CCSM4 – Toward 100k Processors

Tony Craig

CCSM Software Engineering Group
NCAR, Boulder, CO

http://www.ccsm.ucar.edu/

Big Interdisciplinary Team!

- **Contributors:**

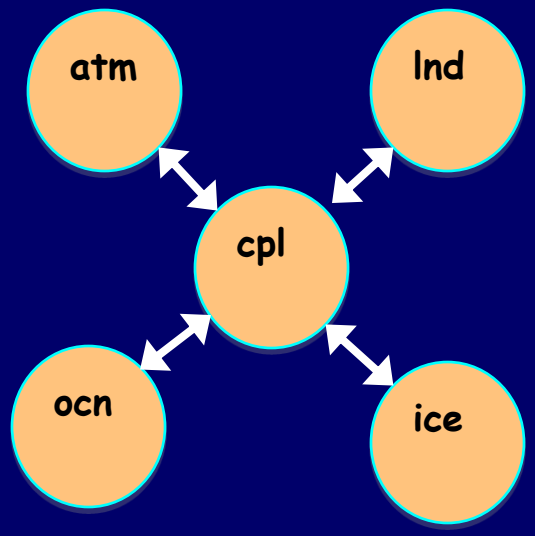
D. Bader (ORNL)
D. Bailey (NCAR)
C. Bitz (U Washington)
F. Bryan (NCAR)
T. Craig (NCAR)
A. St. Cyr (NCAR)
J. Dennis (NCAR)
B. Eaton (NCAR)
J. Edwards (IBM)
B. Fox-Kemper (MIT,CU)
N. Hearn (NCAR)
E. Hunke (LANL)
B. Kauffman (NCAR)
E. Kluzek (NCAR)
B. Kadlec (CU)
D. Ivanova (LLNL)
E. Jedlicka (ANL)
E. Jessup (CU)
R. Jacob (ANL)
P. Jones (LANL)
J. Kinter (COLA)
A. Mai (NCAR)
S. Mishra (NCAR)
S. Peacock (NCAR)
K. Lindsay (NCAR)
W. Lipscomb (LANL)
R. Loft (NCAR)
R. Loy (ANL)
J. Michalakes (NCAR)
A. Mirin (LLNL)
M. Maltrud (LANL)
J. McClean (LLNL)
R. Nair (NCAR)
M. Norman (NCSU)
N. Norton (NCAR)
T. Qian (NCAR)
M. Rothstein (NCAR)
C. Stan (COLA)
M. Taylor (SNL)
H. Tufo (NCAR)
M. Vertenstein (NCAR)
J. Wolfe (NCAR)
P. Worley (ORNL)
M. Zhang (SUNYSB)

- **Funding:**
  - DOE-BER CCPP Program Grant
    - DE-FC03-97ER62402
    - DE-PS02-07ER07-06
    - DE-FC02-07ER64340
    - B&R KP1206000
  - DOE-ASCR
    - B&R KJ0101030
  - NSF Cooperative Grant NSF01
  - NSF PetaApps Award

