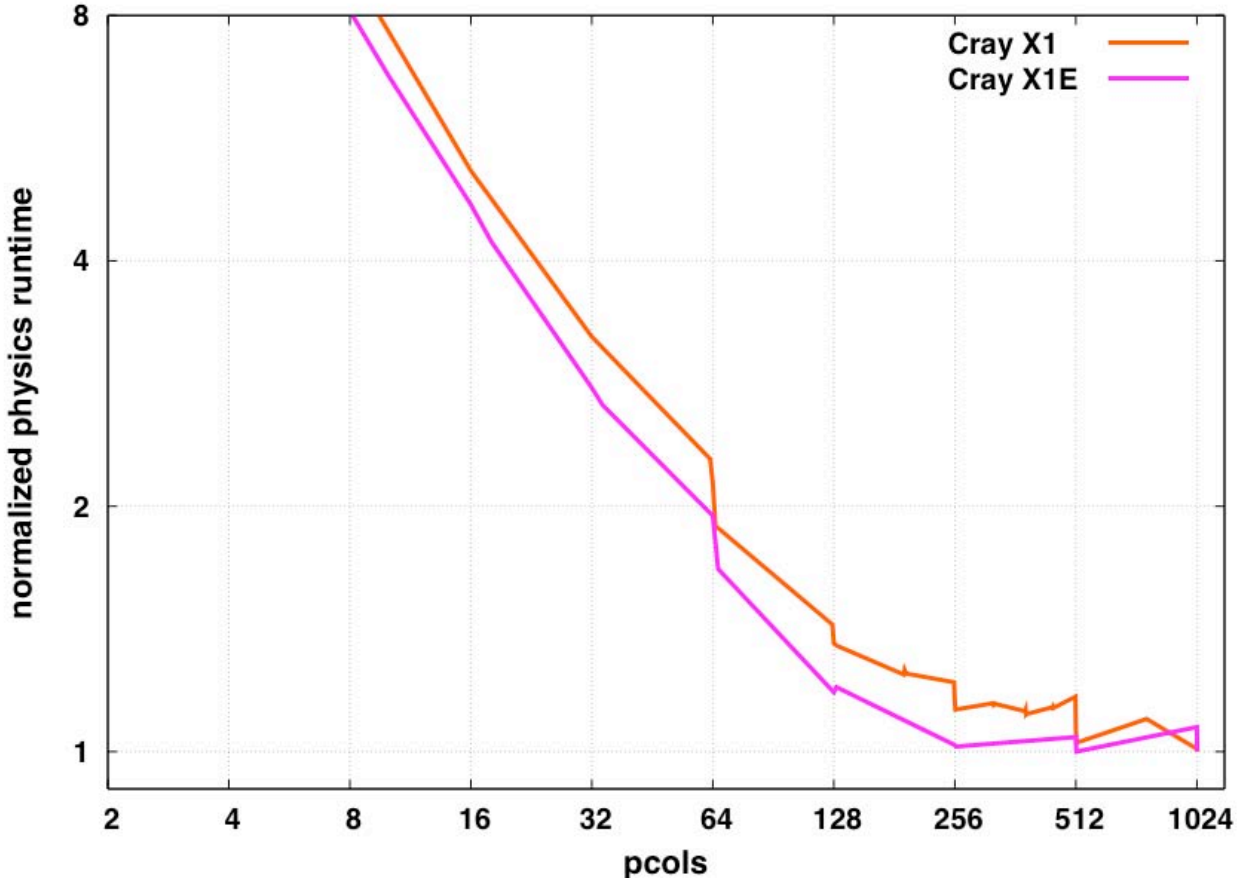
Experimental Platforms

- **Cray X1** at Oak Ridge National Laboratory (ORNL): 128 4-way vector SMP nodes. Each processor has 8 64-bit floating point vector units running at 800 MHz. Nodes are fully connected within 4-node subsets, and are connected via 2-D torus between subsets.
- **Cray X1E** at ORNL: 256 4-way vector SMP nodes. Each processor has 8 64-bit floating point vector units running at 1.13 GHz. Nodes are fully connected within 8-node subsets, and are connected via 2-D torus between subsets.
- **Cray XT3** at ORNL: 5294 single processor nodes (2.4 GHz AMD Opteron) and a 3-D torus interconnect.
- **Earth Simulator**: 640 8-way vector SMP nodes and a 640x640 single-stage crossbar interconnect. Each processor has 8 64-bit floating point vector units running at 500 MHz.
- **IBM p575 cluster** at the National Energy Research Scientific Computing Center (NERSC): 122 8-way p575 SMP nodes (1.9 GHz POWER5) and an HPS interconnect with 1 two-link network adapter per node.
- **IBM p690 cluster** at ORNL: 27 32-way p690 SMP nodes (1.3 GHz POWER4) and an HPS interconnect with 2 two-link network adapters per node.
- **IBM SP** at NERSC: 184 Nighthawk II 16-way SMP nodes (375MHz POWER3-II) and an SP Switch2 with two network adapters per node.
- **Itanium2 cluster** at Lawrence Livermore National Laboratory (LLNL): 1024 4-way Tiger4 nodes (1.4 GHz Intel Itanium 2) and a Quadrics QsNetII Elan4 interconnect.
- **SGI Altix 3700** at ORNL: 128 2-way SMP nodes and a NUMAflex fat-tree interconnect supporting cccNUMA global shared memory. Each processor is a 1.5 GHz Itanium 2 with a 6 MB L3 cache.
- **SGI Altix 3700 Bx2** at NASA: 1024 2-way SMP nodes and a NUMAflex fat-tree interconnect supporting NUMA global shared memory. Each processor is a 1.6 GHz Itanium 2 with a 9 MB L3 cache.

X1E Experiences

1. Modifications required?
   - No. X1 port worked fine.
2. Optimal settings changed compared to X1?
   - No:
     - Long inner loops (implying worse cache locality) are better, to a point.
     - Use load balancing.
     - Use MPI collectives.
     - Do not use MPI derived types when communicating.
     - Do not use OpenMP (because it competes with CSDs).
   - but sensitivities are somewhat different than on the X1.
3. Performance Comparisons?
   - X1E faster than X1! :-)
   - Constant battle as code evolves

