Acceleration of SUZI with hybrid MPI/OpenMP programming

Don Dazlich

CMMAP Mtg January 5-7, 2016, Boulder, CO

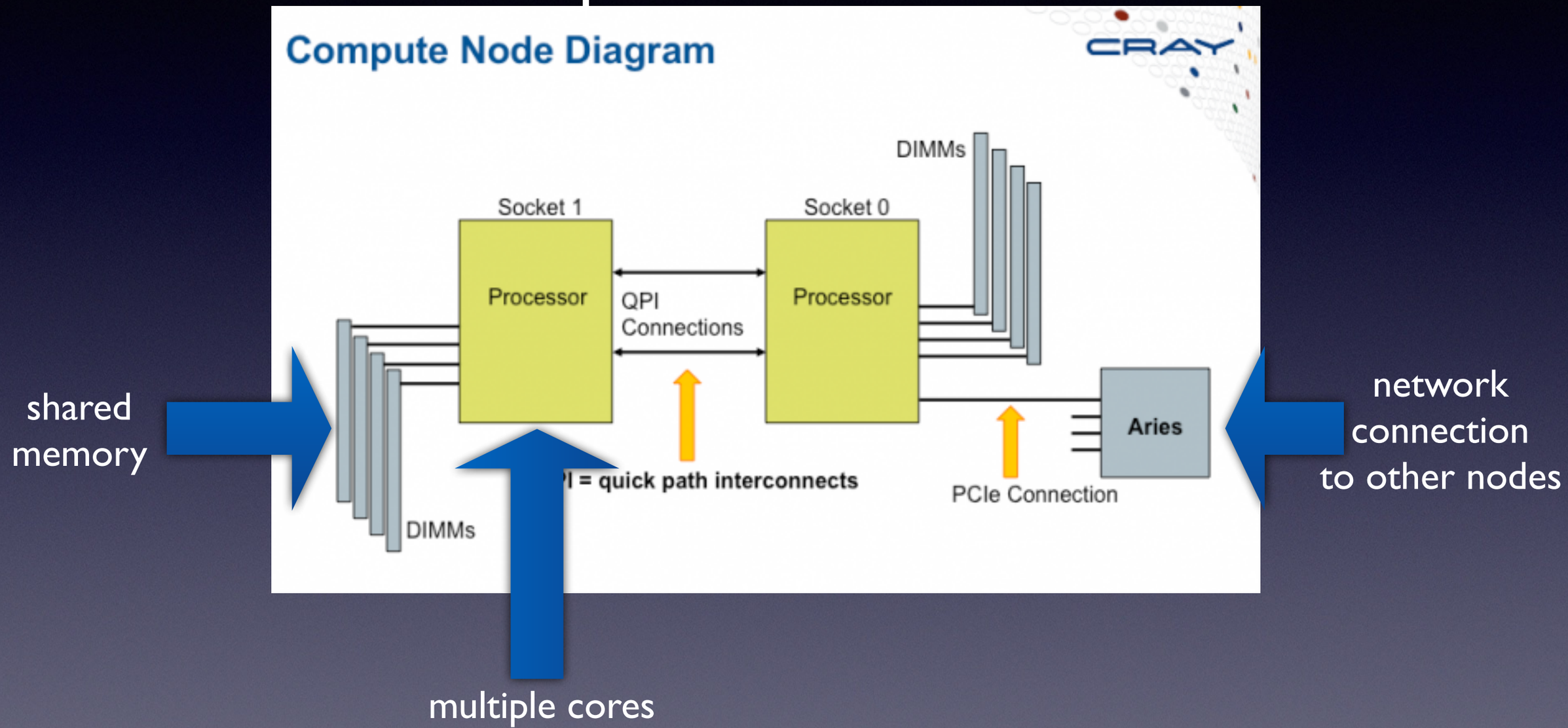
Dynamical Frameworks Session

The computational challenge

- 2 GCM on O(10^2) cores
- 4km GCRM on O(10^5)cores - 4000 times more points x 30 times more timesteps
- 2 SP-GCM - about 100 times the computation as GCM; 100 times slower on the same number of cores.

