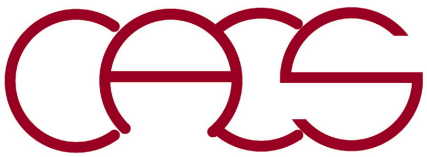
Grid Computing: Application to Science

Aiichiro Nakano

Collaboratory for Advanced Computing & Simulations
Department of Computer Science
Department of Physics & Astronomy
Department of Quantitative & Computational Biology
University of Southern California

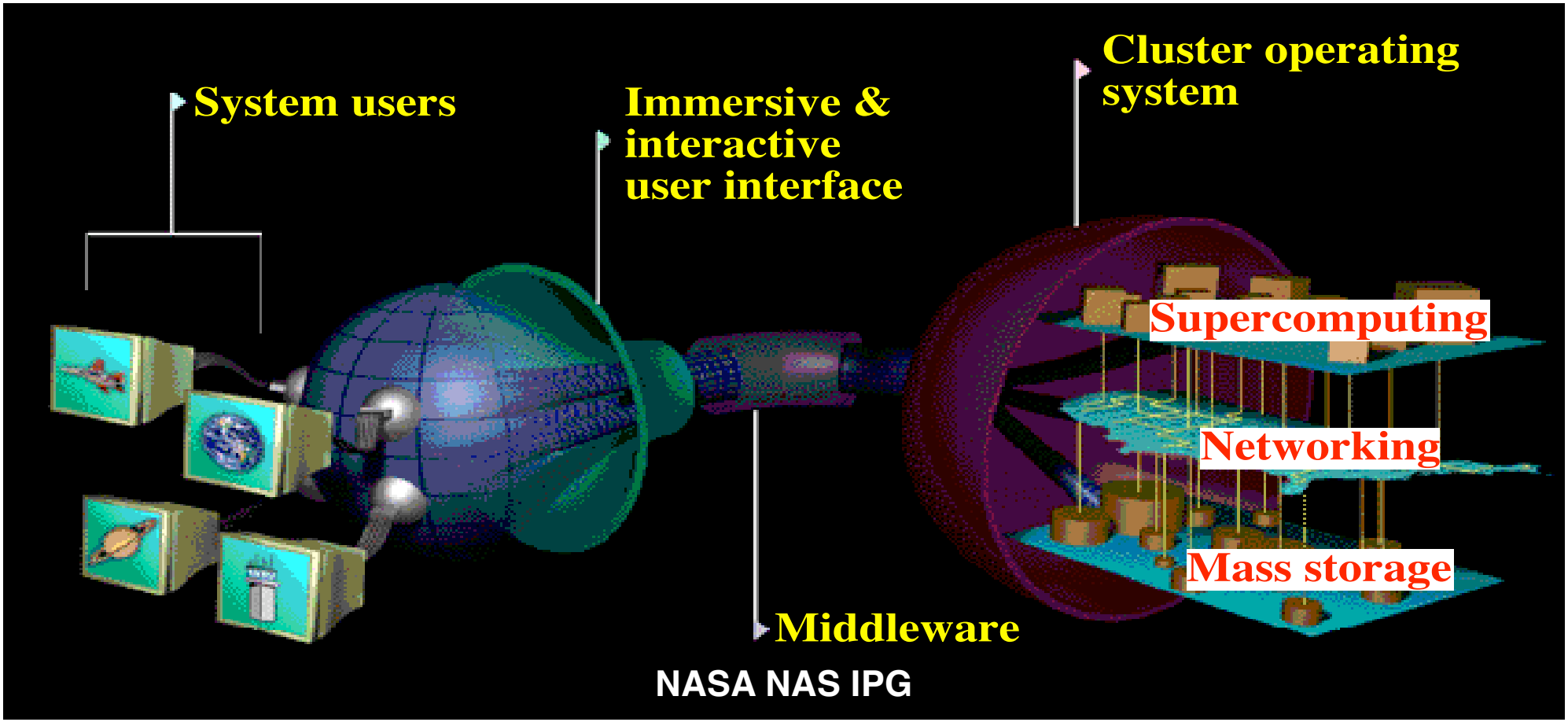
Email: anakano@usc.edu

Grid = gateway to exascale (fault resilience, latency hiding) & cloud computing



Grid Computing

- **World Wide Web:** Universal interface to digital library on the Internet
- **Information Grid:** Pervasive (from any place in the world at any time) access to everything (computing, mass storage, experimental equipments, distributed sensors, *etc*., on high-speed networks)



Application-Level Grid Tools

Grid programming models

- Message passing: MPICH-G2
- Remote procedure call: Ninf-G

Grid application types

- Metacomputing
- Parameter-sweep (high throughput) applications
- Workflow applications
- Portals: Thin-client, graphical user interfaces to the Grid

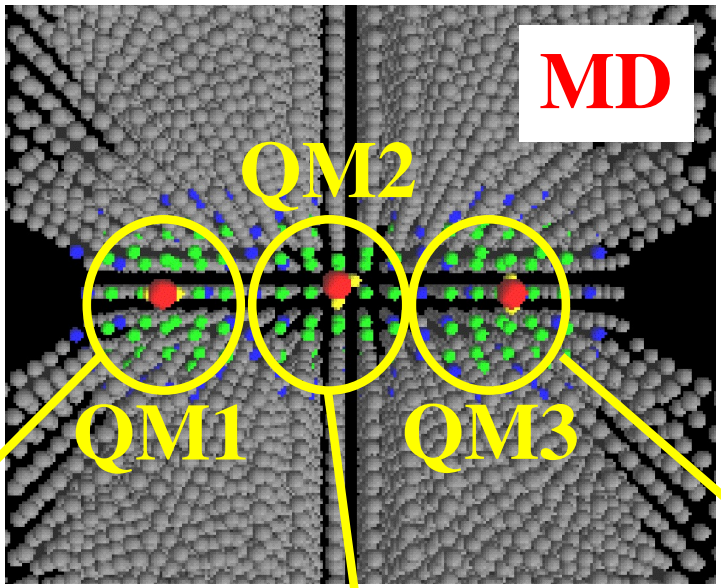
Outline

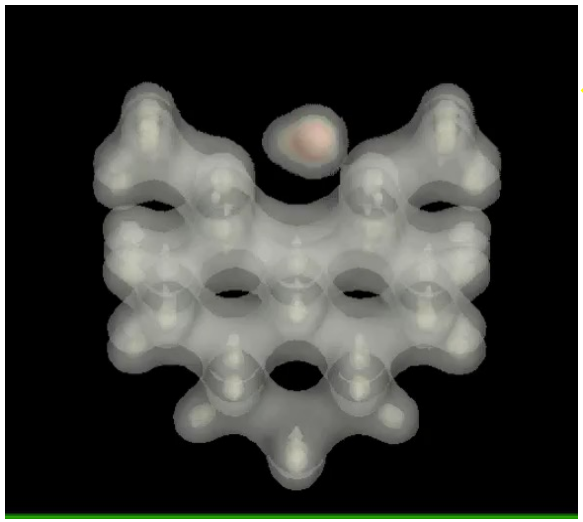
1. Grid programming
   - Metacomputing—multiscale MD/quantum-mechanical (QM) simulations: Grid-enabled MPI (MPI–G2)
   - Task farm: Grid remote procedure call (Ninf–G)
   - Sustainable & adaptive Grid supercomputing
2. Grid software
   - Globus toolkit
   - Open Grid Services Architecture (OGSA)