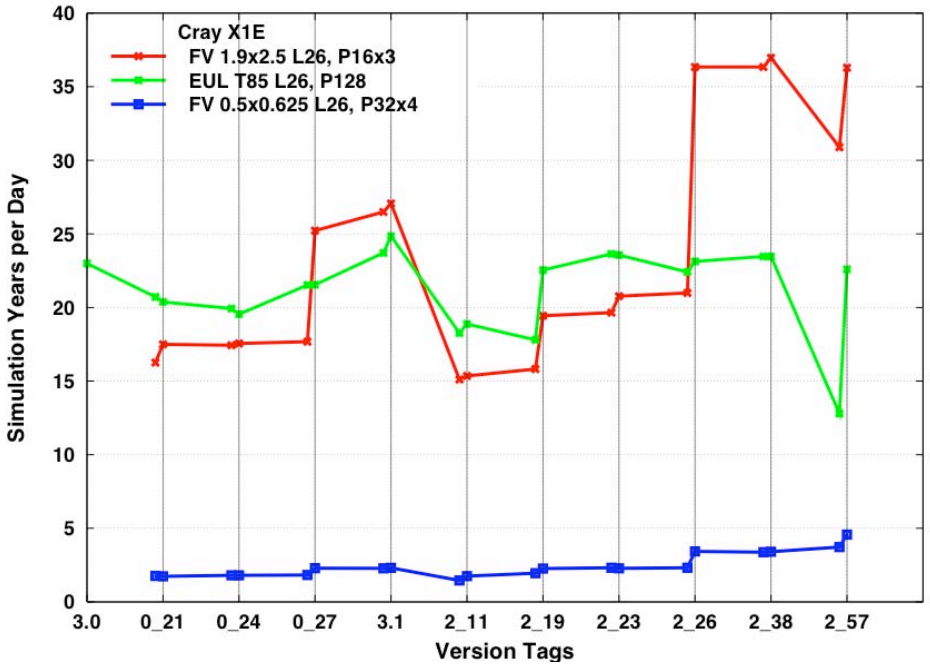
Tuning Physics Data Structures: X1 vs X1E

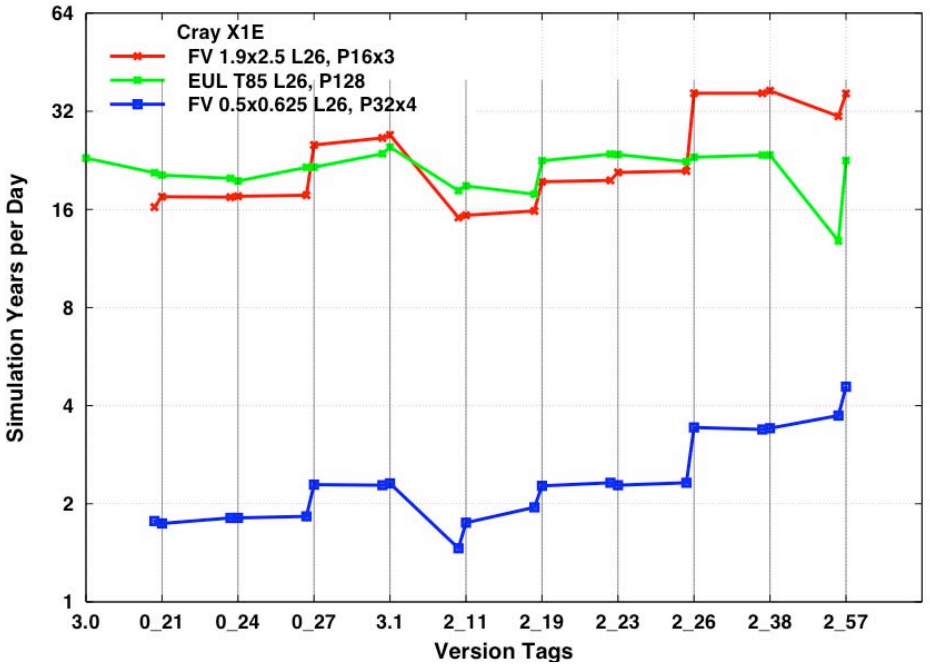


pcols parameter determines vector length and cache locality in physics.

- X1E is slightly less sensitive to pcols parameter than X1.
- pcols = 514 is a good choice for optimum for both systems. 258 is a reasonable choice for the X1E, but is not as good on the X1.

Recent CAM Performance History: X1E





- Performance impact of Mirin/Worley check-ins on the Cray X1E.
- Not all check-ins improved performance, nor were expected to - some improved portability, added new performance tuning options, or fixed bugs.
- Maintaining performance as CAM evolves is (and will be) as important as further improving current performance.

XT3 Experiences

1. Modifications required?
   - Yes: Makefile, system calls, I/O, bug workarounds
2. Optimal settings?
   - pcols = 40
   - Use load balancing.
   - Use MPI collectives.
   - Do not use derived types in point-to-point MPI communication.
3. Performance?
   - I/O performance is important to optimize.

Porting to XT3

1. Modified Makefile
   - Recognize `ftn` and `cc` as being PGI compilers
   - -O1 optimization for a few routines that cause runtime errors when compiled with -fast
2. Eliminated system calls not supported by Catamount
   - Disabled collection of memory statistics (eliminated read of `/proc/<pid>/statm`).
   - Replaced `gettimeofday` with `MPI_Wtime` in performance timers.
3. Optimized I/O performance
   - Increased buffering for output to stdout and stderr (using `setvbuf`).
   - *All* data input files are read from the Lustre file system.
   - *All* output files are written to the Lustre file system.
4. Workaround for Gather runtime error: `MPIDI_PortalsU_Request_FDU_or_AEP: dropped event on unexpected receive queue`
   - Replaced MPI_Gatherv used in model output with point-to-point implementation where simultaneous sends to root are not allowed.

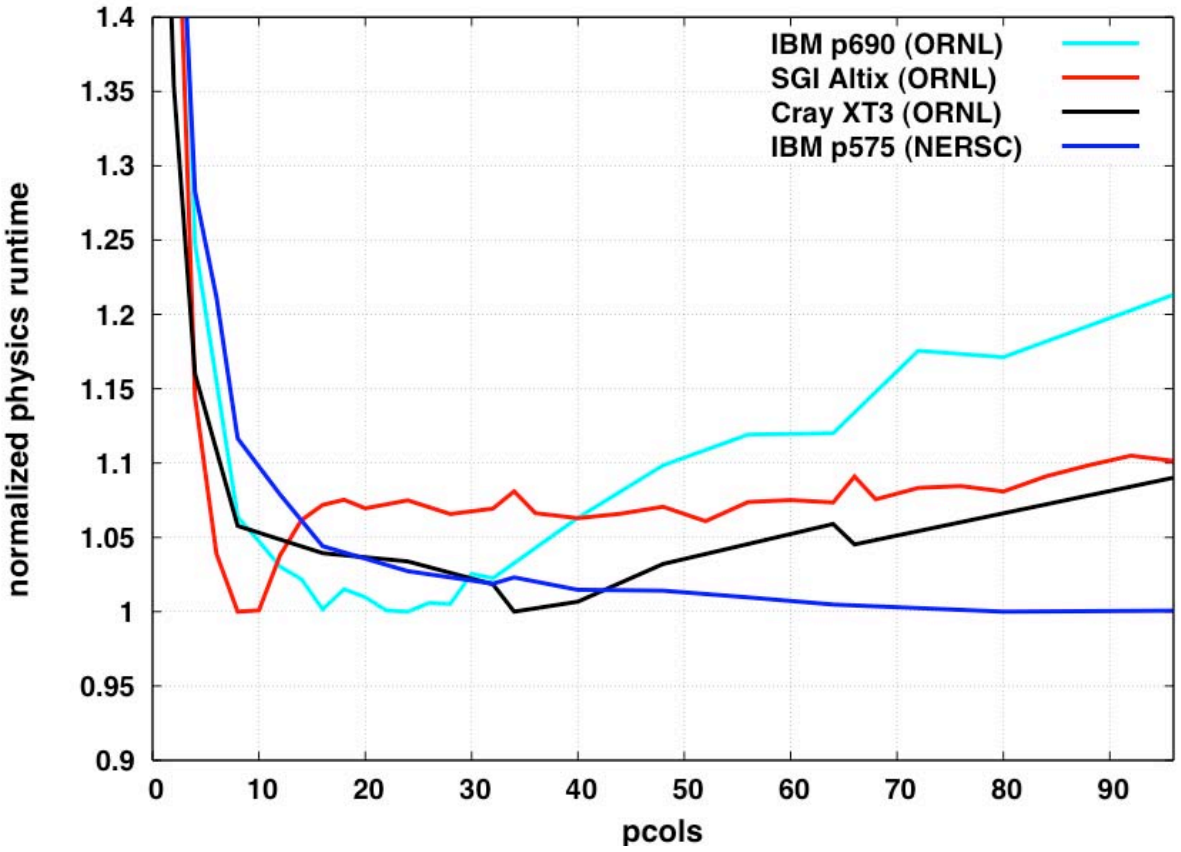
Tuning Physics Data Structures

pcols parameter determines vector length and cache locality in physics.