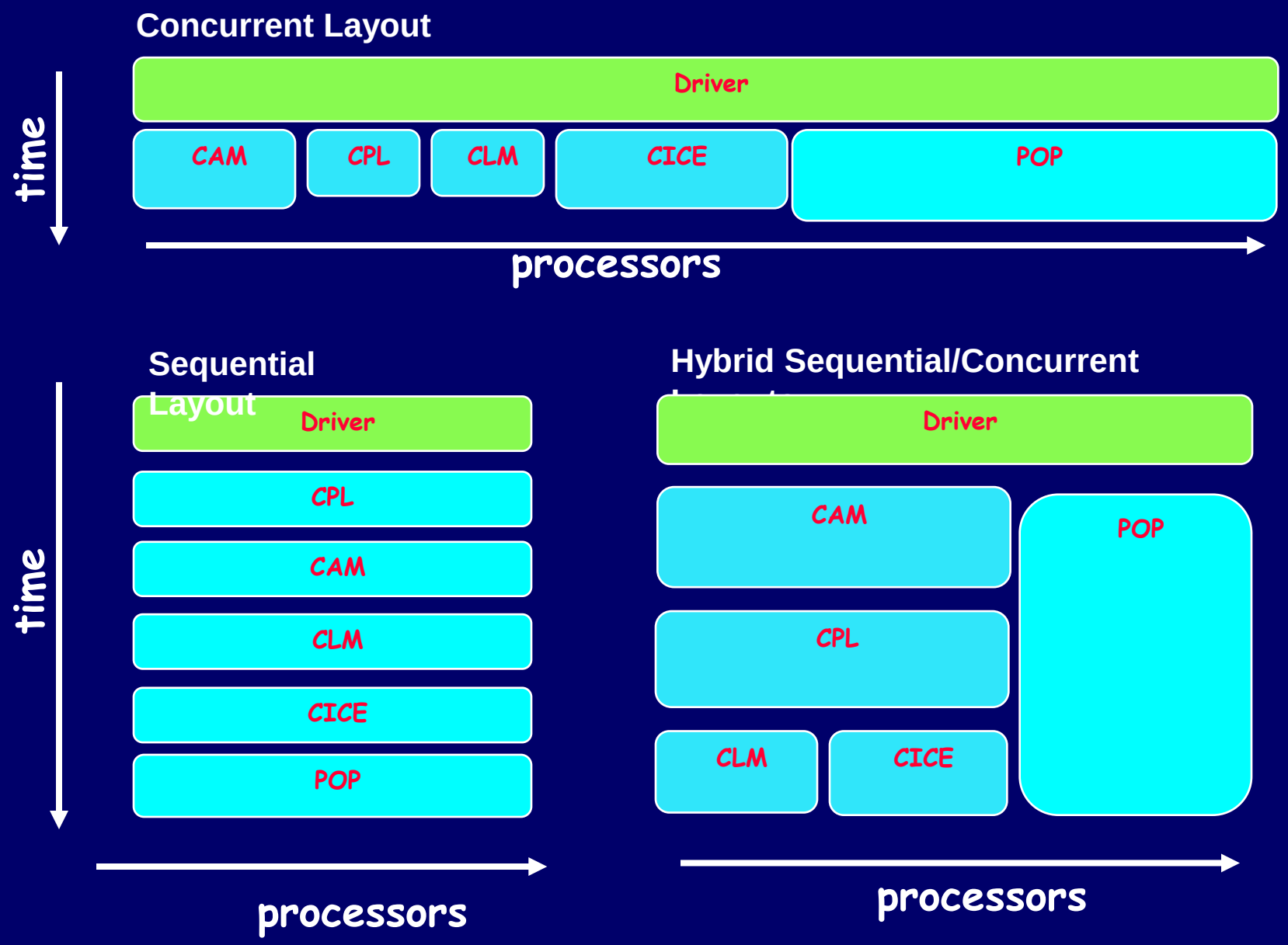
- **Computer Time:**
  - Blue Gene/L time:
    - NSF MRI Grant
      - NCAR
      - University of Colorado
      - IBM (SUR) program
    - BGW Consortium Days
      - IBM research (Watson)
    - LLNL
    - Stony Brook & BNL
  - CRAY XT time:
    - NICS/ORNL
    - NERSC
    - Sandia

What is CCSM4

- Community Climate System Model, a coupled climate model used for basic climate research, contributions to IPCC reports, and hindcasts and forecasts of past, present, and future climates.
- Consists of 4 geophysical component models (atmosphere, land, ocean, sea ice) *on potentially different grids* that exchange *boundary data* with each other ***only*** via communication with a coupler (hub and spoke architecture)
- Long History
- Large Community
- Diverse code base: >1M lines
  - Fortran 90 (mostly)
  - Developed over 20+ years
  - SPMD with MPI+OpenMP