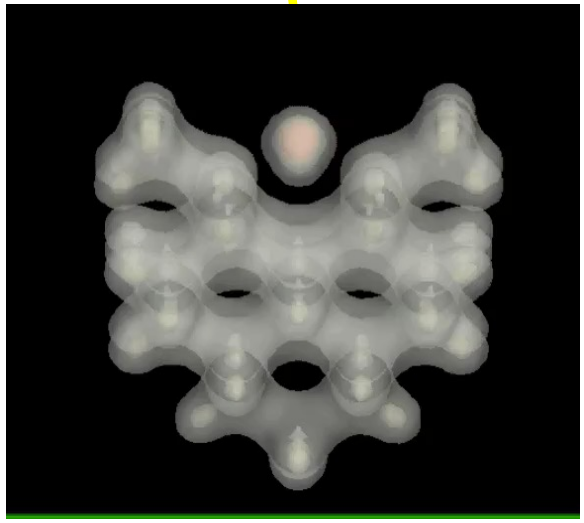
Grid Enabling: Multiple QM Clustering

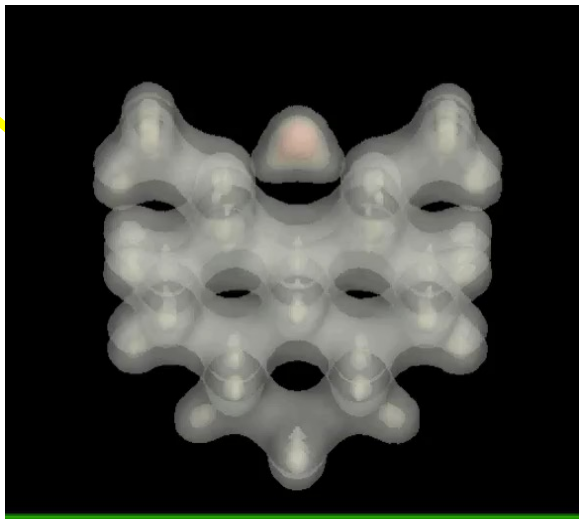
$$E = E_{\mathrm{MD}}^{\mathrm{system}} + \sum_{\mathrm{cluster}} [E_{\mathrm{QM}}^{\mathrm{cluster}}(\{\mathbf{r}_{\mathrm{QM}}\},\{\mathbf{r}_{\mathrm{HS}}\}) - E_{\mathrm{MD}}^{\mathrm{cluster}}(\{\mathbf{r}_{\mathrm{QM}}\},\{\mathbf{r}_{\mathrm{HS}}\})]$$

