

# **An Introduction to NWChem**

David E. Bernholdt

**Computer Science and Mathematics Div.**

**[bernholdtde@ornl.gov](mailto:bernholdtde@ornl.gov)**

***<http://www.csm.ornl.gov/~bernhold/>***

# NWChem Project Background

- **Organization:** Lead by High Performance Computational Chemistry (HPCC) Group at the Pacific Northwest National Laboratory (PNNL)
- **Task:** Produce molecular modeling software tools capable of using MPPs to deliver 10-100 times more computing capability than previously available
- **Time:** 1993-1997 with intermediate deliverables of working code. (Active development continues under other projects)
- **Context:** Various experiments in parallel computing in chemistry. Little or no routine “production” use.

- Two crucial factors for success:
  - Parallel programming model combining ease of use, portability, and scalable high performance
  - Software development approach that would scale with the complexity and capability of the code
- All of HPCC Group familiar with common problems of other large computational chemistry packages
- Some of HPCC Group familiar enough with OO methods to recognize value
- Virtually no actual OO experience in HPCC Group
  - Fortran and C are the *de facto* languages of comp. chem.
  - Insufficient time to learn new language & meet deliverables
  - C++ performance concerns

# A Very Pragmatic Approach

- Use OO *design* to the extent possible
- Implement in Fortran77 and C

## Results...

- At age 7, NWChem 3.3.1 includes
  - 564,000 lines of Fortran77
  - 166,000 lines of C
  - contributions from more than 40 people
  - significant incorporation of legacy code
  - broadest functionality and best efficiency of any parallel computational chemistry package

# Structure of NWChem

Chemistry Methods

MD SCF DFT MP2 CI ...

Chemistry Support

atomscf basis driver geom NWints  
property stepper symmetry

General Support

chemio fft global inp  
ma peigs pstat rtdb

External, OS, etc.

MPI MPL / SHMEM / ... LAPACK ...

# Parallel Programming Libraries

- Developed in conjunction with NWChem, but intended from the start to be independent and general
- PeIGS -- Parallel eigensolver
  - Answers requirements of quantum chemistry applications (orthogonality, eval clusters) not currently addressed by better known packages
  - Performance comparable to or better than other packages for quantum chemistry applications
- Global Array Toolkit
- Both in public domain; multiple users outside of NWChem

# The Global Array Toolkit

## Primitive – collective

`ga_create`, `ga_duplicate`, `ga_destroy`,  
`ga_sync`

## Primitive -- one-sided (MIMD)

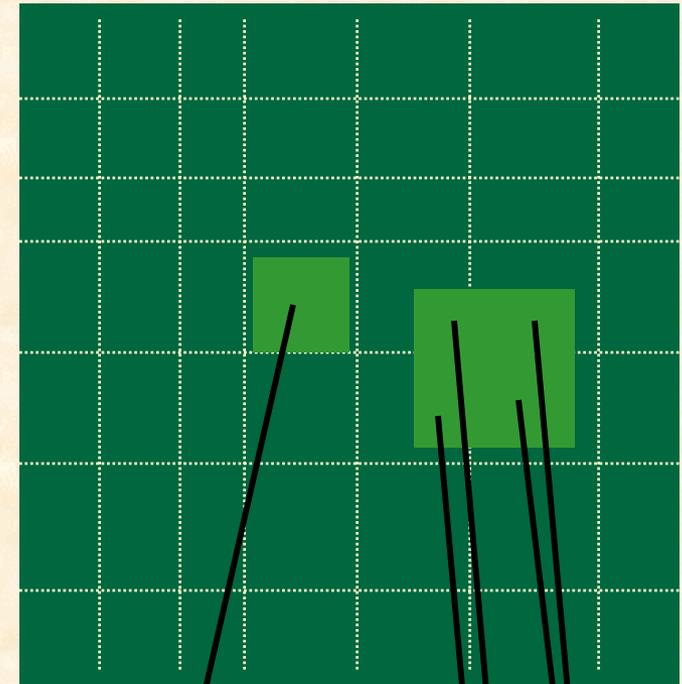
`ga_get`, `ga_put`, `ga_acc`, `ga_access`,  
`ga_release`, `ga_read_inc`, `ga_scatter`,  
`ga_gather`

## Data Parallel Building Blocks

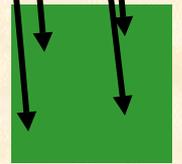
`ga_locate`, `ga_locate_region`,  
`ga_distribute`, `ga_access`. `ga_release`,  
`ga_release_update`