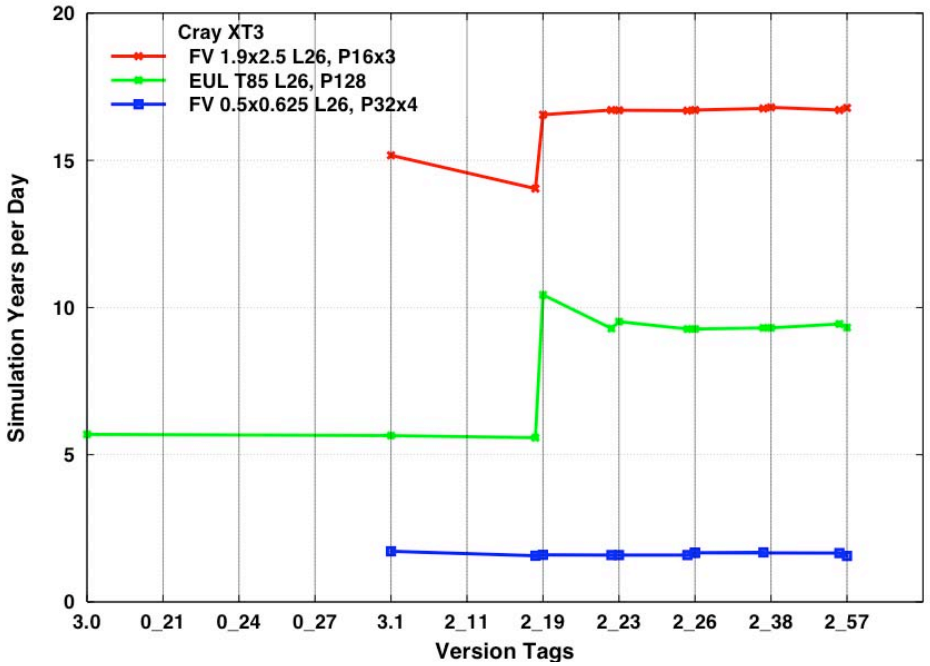
	optimum
X1E :	514
p575 :	80
XT3:	34
p690:	24
Altix:	8

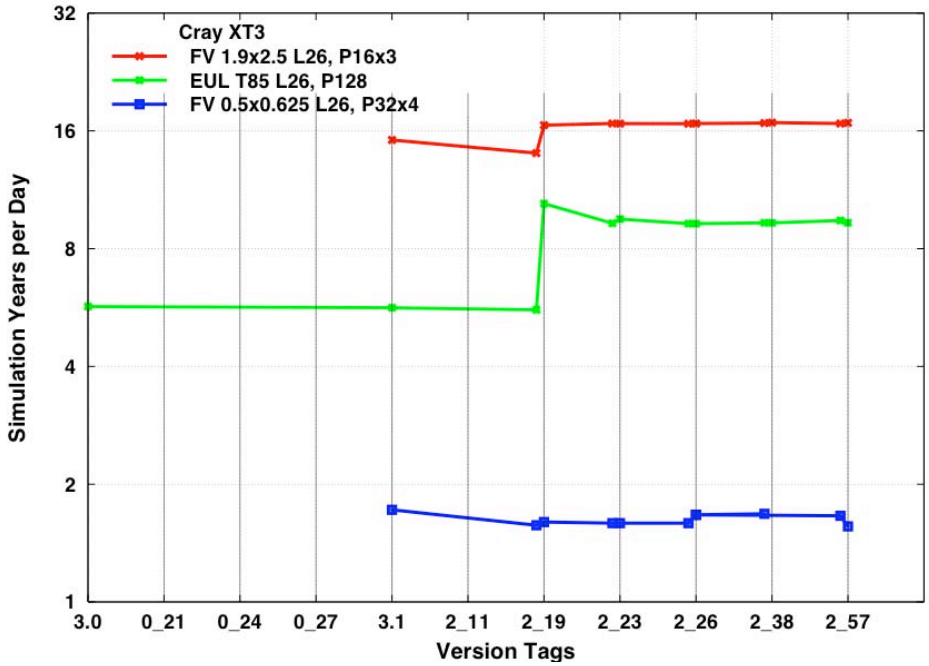
<= 4 bad for all systems.

For XT3, performance reasonable for pcols between 10 and 60.