Divide-&-conquer



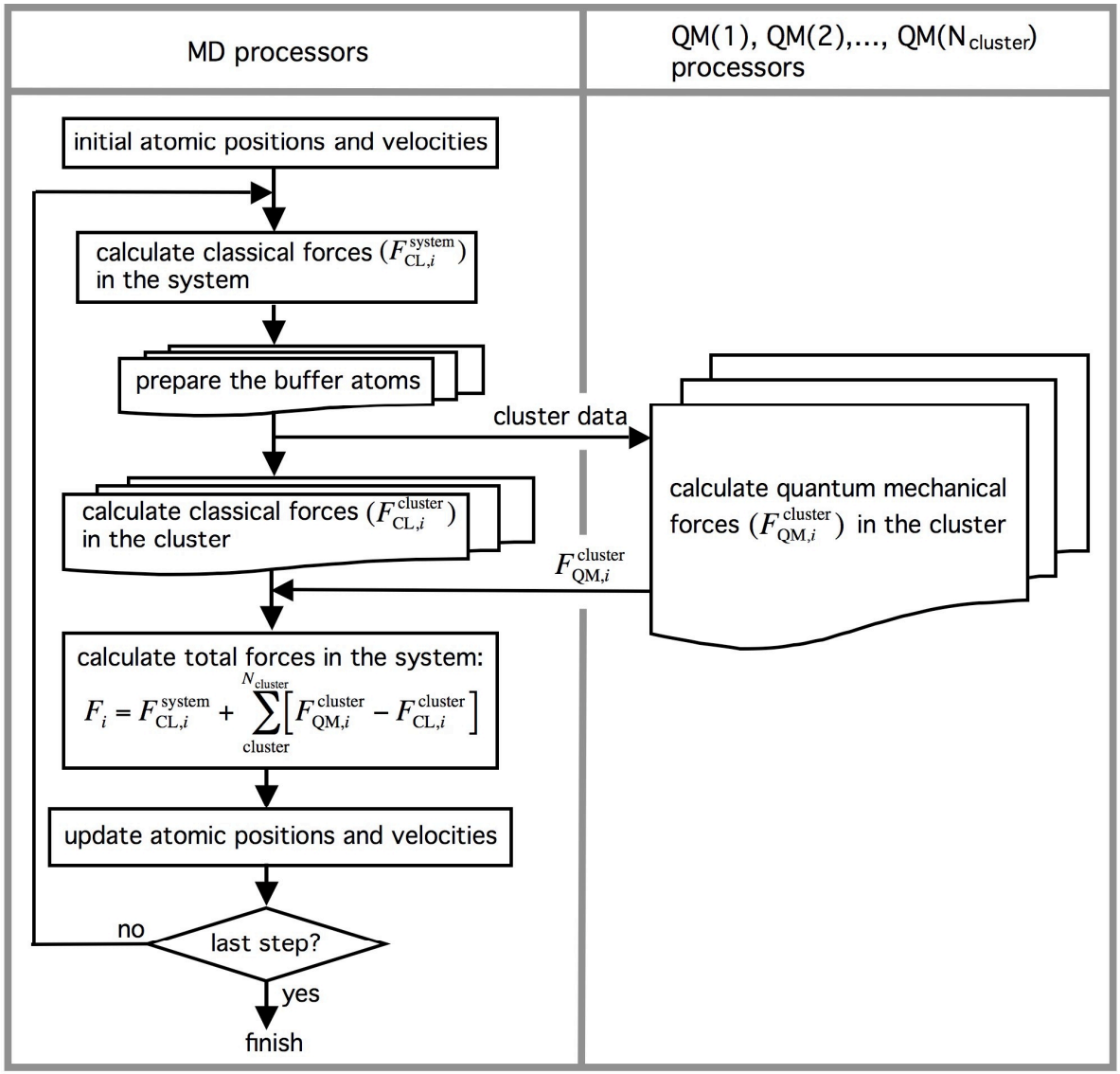






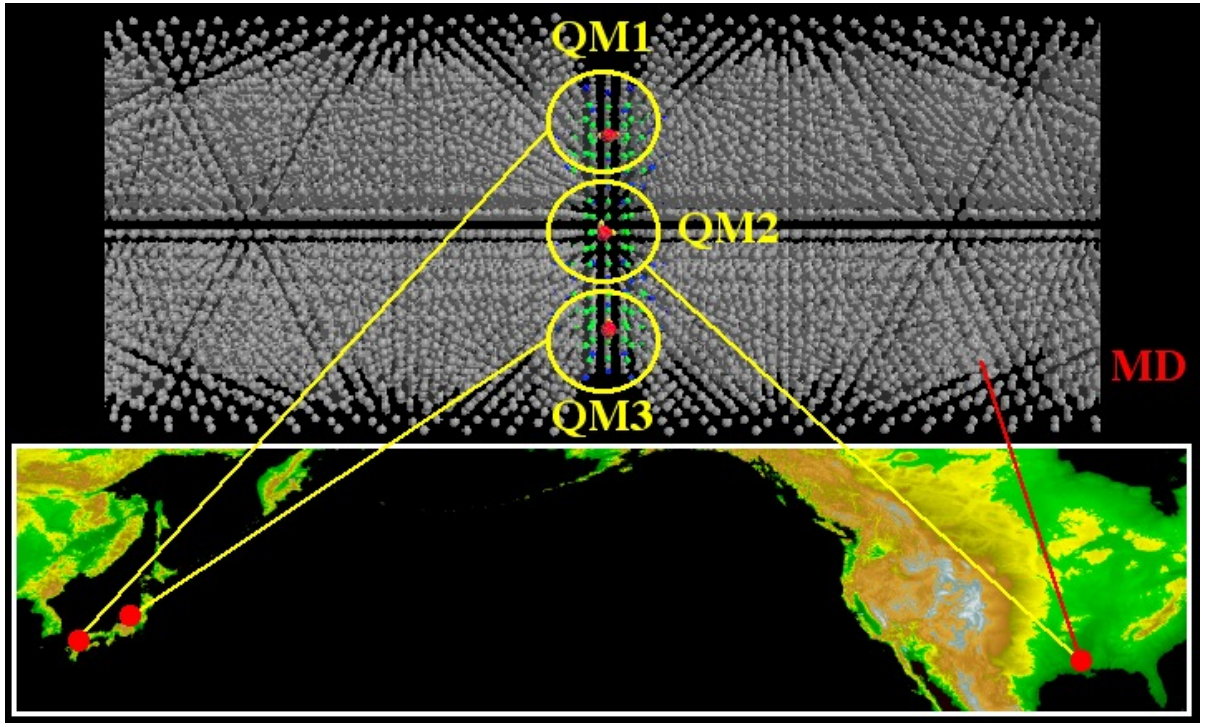
Grid Implementation

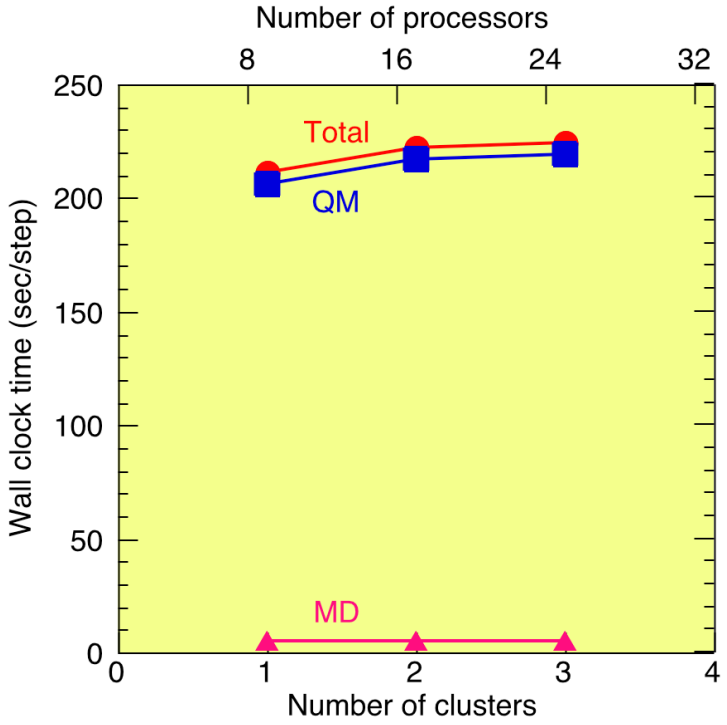
- **Task decomposition (MPI Communicator) + spatial decomposition**
- **Computation/communication overlapping to hide latency**
- **MPICH-G2 (**www3.niu.edu/mpi**)/Globus (**www.globus.org**)**



Global Collaborative Simulation

Hybrid MD/QM simulation on a Grid of distributed PC clusters in the US & Japan





Japan: Yamaguchi — 65 P4 2.0GHz
Hiroshima, Okayama, Niigata — 3×24 P4 1.8GHz
US: Louisiana — 17 Athlon XP 1900+

MD — 91,256 atoms
QM (DFT) — 76*n* atoms on *n* clusters

- **Scaled speedup, $P = 1$ (for MD) + $8n$ (for QM)**
- **Weak-scaling efficiency = 0.94 on 25 processors over 3 PC clusters**

H. Kikuchi, *et al.*, *IEEE/ACM SC02*

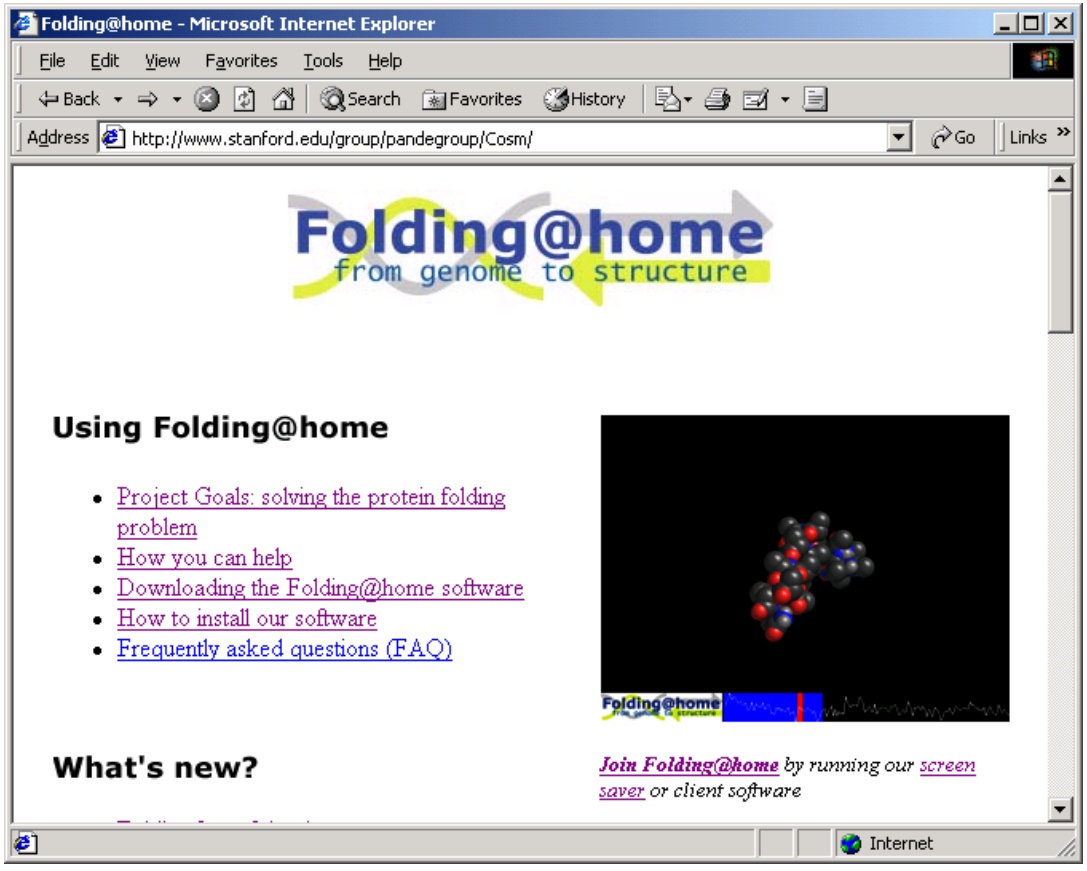
Outline

1. Grid programming
   - > Metacomputing—multiscale MD/quantum-mechanical (QM) simulations: Grid-enabled MPI (MPI–G2)
   - > Task farm: Grid remote procedure call (Ninf–G)
   - > Sustainable & adaptive Grid supercomputing
2. Grid software
   - > Globus toolkit
   - > Open Grid Services Architecture (OGSA)