## Linear Algebra – collective

`ga_zero`, `ga_ddot`, `ga_dgemm`, `ga_scale`,  
`ga_add`, `ga_fill`, `ga_symmetrize`,  
`ga_transpose`, `ga_diag`, `ga_lu_solve`,  
`ga_cholesky`, `ga_spd_invert`, `ga_llt_solve`,  
`ga_solve`, `ga_copy`



Process A



Process B

# GA Programming Model Concepts

- Modern computers have non-uniform memory access times (NUMA): registers, cache, local memory, **remote memory**, virtual memory, disk, ...
- GA model exposes NUMA nature to programmer (ga\_get, ga\_put, etc) but otherwise appears to be globally shared memory
- “Traditional” shared memory models try to hide any local/remote non-uniformity in hardware or operating system (c.f. distributed shared memory systems)
- In order to produce scalable, portable, and efficient parallel algorithms, programmer must take into account NUMA nature of system

# NWChem as “Community Software”

- Original development group small, and localized at PNNL.
- Early external contributors brought in to obtain specific functionality
- Intent was always to make code available at low or no cost
- Distribution policies have slowly broadened. Only in last 2-3 years that “anybody” could get binaries; source still requires justification.
- Broadening user base means non-collaborators start to become contributors
- Support still a major concern w.r.t. distribution

# Options for Working with NWChem

- Improvements/extensions to existing functionality should be contributed back to PNNL for incorporation
  - May want to get access to CVS repository for code before doing anything substantial on this line
- Anything which is readily supportable, sufficiently robust, and of general interest might be considered for actual incorporation into NWChem distribution
- Anything can be written to work with (even link into) NWChem and distributed separately
- NWChem and associated parallel tools can be treated as a library of resources for use in separate codes

# My Place in the Scheme of Things

- PNNL (1993-1995)
  - Prototyping of algorithms (esp. SCF, MP2), tools
  - Design of Global Array Toolkit
  - Overall architecture & design of NWChem
  - RI-MP2 module
- Syracuse University (1995-2000)
  - Continued work on RI-MP2, RI-MP2 gradients
  - Local MP2
  - (old) coupled cluster gradients
  - Other (non-chemistry) computational science, esp. DoD HPC Modernization Program
- ORNL
  - Computer Science
  - Common Component Architecture
  - Interested in developing more chemistry activities, perhaps in time a computational chemistry group

# Common Component Architecture

- Framework for the construction of complex HPC software systems from small (parallel/distributed) components
  - Multi-scale & multi-physics problems
  - Simplify large monolithic systems (i.e. NWChem)
- Analogous to component models being developed for CORBA or EJB, but high performance, targeted to simulation science
- Allow application developers to focus on what they do best, by making it easier to take advantage of computational components developed by others

# NWChem Capabilities

- *Primarily* molecular systems
- *Primarily* Gaussian bases
- Molecular dynamics
- Hartree-Fock Self-Consistent Field
- Density Functional Theory
- Multiconfigurational SCF
- MP2, RI-MP2
- Coupled cluster (new code released soon)
- CI, MRCI (through interface w/ COLUMBUS)

- Effective core potentials (ECPs)
  - Reduce number of electrons treated explicitly
  - Spin-orbit effects
  - Relativistic effects
- True relativistic methods
  - Spin-orbit DFT
  - Douglas-Kroll-Hess
  - Relativistic integrals (build your own method)
  - Access to other relativistic codes (external to NWChem)
- Mixed quantum+classic dynamics (QM/MM)
  - Treat region of interest with quantum mechanics and surrounding region with molecular mechanics. Do MD or Monte Carlo simulations

- Pseudopotential Plane-Wave DFT
  - Periodic and aperiodic boundary conditions
  - Hamann and Troulier-Martins norm-conserving pseudopotentials, Vanderbilt under development
  - Car-Parrinello or steepest-descent (CG) dynamics, simple Nose-Hoover thermostat
  - LDA, LSDA, PBE96 functionals
  - 128 atom SiC cluster on 32 node IBM SP 2
- Periodic (Gaussian) DFT (GAPSS)
  - Correlation functionals: VWN, LYP, Perdew86, Perdew-Burke-Ernzerhof (PBE96)
  - Exchange functionals: Slater, Becke88, PBE96
  - Gradients for nuclear positions, not cell size
  - “Hand” optimization or nuclear coordinates required
  - Has been used for calculations of up to 50 atoms per unit cell on EMSL IBM SP
  - Code may be rusty (no development in past year)

# NWChem vs GAMESS, Gaussian, ...

- GAMESS, Gaussian are molecular only
- GAMESS: SCF, MCSCF, MP2, CI, DFT(?)
- Gaussian: SCF, DFT, MPn, approx. & true coupled cluster
- GAMESS reasonably parallel
- Gaussian parallel for small numbers of nodes