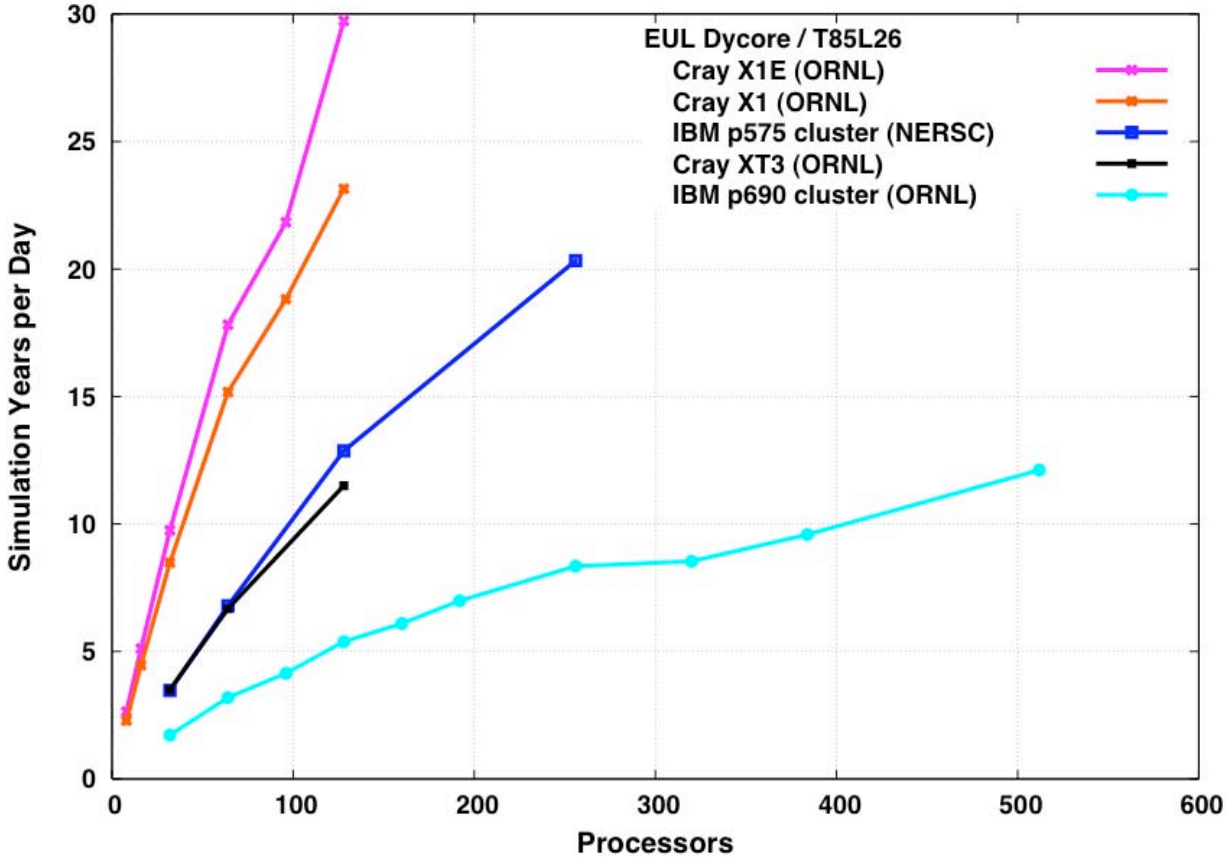
Recent CAM Performance History: XT3





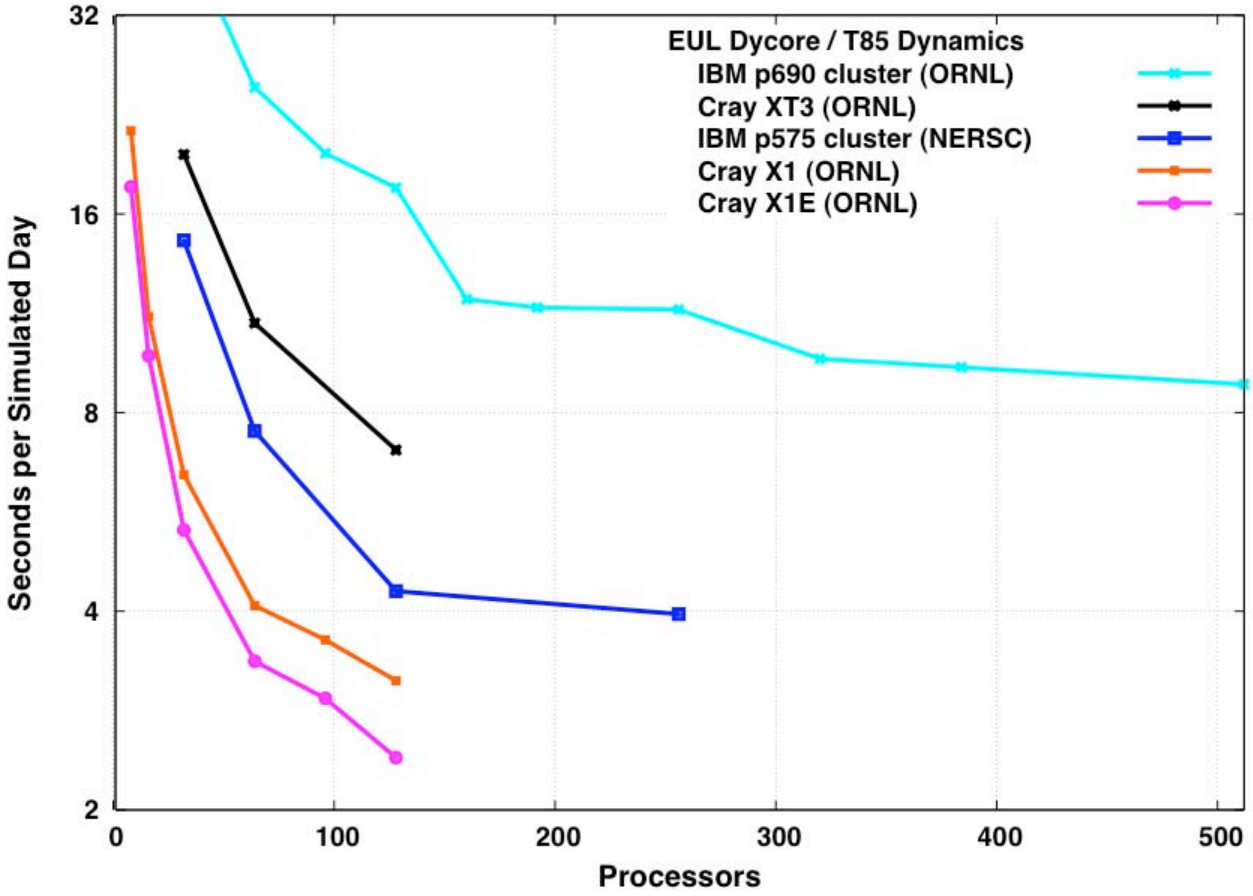
- Performance impact of each Mirin/Worley check-in on the Cray XT3.
- Beyond initial optimization (cam3_2_19), little attention has been paid to XT3 performance so far. Performance improvement on graphs due entirely to changes in buffer size for `stdout` and `stderr`. (Lustre file system used exclusively for all runs.)