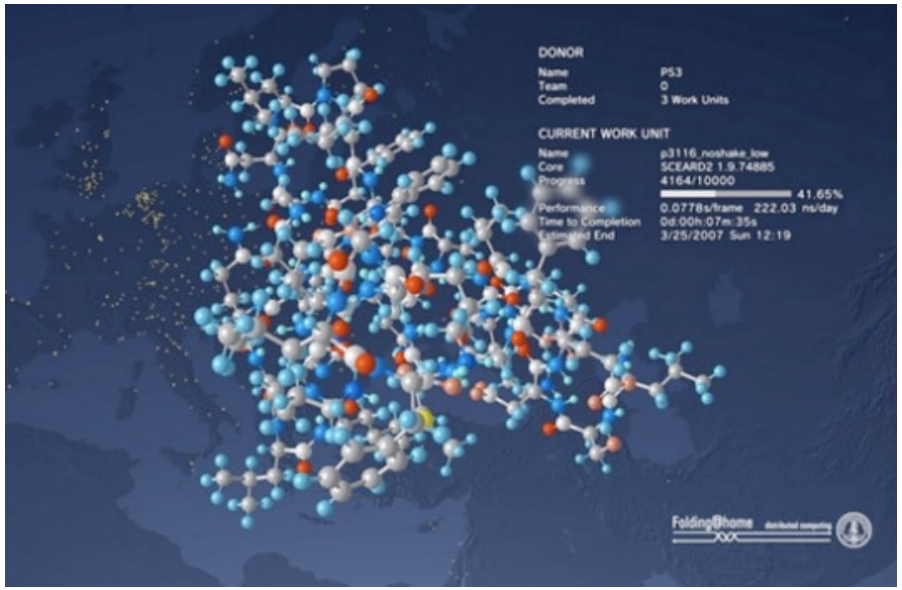
Task Farm Applications

Number CPUs	Number Active CPUs	Number Users	Number Teams	Last Update
423995	90438	210257	25971	2003-05-05 20:04:04

OS type	Active	Total
Windows	86473	369859
Mac OS X	2653	24129
Linux	1294	29931
Other	0	13
Total	90420	423932



Screen saver
(*cf*. OpenGL idle event handler)

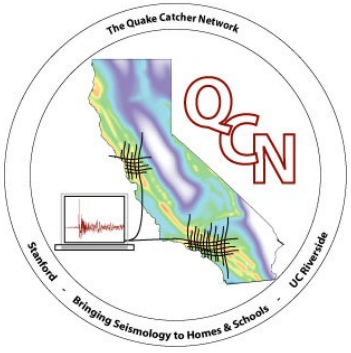


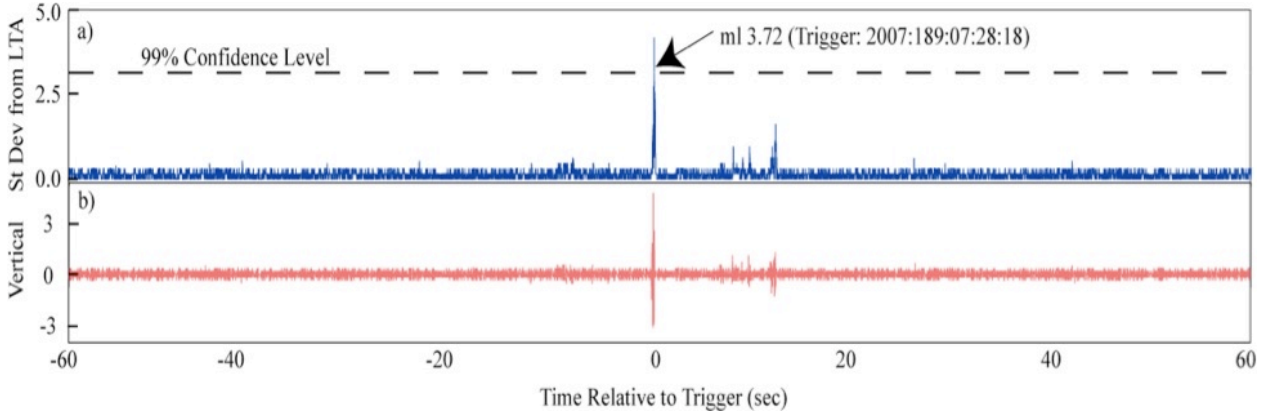
http://folding.stanford.edu

Quake-Catcher Network

- **Network of accelerometer-equipped laptops/desktops for early earthquake warning & research**
- **Clustering accelerometer time series data to detect earthquakes**

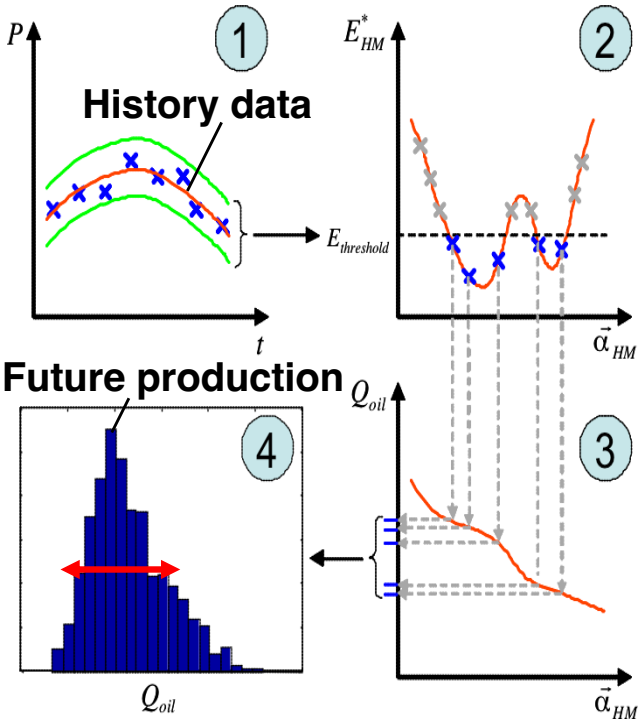
http://qcn.stanford.edu



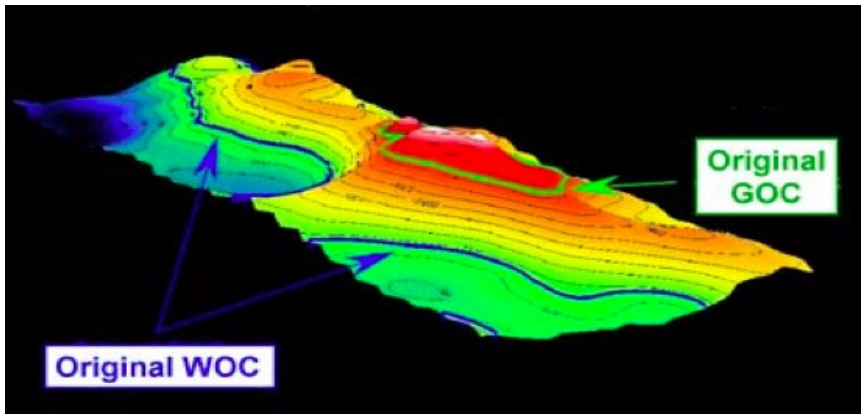


Parallel History Matching

- **Inverse problem: Calibration of reservoir simulation models to the observed production data**