Supercomputer architecture

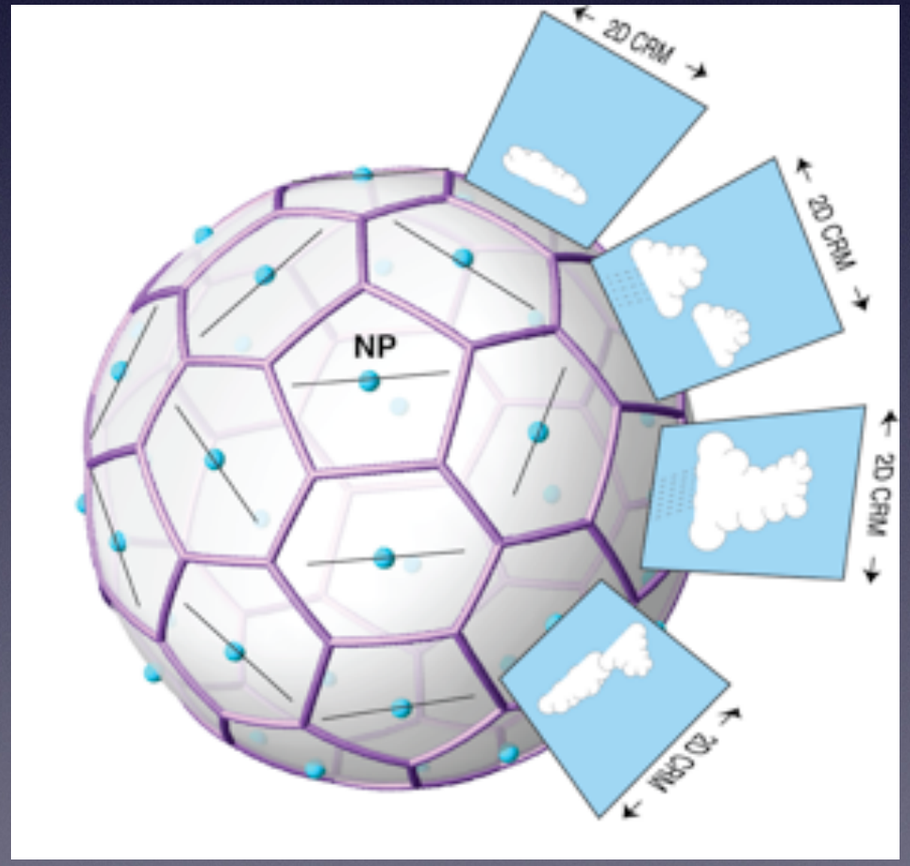
Example: NERSC Edison



2 processors x 12 cores/processor = 24 cores/node