Performance: T85



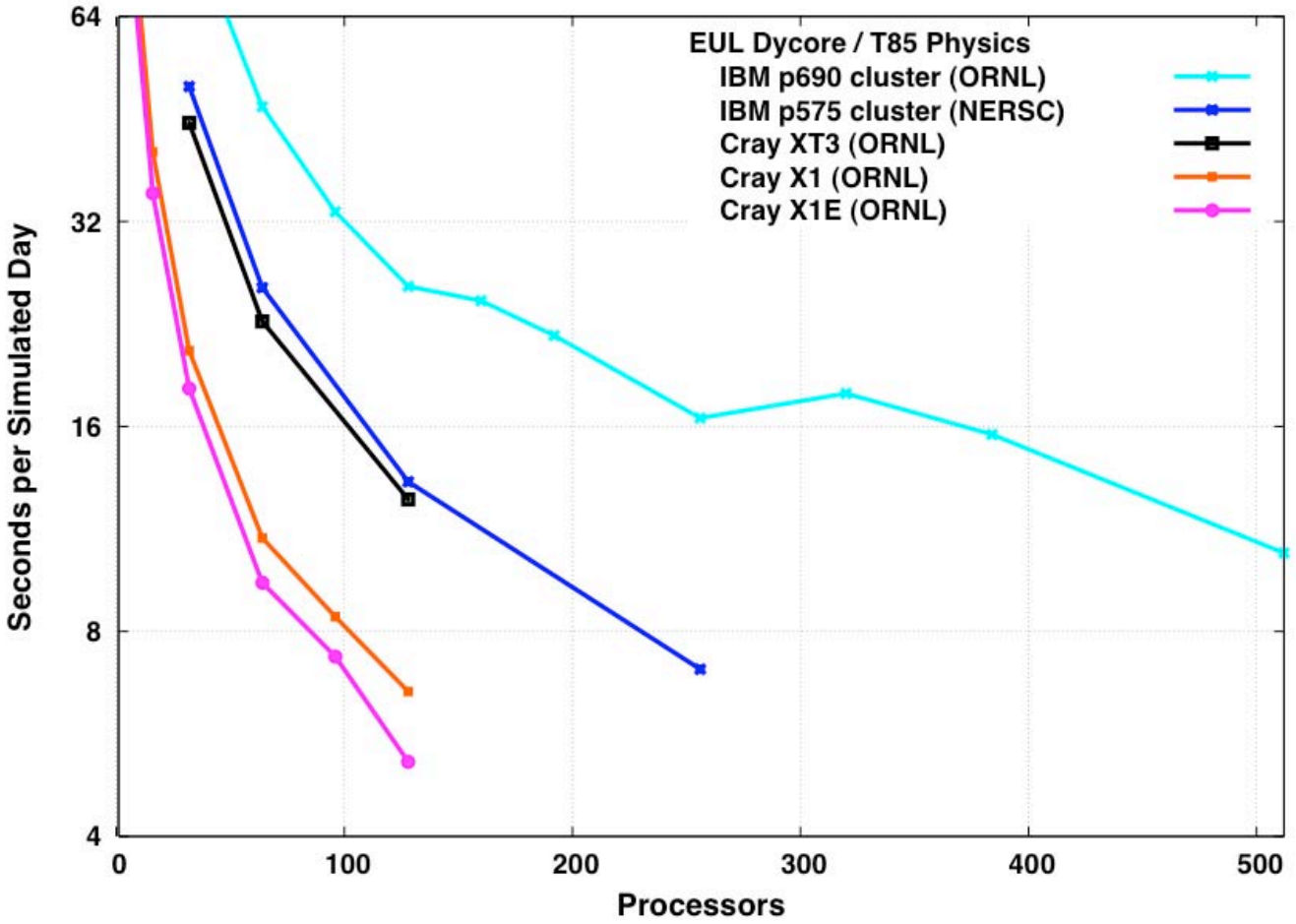
- EUL dycore limits scalability for this problem size to 128 MPI processes. IBM systems use OpenMP parallelism to exploit additional processors.
- X1E is 28% faster than X1 for this benchmark, and 2.5 times faster than the XT3.
- XT3 has same performance as p575 for 32 and 64 processors. OpenMP gives p575 an advantage for 128 processors (and ability to use even more processors).

T85 Performance Diagnosis: Dynamics



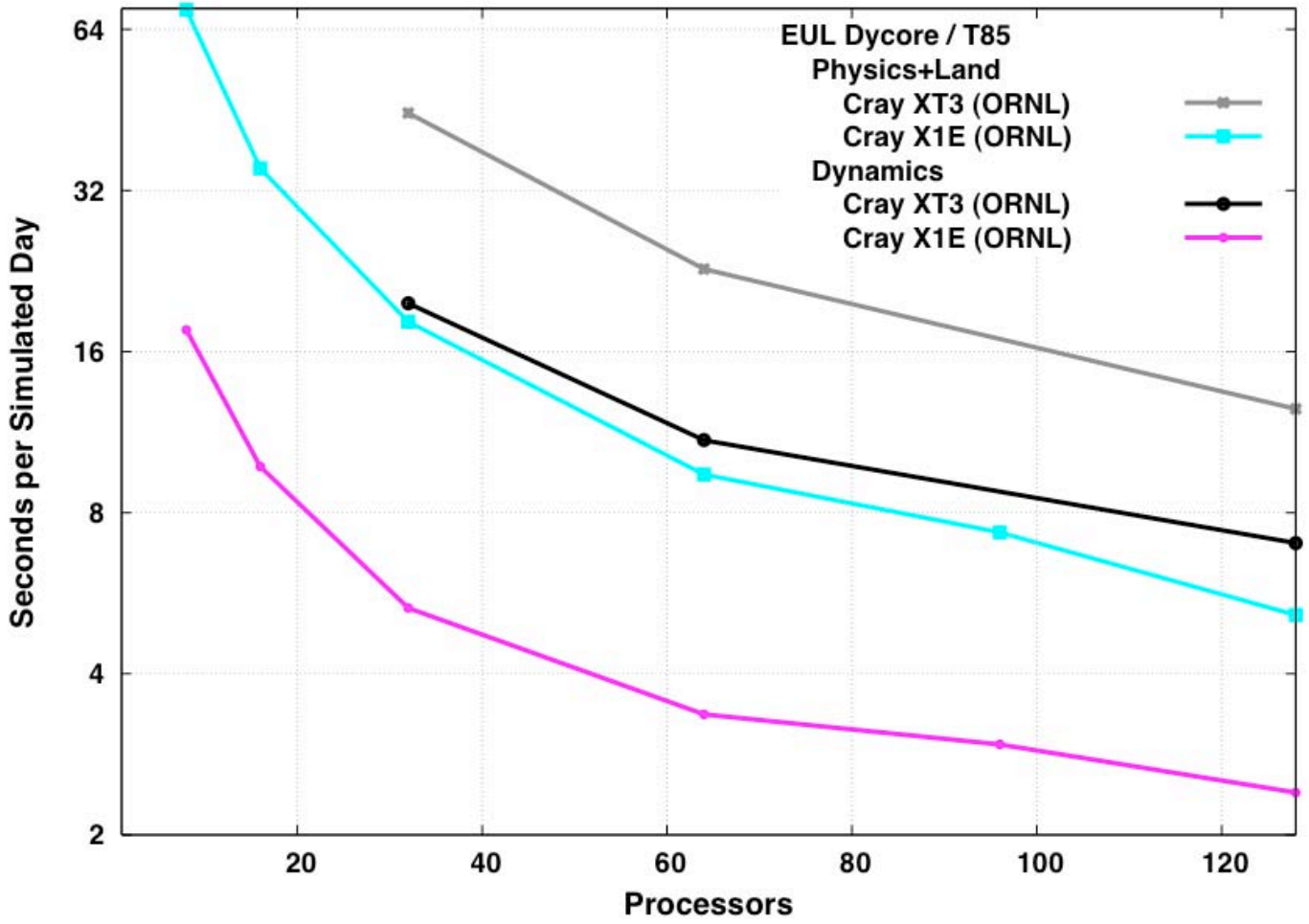
- Dynamics scales well to 128 processors on all systems. OpenMP parallelism does not improve dynamics performance significantly when using more than 160 processors.
- Dynamics faster on p575 than on XT3 for all processor counts.

T85 Performance Diagnosis: Physics



- Physics (also) scales well to 128 processors on all systems. OpenMP parallelism allows more processors to be used to some effect.
- Physics faster on XT3 than on p575 for same processor counts.

