NWChem: Computational Chemistry
Software for Parallel Computers

Edoardo Aprà
William R. Wiley Environmental Molecular Sciences Laboratory,
Pacific Northwest National Laboratory
Outline of Talk

- Overview of NWChem
- Parallel Performance
- Hardware and Software requirements
Molecular Science Software Group

Edoardo Aprà  
Eric Bylaska  
Mahin Hackler  
Bert deJong  
So Hirata  
Lisa Pollack  
Tjerk Straatsma  
Marat Valiev  
Theresa Windus

Gary Black  
Liz Jurrus  
Carina Lansing  
Bruce Palmer  
Karen Schuchardt  
Eric Stephan  
Erich Vorpagel

Manoj Kumar Krishnan  
Jarek Nieplocha  
Bruce Palmer  
Vinod Tipparaju

Interface Between the User and the Software

User

Science  
Computer

Global Arrays

parallel computing libraries and tools software

Interface with the Computer

high performance computational chemistry software

Interface with the Science

NWChem

extensible computational chemistry environment

Ecce
Why NWChem Was Developed

- Developed as part of the construction of the Environmental Molecular Sciences Laboratory (EMSL)
- Envisioned to be used as an integrated component in solving DOE’s Grand Challenge environmental restoration problems
- Designed and developed to be a highly efficient and portable Massively Parallel computational chemistry package
- Provides computational chemistry solutions that are scalable with respect to chemical system size as well as MPP hardware size
NWChem Architecture

Run-time database

- Energy, structure, ...
- SCF energy, gradient, ...
- DFT energy, gradient, ...
- MD, NMR, Solvation, ...
- Optimize, Dynamics, ...

Generic Tasks
- Object-oriented design
- abstraction, data hiding, APIs

Molecular Calculation Modules
- Parallel programming model
- non-uniform memory access, global arrays, MPI

Molecular Modeling Toolkit
- Infrastructure
- GA, Parallel I/O, RTDB, MA, ...

Molecular Software Development Toolkit
- Program modules
- communication only through the database
- persistence for easy restart

Integral API
- Geometry Object
- Basis Set Object
- PeIGS

Parallel IO
Memory Allocator
Global Arrays

• Object-oriented design
• abstraction, data hiding, APIs
• Parallel programming model
• non-uniform memory access, global arrays, MPI
• Infrastructure
• GA, Parallel I/O, RTDB, MA, ...
• Program modules
• communication only through the database
• persistence for easy restart
Global Arrays

Distributed dense arrays that can be accessed through a shared memory-like style

Physically distributed data

single, shared data structure/
global indexing

e.g., access A(4,3) rather than buf (7) on task 2

Global Address Space
Global Arrays

- Shared memory model in context of distributed dense arrays
- Complete environment for parallel code development
- Compatible with MPI
- Data locality control similar to distributed memory/message passing model
- Extensible and scalable
- Compatible with other libraries: ScaLapack, Peigs, etc ...
- Part of bigger system for NUMA programming:
  - Parallel I/O extensions: Disk Resident Arrays
  - Per-processor private files Exclusive Access Files
Global Arrays and MPI are completely interoperable. Code can contain calls to both libraries.

Application programming language interface

Global Arrays and MPI are completely interoperable. Code can contain calls to both libraries.

Structure of GA

Message Passing
Global operations

distributed arrays layer
memory management, index translation

ARMCI
portable 1-sided communication
put, get, locks, etc

system specific interfaces
LAPI, GM/Myrinet, threads, VIA...

Fortran 77 C C++ Python Babel

F90 Java

Babel

Application programming language interface

Global Arrays and MPI are completely interoperable. Code can contain calls to both libraries.
Mirrored Arrays

- Create Global Arrays that are **replicated** between SMP nodes but **distributed** within SMP nodes
- Aimed at fast nodes connected by relatively slow networks (e.g. Beowulf clusters)
- Use memory to **hide latency**
- Most of the operations supported on ordinary Global Arrays are also supported for mirrored arrays
- Global Array toolkit augmented by a merge operation that adds all copies of mirrored arrays together
- Easy conversion between **mirrored** and **distributed** arrays
Mirrored Arrays (cont.)