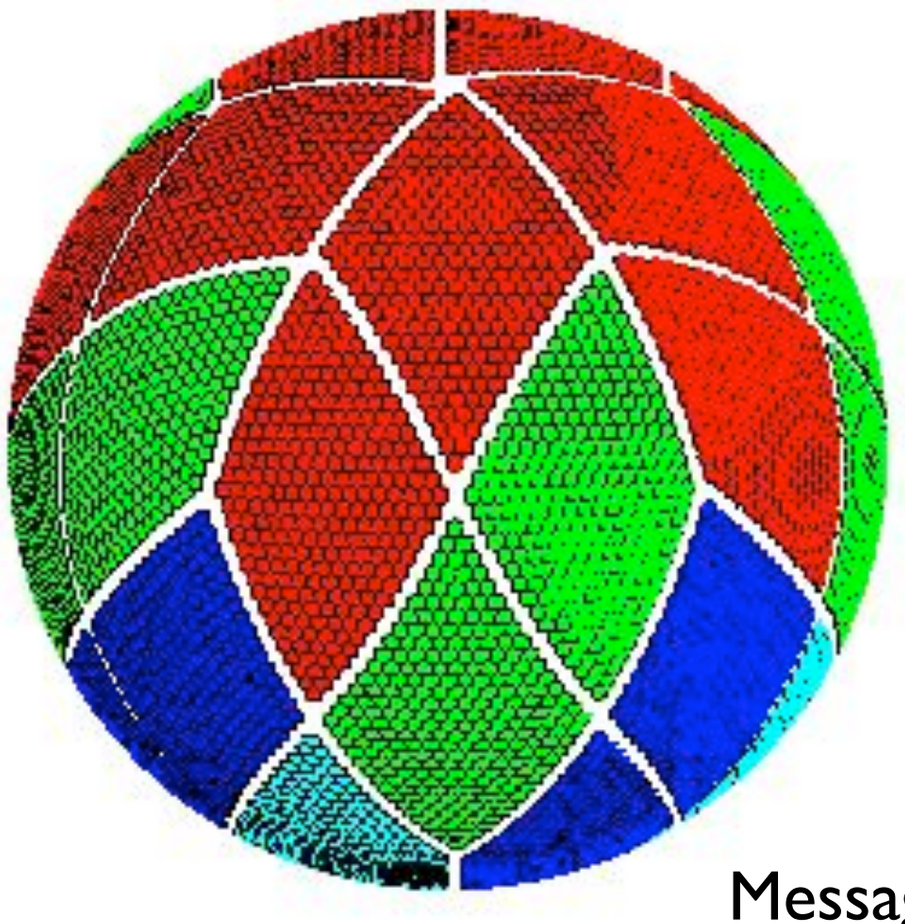
SUZI*

*Superparameterized physics
Unified system of equations
Z-grid™ horizontal discretization
Icosahedral grid