T85 Performance Diagnosis: X1E vs. XT3

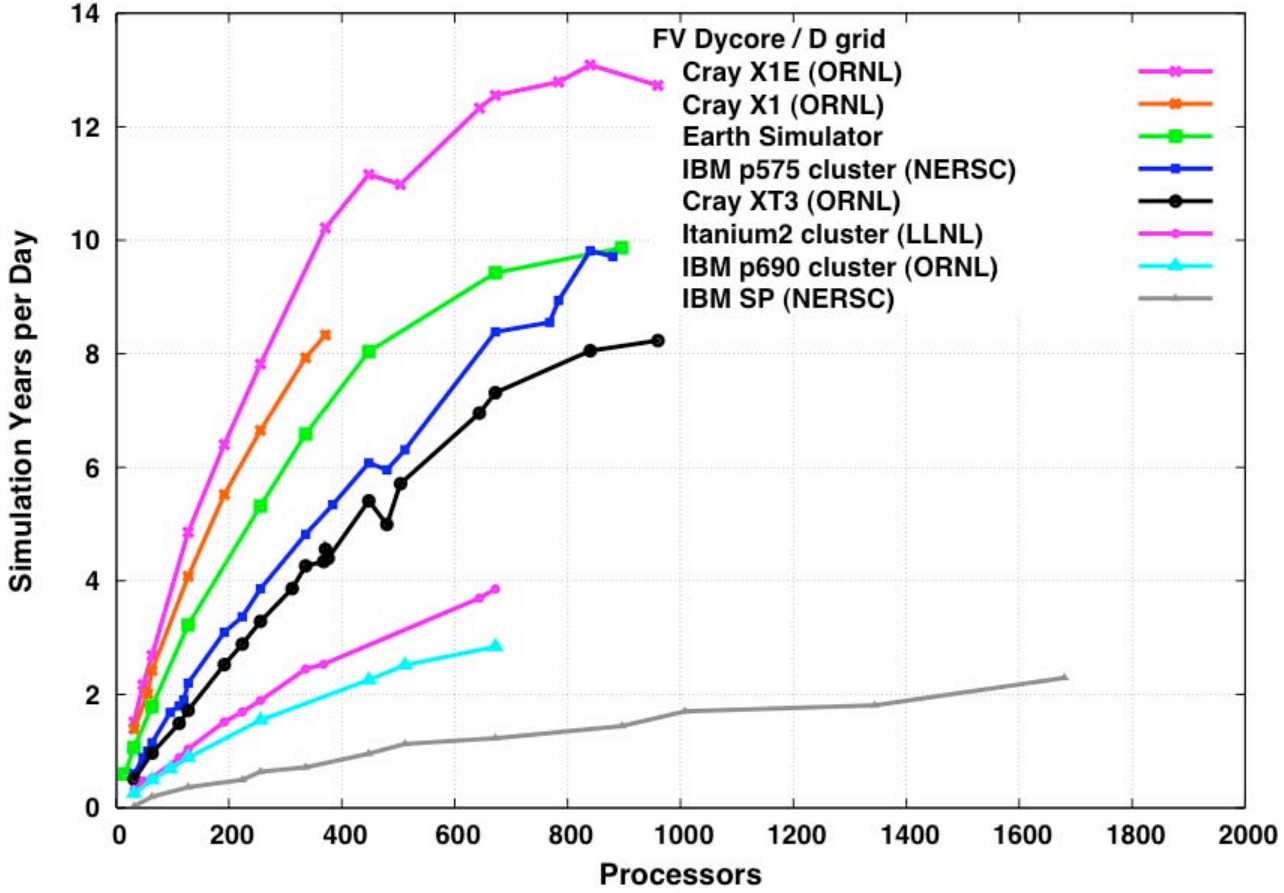


- Physics 2-4 times as costly as dynamics.
- Physics and Dynamics scaling similar on both systems up to 128 processors.

T85 Performance Comparisons

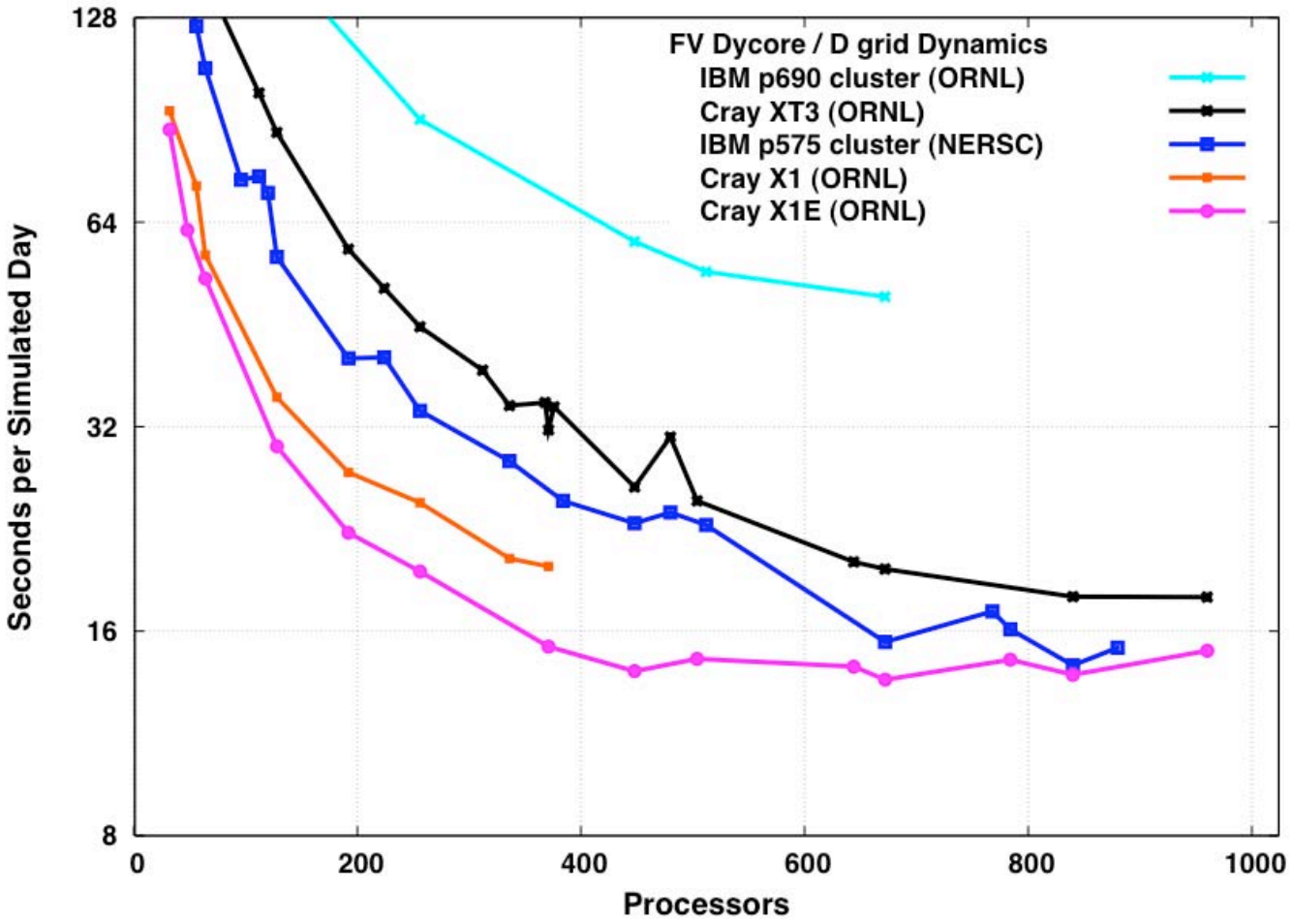
1. The X1E is 16% faster than the X1 for 8 through 96 processors (21% faster for dynamics; 15% faster for physics). For 128 processors the X1E advantage jumps to 28% (31% for dynamics and 27% for physics). The reason for the increase in the X1E advantage is not clear from the available performance data.
2. For the physics, the XT3 is 13% faster than the p575 cluster for 32 and 64 processors, and 6% faster for 128 processors. The change is due to the increasing percentage of time spent in the land model and in a global sum, both of which run faster on the p575 cluster than on the XT3.
3. For the dynamics, the p575 cluster is faster than the XT3 by 35% for 32 processors, increasing to 64% faster for 128 processors. The p575 advantage appears to be due both to higher peak processor speed for fmadd rich computations and to lower MPI communication overhead (from exploitation of OpenMP parallelism).
4. The ability to use more processors in the physics than in the dynamics in a pure MPI implementation would improve scalability on the Cray systems.