New CCSM4 Architecture



A Few Basic Facts

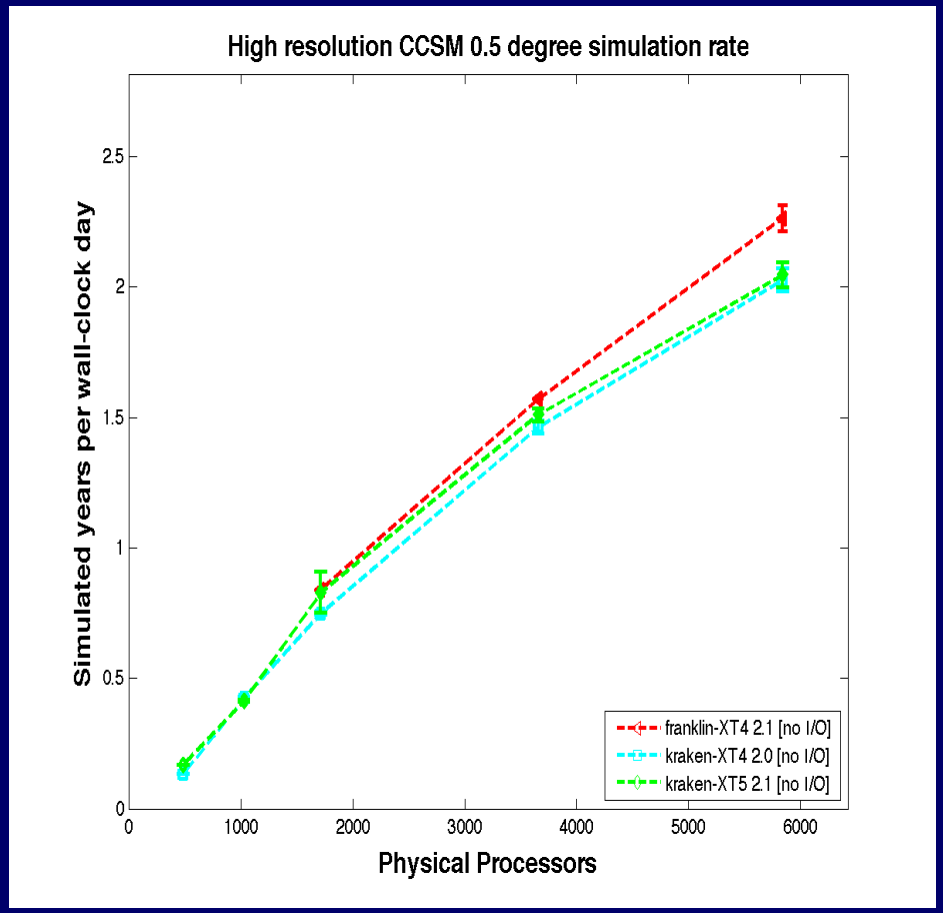
- Typical Resolutions are 0.5 to 3 degrees globally with < 60 vertical levels
- Finite difference type implementations (3d orthogonal+)
- Lots of vertical physics, typically decompose in the horizontal dimension
- Each component is relatively independent
- Each component has "physics" and "dynamics"
- Many diverse algorithms
- Time varying solution
- Want to resolve the diurnal cycle; use a coupling timestep of 1800 seconds (30 minutes)
- Must run the model for multiple centuries; target 5 years per day minimum throughput