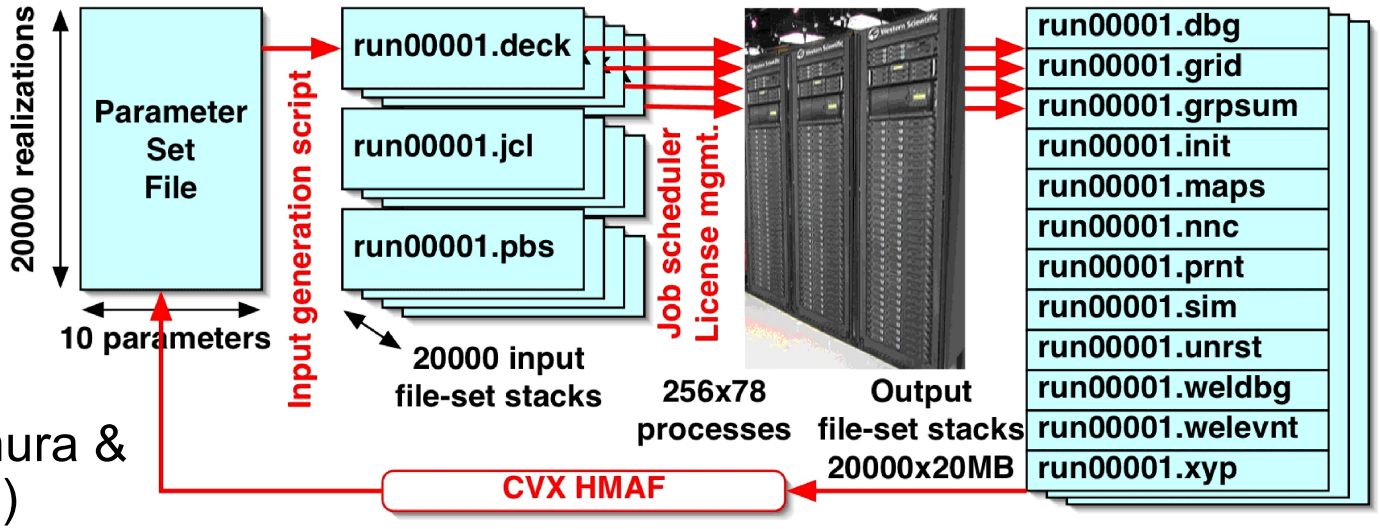


HMAF:
Landa & Guyaguler, *SPE* 84465 ('03)



- Real field—offshore Africa
- 32 wells
- 30 years production history
- 30 years production forecast

- **CVX History Match & Associated Forecast (HMAF) framework: History matching & the assessment of uncertainties associated with flow prediction**

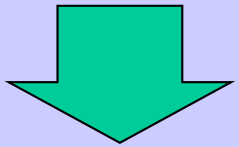


Parallel HMAF:
Landa, Kalia, Nakano, Nomura & Vashishta, *IPTC* 10751 ('05)

Grid Remote Procedure Call (RPC)

- **Simple RPC API (application program interface)**
- **Existing libraries & applications into Grid applications**
- **IDL (interface definition language) embodying call information, with minimal client-side management**

```
double A[n][n],B[n][n],C[n][n];      /* Data Declaration */
dmmul(n,A,B,C);                      /* Call local function */
```



```
grpc_function_handle_default(&hdl, "dmmul");
grpc_call(hdl,n,A,B,C);   /* Call server side routine */
```

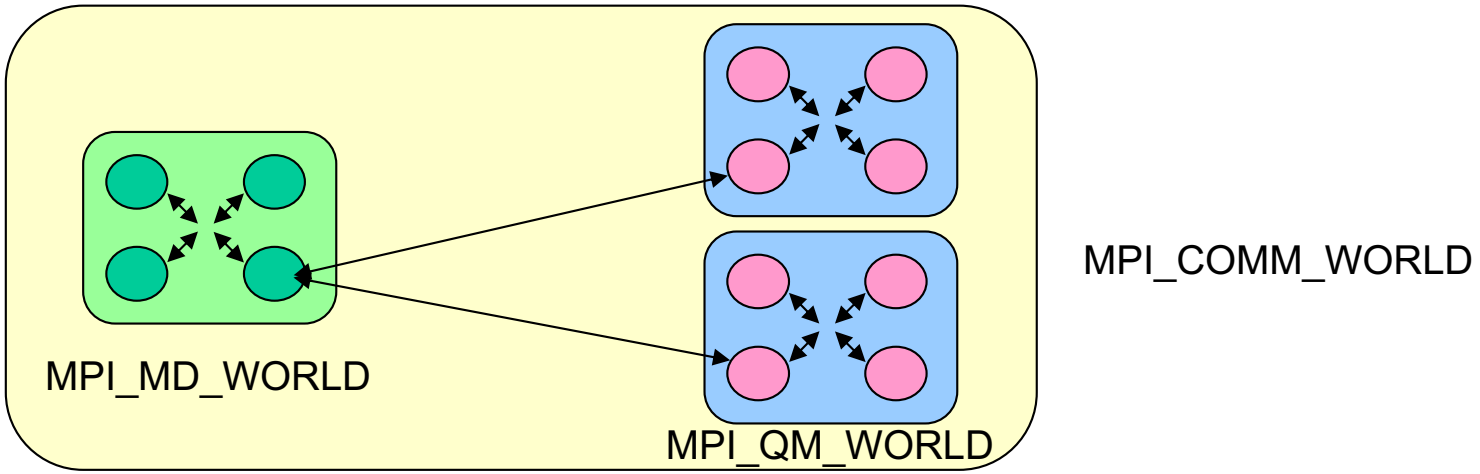
- **Ninf–G Grid RPC system**
  http://ninf.apgrid.org