Performance: FV / D Grid



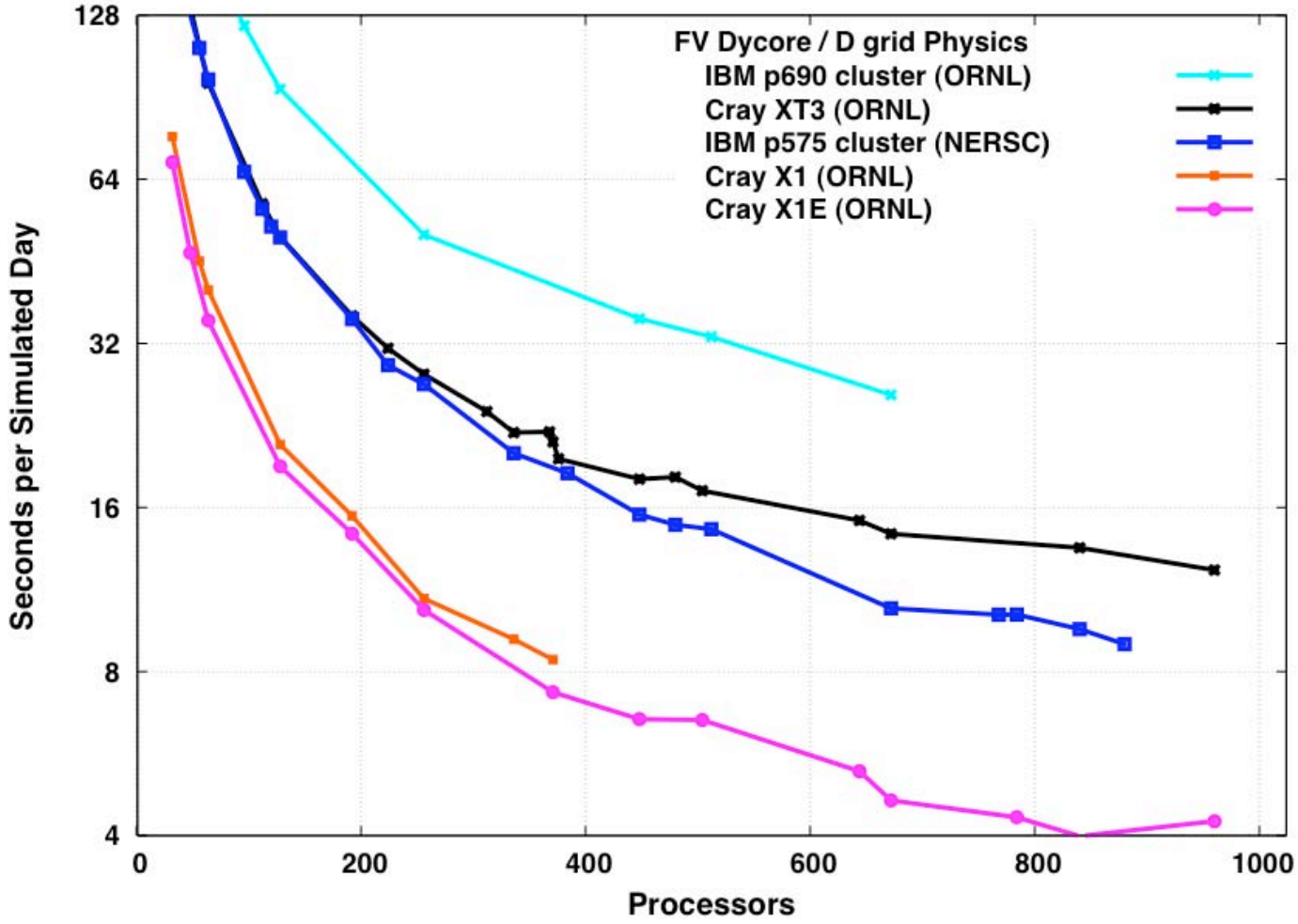
- Earth Simulator results courtesy of D. Parks. SP results courtesy of M. Wehner. Maximum number of MPI processes is 960. IBM systems and Earth Simulator use OpenMP to increase scalability.
- Recent performance optimizations backported into CAM 3.1 for these experiments.

D Grid Performance Diagnosis: Dynamics



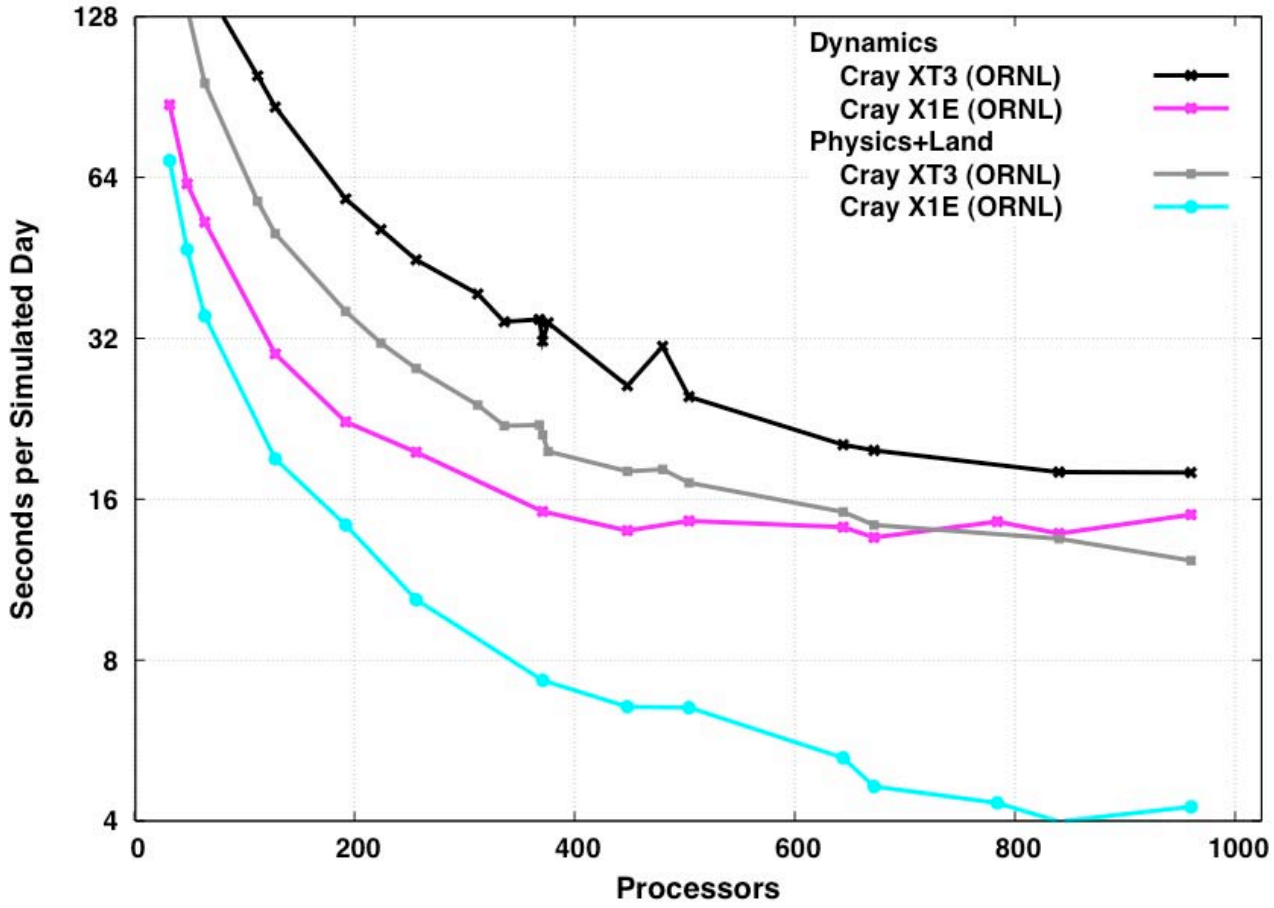
- Dynamics performance relatively flat for more than 700 processors.
- On X1E, dynamics best performance at 448 processors.
- X1 runs may not have all FV performance optimizations used in X1E runs.