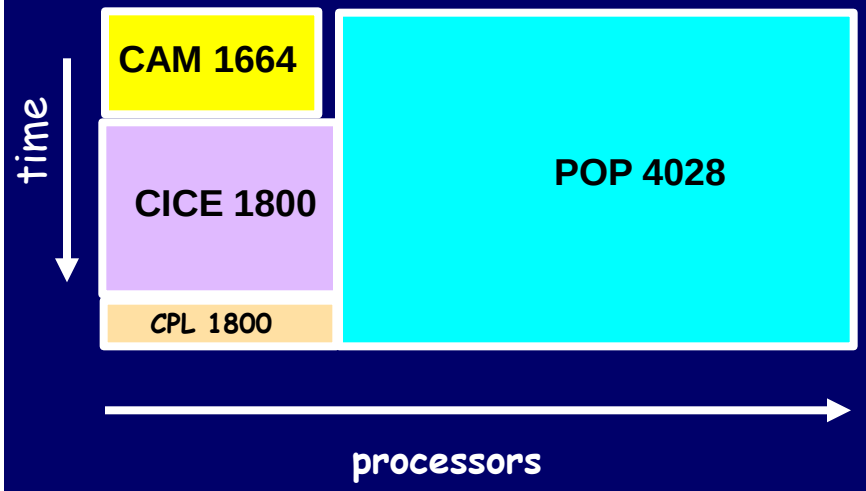
Challenges to CCSM4 Scaling

- Communication
  - Lots of Neighbor Communication
  - Reproducible global sums across different decompositions
  - Iterative algorithms that require global sums (ocean barotropic solver)
- Internal Model Load Balance
  - Spatial variability of cost (sea ice, vegetation, atm convection, radiation, clouds)
  - Temporal variability (diurnal and seasonal)
- IO – single netcdf file
- Memory Use – non scalable memory, memory access
- Resolving Timescales: at 1 degree resolution
  - 9/13 years/day on IBM p6 256/512 pes (bluefire)
  - 5/10 years/day on IBM BG/P 1000/4000 pes (intrepid)
  - 15 years/day on Cray XT5 2000 pes (jaguarpf)
  - Notes:
    - 15 years/day ~ 6 seconds/model day or 0.12 seconds/coupling period
    - 4000 pes -> 20 horizontal grid cells/pe at 1 degree resolution
    - A doubling of each horizontal dimension increases scalability by ~4x but reduces throughput by 2x (for CCSM in an ideal world)