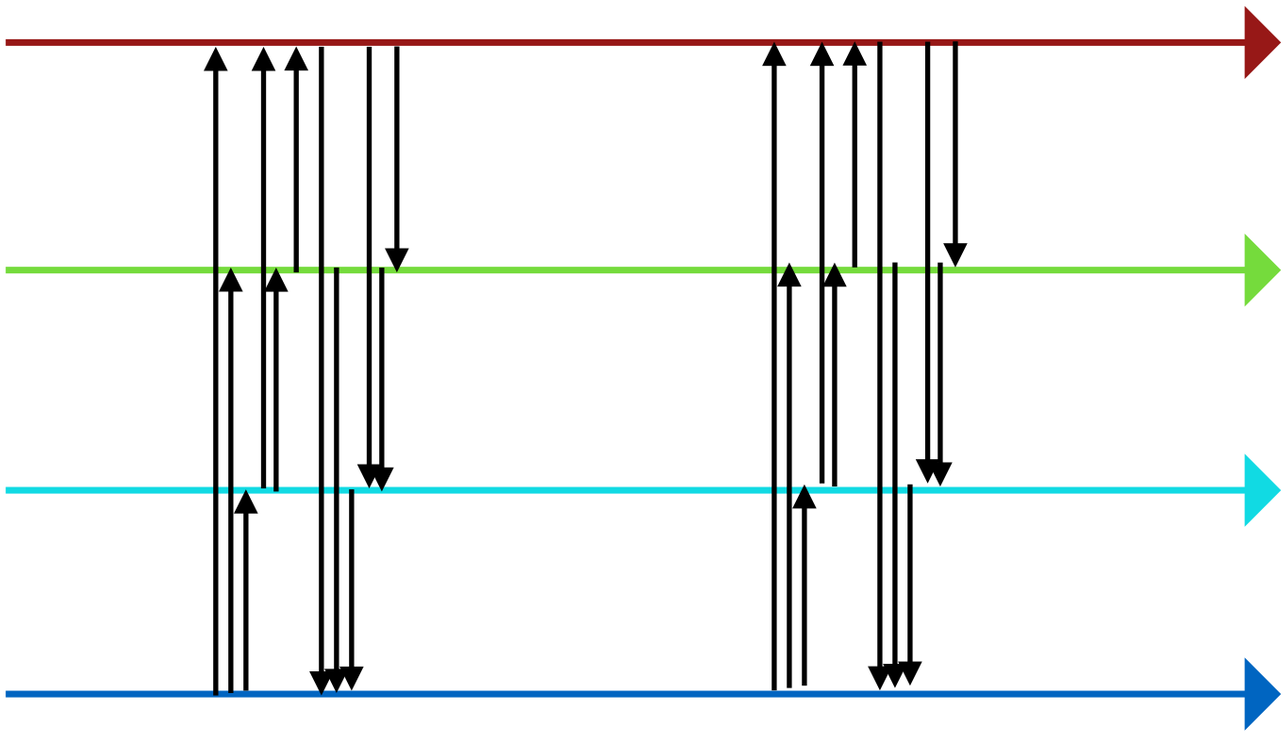


SUZI - computational logic

Domain decomposition



Message passing



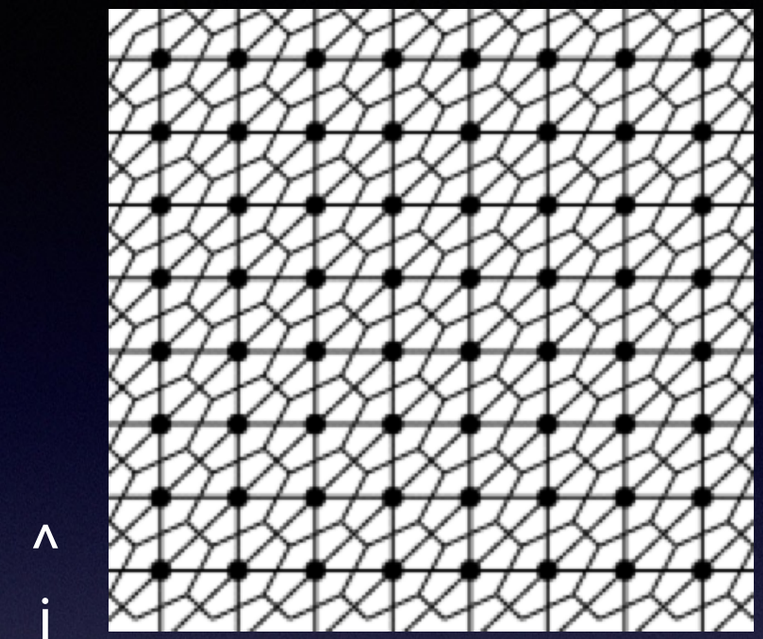
Message passing paradigm:

Each process sees only its own local memory

Often requires extensive code rewrite to implement

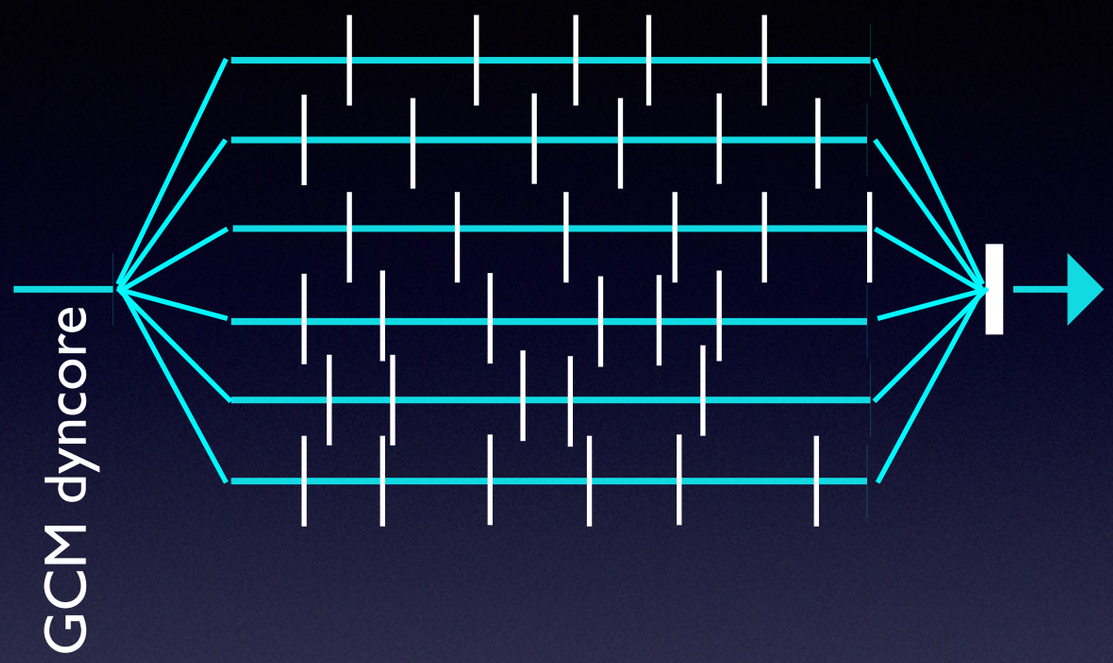
Implemented through subroutine calls and provides explicit control to programmer

Current SP implementation



The CRM computations for each grid cell are independent and could proceed simultaneously were more processors allotted.

Shared memory programming - OpenMP




```
!$OMP PARALLEL DO COLLAPSE(2) SCHEDULE(DYNAMIC)       &
!$OMP&         PRIVATE(ii, jj, k, i,         &
!$OMP&                 nstatsteps, n, threadnum,    &
!$OMP&                 icyc, tendqwv, tendqcl, tendqci,            &
!$OMP&                 u2k, v2k, w2k, tautemu, tautemv, tautem,    &
!$OMP&                 cld3d, cld2d,                               &
!$OMP&                 count2, count3, start2, start3, t_phys_tend )  &
!$OMP&         COPYIN(fzero, qrad, radlwup, radlwdn,               &
!$OMP&                radswup, radswdn, radqrlw, radqrsw )


do jj = 1,jm
do ii = 1,im
.
.
.
enddo  ! gcm i loop
enddo  ! gcm j loop
!$OMP END PARALLEL DO
```

Have one (or a few) MPI tasks per node. Spawn threads on the other cores of a node. The threads all have access to the same memory per node.

OpenMP accomplishes this with compiler directives added to code that look like comments. Identify parallel areas (like the grid cell do loops) and bracket it with directives. Little code needs rewriting, although one may need to rewrite code for optimization sake.

Because the threads have access to the same memory one must be careful that the threads don't interfere writing and reading from it (race conditions). One must also make sure that variables are appropriately declared shared or private. OpenMP code can be difficult to debug. Results are often unreproducible.

Hybrid MPI/OpenMP in SUZI

- Edison example: 10k (2 degree) grid points, launch 160 tasks on 80 nodes. Launch 12 threads per task, each thread does 5 or 6 grid cells - 1920 cores used.
- Implementing this piecemeal. First iteration, comment all CRM code except the advective forcing routine. This worked and sped up the CRM part of execution by a factor of 10 (12 possible).
- Have since turned on SGS. Stuck with radiation (clearsky).

Summary

- SP models have lots of data parallelism built into them beyond that of the GCM dynamical core, and the bulk of the computations are done in the CRMs.
- Have begun to implement OpenMP in SUZI. When complete a O(10) speedup is expected on Edison. Go from 8 simulated months per day to 7 years per day.
- Future computers to have more cores per node. Cori phase 1 32 cores per node, phase II 60 cores (MIC processor)
- Same principle should work with GPUs and OpenACC
- Feasible to do higher resolution CRMs, ensemble CRMs.