Outline

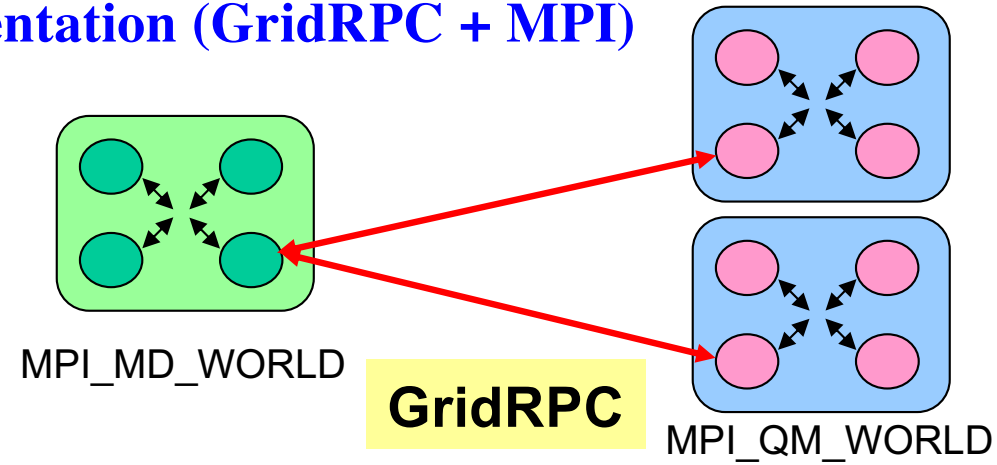
1. Grid programming
   - > Metacomputing—multiscale MD/quantum-mechanical (QM) simulations: Grid-enabled MPI (MPI–G2)
   - > Task farm: Grid remote procedure call (Ninf–G)
   - > Sustainable & adaptive Grid supercomputing
2. Grid software
   - > Globus toolkit
   - > Open Grid Services Architecture (OGSA)

Combined GridRPC+MPI MD/QM

- **Original implementation (MPICH-G2)**

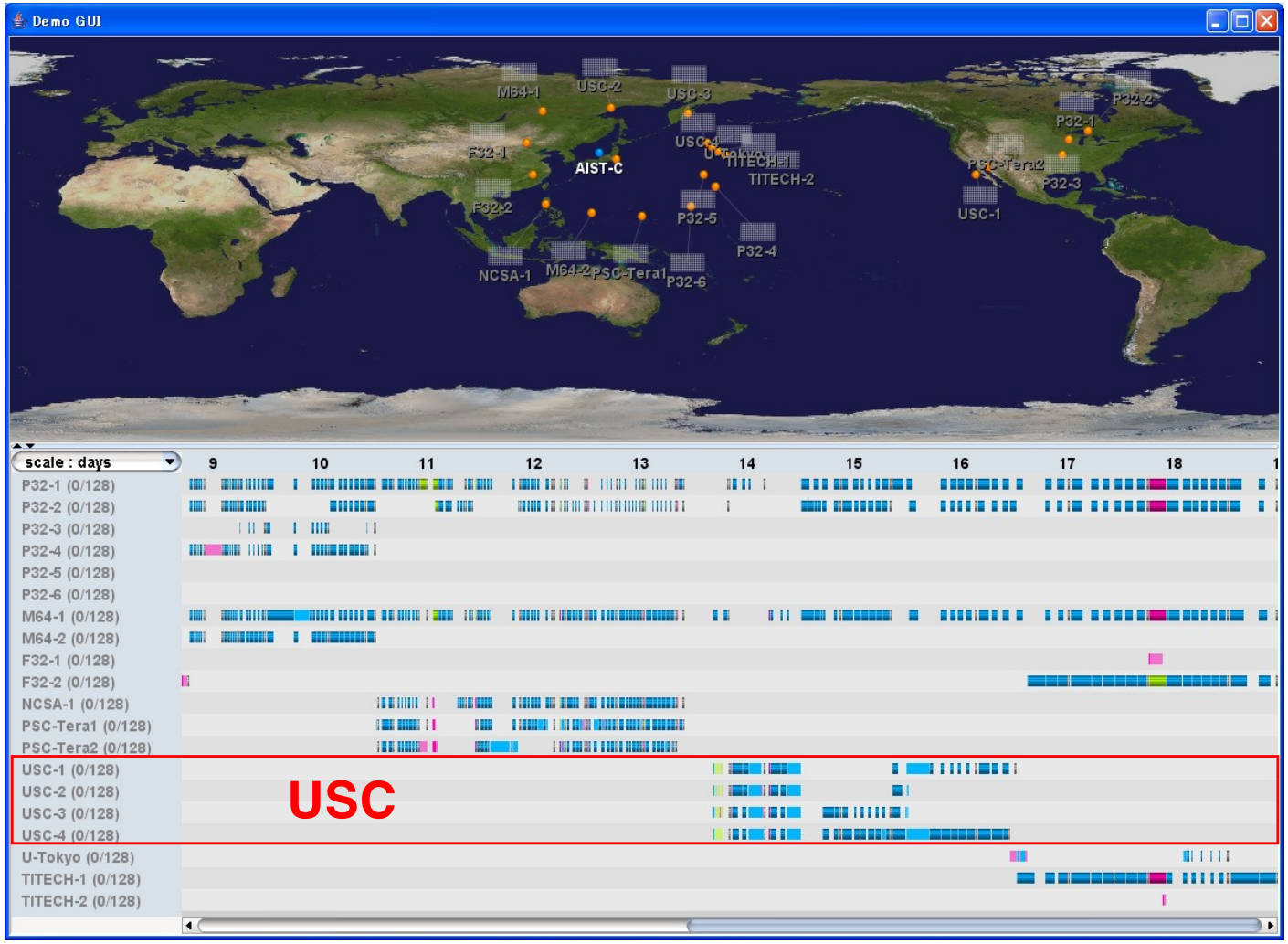


- **Hybrid implementation (GridRPC + MPI)**



- **Flexibility: Dynamically add/subtract, allocate & migrate resources**
- **Fault tolerance: Automatically detect & recover from explicit (OS down or disconnected networks) & implicit (job stuck in a queue) faults**
- **Scalability: Manage 1000's of computing resources efficiently**

Global Grid QM/MD



H. Takemiya, Y. Tanaka, S. Sekiguchi (AIST)

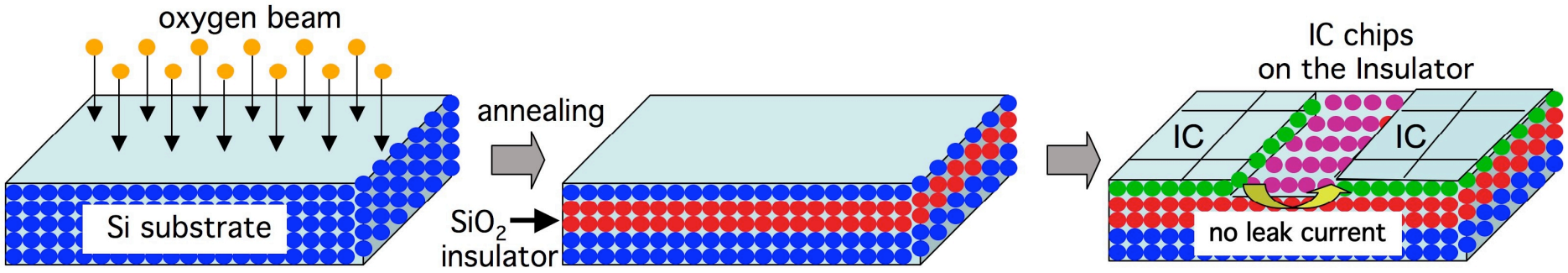
S. Ogata (NIT)