CCSM4 High (0.5/0.1) Resolution

A New Milestone on Cray XT5





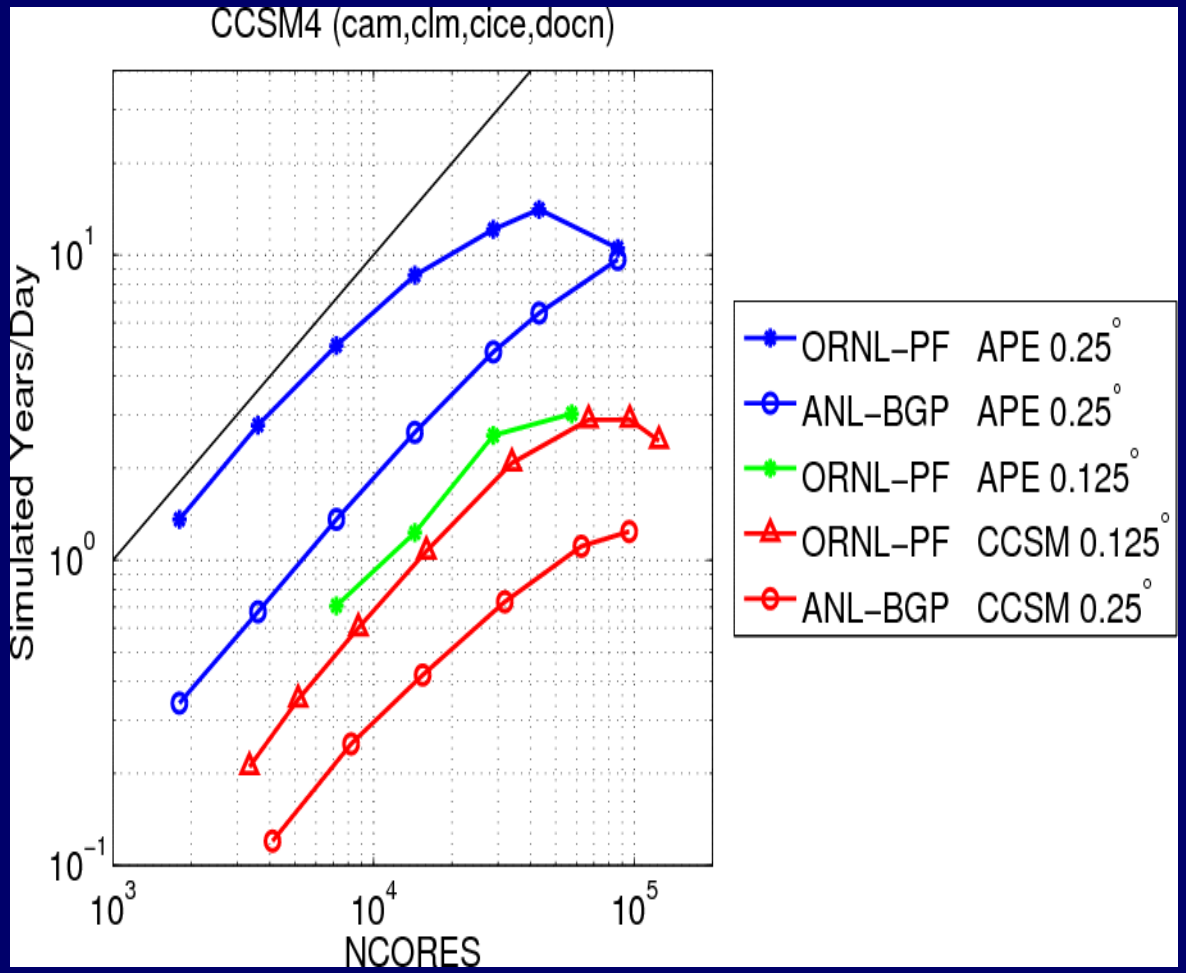
* 0.5 degree atm/lnd, 0.1 degree ocn/ice
* 1.9 sypd on 5848 kraken XT5 pes
* 155 years completed
* 100TB of data generated (generating 0.5-1 TB per wall clock day)
* 18M CPU hours used

(Courtesy of John Dennis)

CAM/HOMME Dycore

Cubed-sphere grid overcomes dynamical core scalability problems inherent with lat/lon grid

Work of Mark Taylor (SciDAC), Jim Edwards (IBM), Brian Eaton(CSEG)



PIO library used for all I/O (work COULD NOT have been done without PIO)

- **BGP (4 cores/node): Excellent scalability down to 1 element per processor (86,200 processors at 0.25 degree resolution).**
- **JaguarPF (12 cores/node): 2-3x faster per core than BGP, scaling not as good - 1/8 degree run loosing scalability at 4 elements per processor**

This was not technically possible two years ago, so

How Did We Get Here

- Extend OpenMP capability in all components
- Refactor to remove global arrays in IO and initialization
- New Decomps
  - Space filling curves, block decomps
  - 3d decomps in atm model for dynamics
  - Physics/Dynamics rearrangers
- New algorithms
  - Homme dycore
- Tune Communication Routines
  - Message Aggregation
  - Flow Control
- PIO – parallel IO library
  - http://code.google.com/p/parallelio/

Challenges at Exascale

- Reliability and Fault Tolerance !!
  - Hardware failure
  - OS interupts
- MPI Memory Use
- Internal Physics Load Balance
- Flops/Communication
- IO – amount of data, post-processing, asychronous IO
- Machines are getting bigger (wrt processors) more quickly than CCSM is increasing resolution
- No "Big Gorilla" Algorithm/Cost in CCSM4
- Physics getting more expensive (adding chemistry, tracers, complexity) -> model more expensive but NOT more scalable
- Completing simulations in a timely manner
- Single processor performance
- Ensembles are only a partial solution