- Distributed
- Mirrored
- Replicated
Mirrorred Arrays in NWChem/DFT

![Graph showing wall time (seconds) vs. number of processors for different configurations: D - D: Ethernet Distributed, M - M: Ethernet Mirrored, d - d: Elan3 Distributed, m - m: Elan3 Mirrored, and Myrinet Distributed, Myrinet Mirrored. The graph demonstrates performance improvements with increasing processor count.](attachment:image)
NWChem Capabilities

► **Quantum Mechanical Capabilities:**
  - Hartree-Fock & density functional theory at the local and nonlocal levels (with $N^3$ and $N^4$ formal scaling) energies, gradients, & second derivatives. Linear scaling quadrature and exchange.
  - TDDFT
  - Multiconfiguration self consistent field (MCSCF) energies and gradients.
  - Many-body perturbation theory energies and gradients.
  - Effective core potential energies, gradients, and second derivatives.
  - Coupled cluster [CCSD and CCSD(T)] and configuration interaction energies.
  - Tensor Contraction Engine module, that can generate unrestricted CISD, CISDT, CISDTQ, LCCD, CCD, LCCSD, CCSD, QCISD, CCSDT, CCSDTQ, MBPT(2), MBPT(3), MBPT(4) wavefunctions
  - Segmented and generally contracted basis sets including the correlation-consistent basis sets.
  - Plane-wave pseudo-potential codes (periodic and free-space) with dynamics; PAW.

► **Classical Mechanical Capabilities:**
  - Energy minimization; molecular dynamics simulation; ab initio dynamics
  - Free energy calculation
  - Supports variations such as: multiconfiguration thermodynamic integration or multiple step thermodynamic perturbation, first order or self consistent electronic polarization, simple reaction field or particle mesh Ewald, and quantum force-field dynamics

► **Mixed QM + MM models and ONIOM**
 Supported Platforms

- CRAY T3D and T3E
- IBM SP
- SGI MIPS and Altix SMP systems
- SUN workstations running SOLARIS
- HP PA-RISC workstations running HPUX
- Fujitsu VX/VPP
- HP Alpha SMP servers running Tru64 or Linux.
- HP Alpha SC series.
- Workstation networks
- IA32-based workstations running Linux or FreeBSD
- IA32 Linux Clusters with Giganet switch using the VIA protocol
- IA32 Linux Clusters with Myrinet switch using the GM software
- IA32-based PC running Microsoft 32-bit Windows
- PowerPC workstations running Linux
- Itanium1/2 servers running Linux or HPUX
- Linux IA32, Itanium and Alpha clusters with Elan3 switch
- AMD Opteron port in progress
- Itanium cluster with Elan4 port in progress
Parallel scaling of the DFT code of NWChem

Si$_8$O$_7$H$_{18}$
347 Basis f.
LDA wavefunction

Wall time (seconds)

number of CPUs
Parallel scaling of the DFT code of NWChem

![Graph showing parallel scaling of the DFT code of NWChem with various processors and configurations.]

- $\text{Si}_{28} \text{O}_{67} \text{H}_{30}$
- 1687 Basis functions
- LDA wavefunction
Parallel scaling of the DFT code of NWChem

Si$_{28}$O$_{148}$H$_{66}$

3554  Basis functions
LDA wavefunction

Wall time (seconds)

Number of CPUs

EDO Apra - MSSG - Pacific Northwest National Laboratory
Parallel scaling of the MP2 code of NWChem

H$_2$O$_7$

287 Basis functions

MP2 Energy & Gradient
Haloalkanedehalogenase: Accumulated Time per Step

Number of Processors

Scaling

IBM-SP

HP

Haloalkane dehalogenase
(41,259 atoms)
NWChem Molecular Dynamics

Haloalkanedehalogenase: Scaling

Haloalkanedehalogenase: Time per Step
Parallel efficiencies of the NWChem PSPW module for calculations of $C_{40}$, and $(H_2O)_{16}$ clusters. Calculations performed on a linux cluster made up of dual processor 500 MHz Pentium nodes connected via a Giganet high-speed network and on the EMSL IBM SP.
Hardware and Software requirements

- low latency and high bandwidth for
  - I/O
  - communication
- availability of large amount of aggregate memory and disk
- flat I/O model using local disk
- 64-bit addressing (required by highly correlated methods)
- extensive use of Linear Algebra:
  - BLAS
  - FFT
  - Eigensolvers
- use of other scientific libraries (e.g. MASS for exp and erfc)
Acknowledgements

- NWChem Development Team
- Global Array developers
- Environmental Molecular Sciences Laboratory (EMSL) operations supported by DOE’s Office of Biological and Environmental Research
How do you get NWChem?

- http://www.emsl.pnl.gov/pub/docs/nwchem => Register
- Website with lots of other NWChem information
- Print, fill-out, and sign site agreement form and fax
- back to PNNL, where Form will be signed by PNNL official and
- download information will be sent via fax
- nwchem-support@emsl.pnl.gov for HELP!
- Mailing lists: nwchem-users@emsl.pnl.gov and nwchem-developers@emsl.pnl.gov
NWChem authors