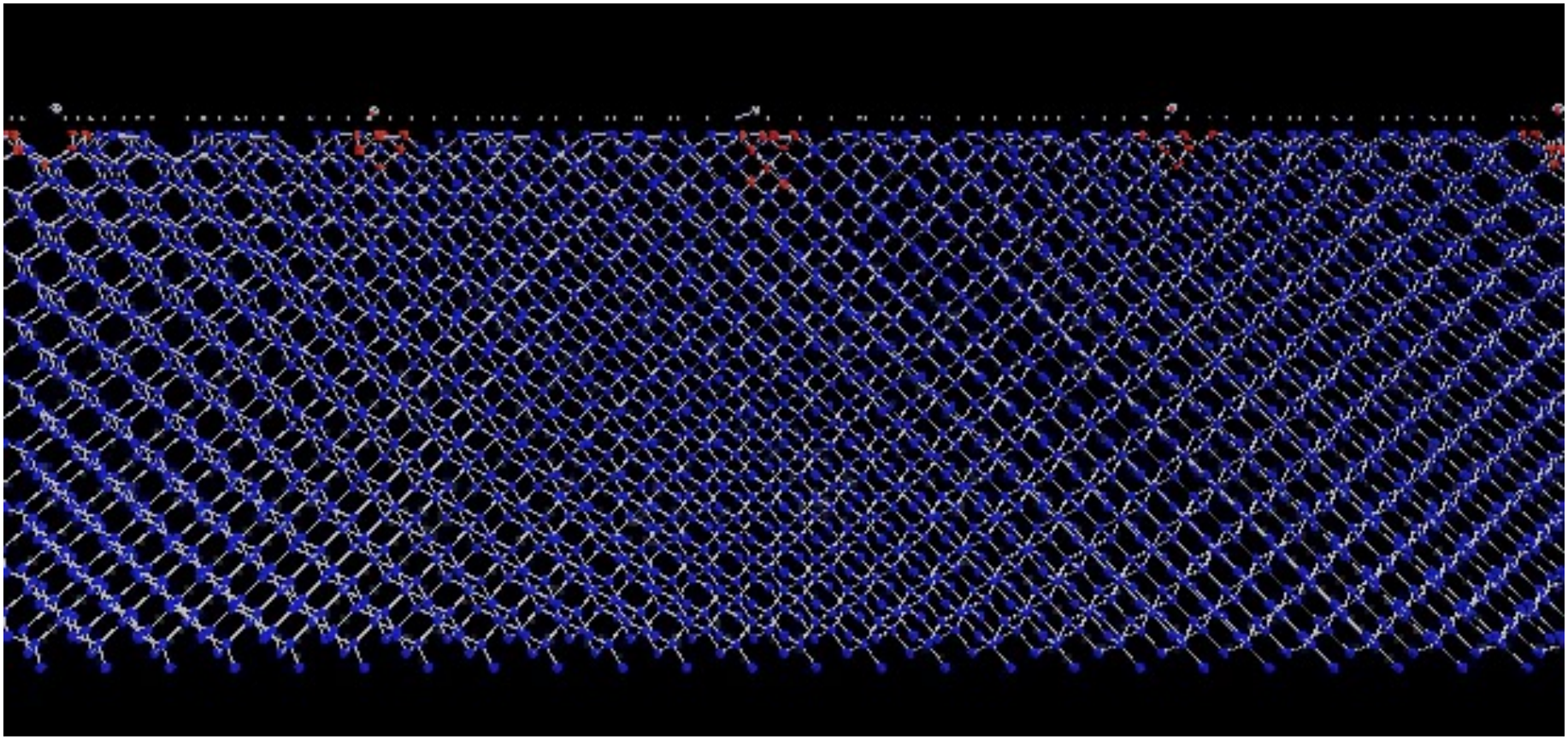
R.K. Kalia, A. Nakano, P. Vashishta (USC)

- **Hybrid GridRPC**(ninf.apgrid.org)**+MPI**(www.mcs.anl.gov/mpi) **Grid computing**
- **153,600 cpu-hrs metacomputing at 6 sites in the US (USC, PSC—Pittsburgh, NCSA—Illinois) & Japan (AIST, U Tokyo, TITech)**

H. Takemiya *et al.*, *Proc. IEEE/ACM SC06*; Y. Song *et al.*, *Int'l J. Comput. Sci.* ('09); *CCGrid09*

SIMOX (Separation by Implantation by Oxygen)