D Grid Performance Diagnosis: Physics



- Vector systems show most advantage in physics.
- XT3 performance similar to that of p575 for small processor counts.

D Grid Performance Diagnosis: X1E vs. XT3



- Dynamics and physics scale similarly on XT3.
- On X1E, physics scales much better than dynamics.

D Grid Performance Comparisons

1. The X1E is 9% faster than the X1 for 8 processors, increasing to 23% faster for 371 processors. The improvement is due entirely to the performance advantage in the dynamics, which grows from 7% for 32 processors to 31% for 371 processors. (The advantage in the physics varies between 5% and 15%.) The increase in the dynamics advantage is due primarily to the increasing percentage of time spent in a section of code that does not scale well (due to load imbalance) in which the X1E performance advantage is over 35%.
2. For the physics, the XT3 is 5% faster than the p575 cluster for 32 processors, but the advantage disappears by 128 processors, and the p575 cluster is faster by 41% for 840 processors. The change is due to the increasing percentage of time spent in a global sum and an associated write statement. This time grows with processor count, and takes approximately 5 times longer on the XT3 than the p575 cluster for all processor counts.

D Grid Performance Comparisons

3. OpenMP allows the p575 cluster to use different domain decompositions than the XT3 for the same processor count, which makes it difficult to compare the dynamics performance on the two systems. The p575 performance advantage varies between 10% and 30%, and is greatest when the p575 cluster can use a 1D domain decomposition while the XT3 must use a 2D domain decomposition. The p575 performance advantage is not a simple function of processor count but, in general, does not grow with processor count. For large processor counts short simulations do not show any advantage to the p575, while longer simulations do. The source of this "performance degradation" on the XT3 is under investigation, but may be due to diagnostic writes to standard out.

Current Limits to FV Scalability

1. Polar filter introduces load imbalances, especially on vector systems (because the small number of short FFTs do not vectorize well).
2. Requirement that at least 3 latitudes and 3 vertical levels be present in each "block" of domain decomposition within FV dycore. For D grid with 26 vertical levels, limit is a 120x8 processor grid, or 960 MPI processes. For 1.9x2.5 degree grid, limit is a 32x8 processor grid, or 256 MPI processes.
3. Physics can use many more processors, but currently limited to the same number of MPI processes as the dynamical core. OpenMP can be used to assign more processors to physics than to dynamics, mitigating this to some degree. There is also some OpenMP parallelism available within the dynamics.
4. On vector systems, additional parallelism in physics is of limited utility, as vector length drop below 220 for D grid when using more than 960 processors (and drops below 110 in radiation routines).

New and Planned Activities

1. Dynamics Scaling
   - Generalize dynamics/physics interface to support dycores not using lon/lat grid. Investigate new, more scalable dycores, for example, FV on a cubed sphere computational grid.
2. Physics Scaling
   - Add support to use different numbers of MPI processes in the dynamics and in the physics (generalizing current OpenMP approach to pure MPI codes).
3. Increasing model resolution exacerbates the current I/O bottlenecks and memory impact of the (few) remaining global arrays.
   - Investigate parallel I/O on target platforms.

Summary: X1E

1. X1E achieves best performance on both T85 and D grid benchmarks, demonstrating good performance improvement over X1. Maintaining good performance as model evolves is an important continuing activity.
2. Scaling is limited by domain decomposition limits in dycores, by load imbalances in dynamics, and by short vector lengths for larger processor counts.
3. Proposed model modifications will address load imbalances and some of the FV domain decomposition limits (by not requiring vertical decompositions) but not short vector lengths.
4. New physics will add significant new computational and communication demands. X1E should continue to perform well under these new demands once new code is vectorized.

Summary: XT3

1. XT3 demonstrated good scalability and reasonable performance relative to similar architectures.
2. I/O was the primary focus of performance optimizations up until now, and will continue to be so in the immediate future.
3. Scaling primarily limited by domain decomposition limits in dycores, and OpenMP is not available to extend limits.
4. Proposed model modifications will increase FV domain decomposition scalability limits. They will also improve performance by decreasing communication overhead by removing remaps necessitated by vertical decomposition.
5. New physics will add significant new computational and communication demands. Ability to use more MPI processes in physics than in dynamics will improve scalability and performance on the XT3 when the physics is more expensive than the dynamics.

Source Material

A. Mirin and W. Sawyer, *A Scalable Implementation of a Finite-Volume Dynamical Core in the Community Atmosphere Model*, International Journal for High Performance Computer Applications, 19(3), August 2005, pp. 203-212.

W. Putman, S-J. Lin, and B-W. Shen, *Cross-Platform Performance of a Portable Communication Module and the NASA Finite Volume General Circulation Model*, International Journal for High Performance Computer Applications, 19(3), August 2005, pp. 213-223.

L. Oliker, J. Carter, M. Wehner, A. Canning, S. Ethier, A. Mirin, G. Bala, D. Parks, P. Worley, S. Kitawaki, and Y. Tsuda, *Leading Computational Methods on Scalar and Vector HEC Platforms*, in Proceedings of the ACM/IEEE International Conference for High Performance Computing, Networking and Storage (SC05), Seattle, WA, November 12-18, 2005.

P. Worley and J. Drake, *Performance Portability in the Physical Parameterizations of the Community Atmospheric Model*, International Journal for High Performance Computer Applications, 19(3), August 2005, pp. 187-201.

P. Worley, *Benchmarking using the Community Atmosphere Model*, in Proceedings of the 2006 SPEC Benchmark Workshop, Austin, TX, January 23, 2006.