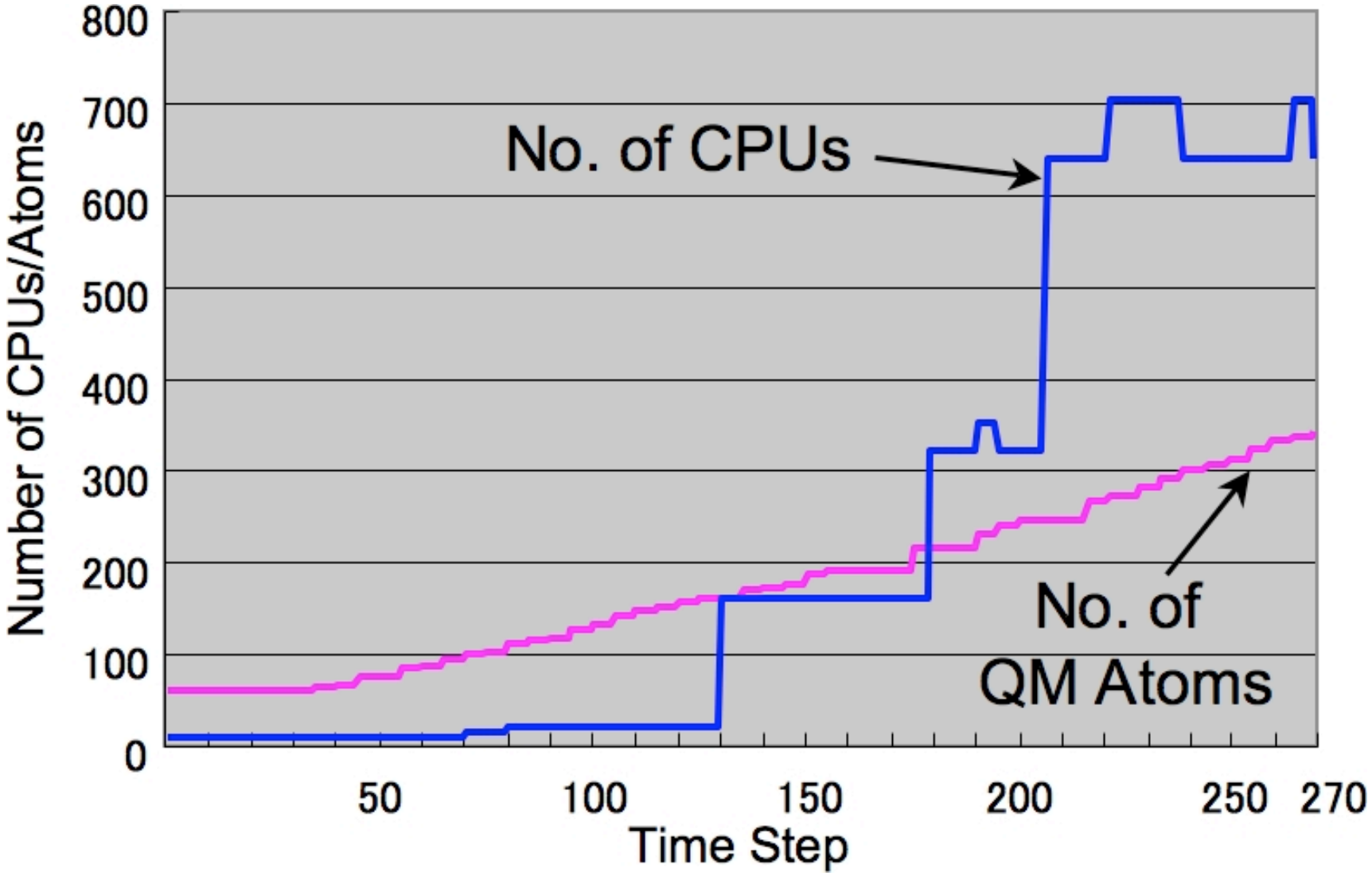




Red: quantum mechanically treated atoms ~ $O(N^3)$

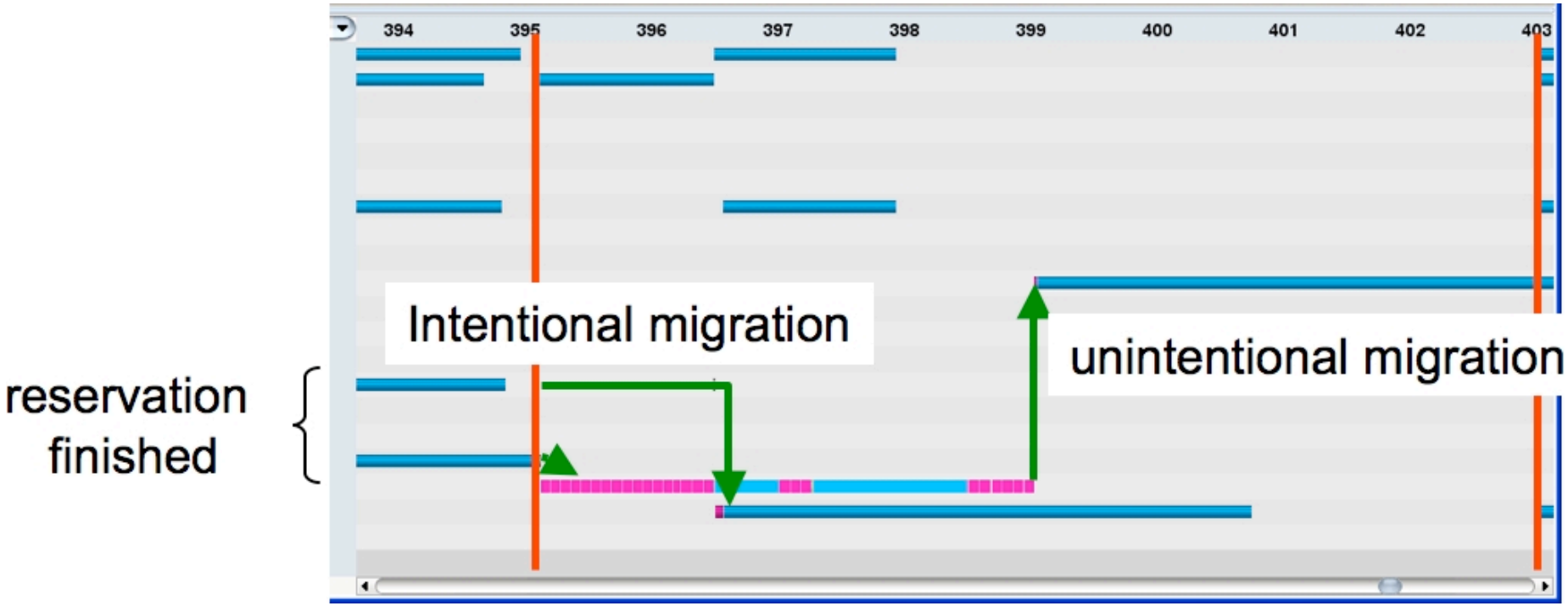
Flexibility: Adaptive MD/QM

- **Flexibility: Automated increase of the number of QM atoms on demand to maintain accuracy & associated dynamic re-allocation of CPUs**



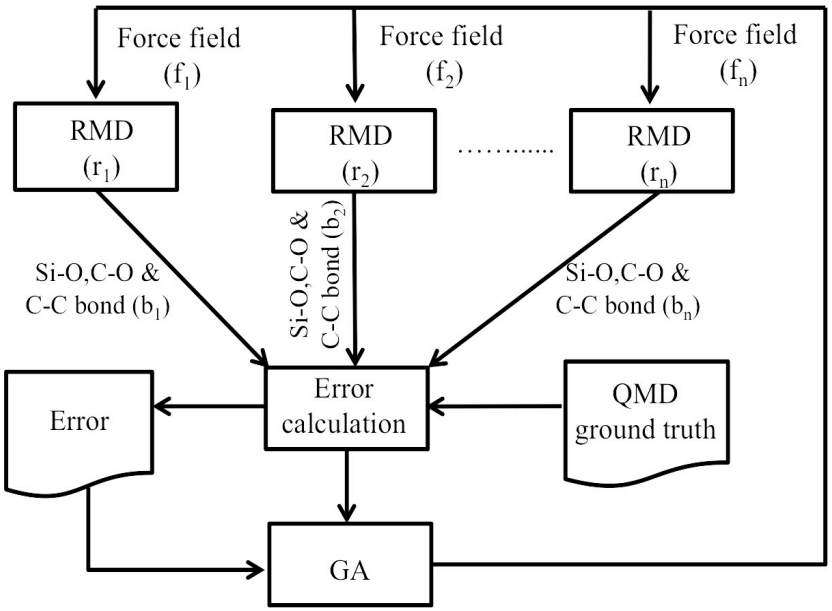
Fault Tolerance

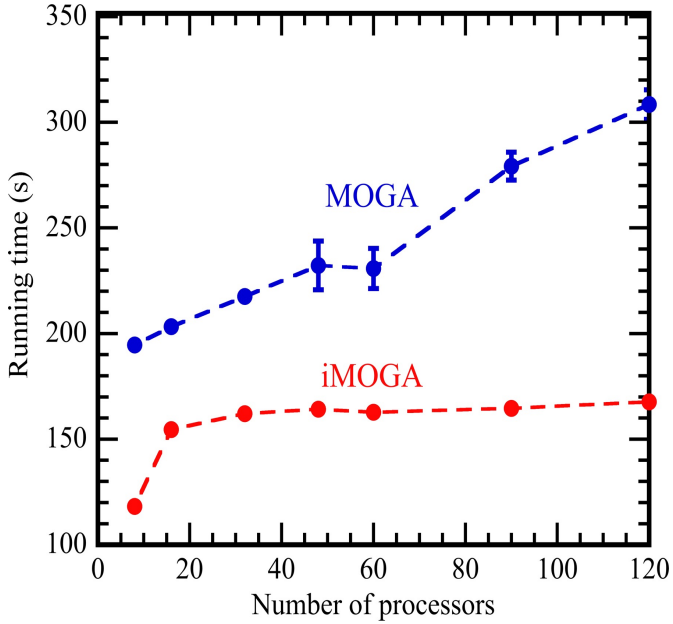
- **Automated migration in response to unexpected faults**



In Situ Simulation & Learning Workflow

- **Train reactive force-field parameters by dynamically fitting a large number of reactive molecular dynamics (RMD) trajectories to quantum molecular dynamics (QMD) trajectories on-the-fly**
- **Pareto-frontal uncertainty quantification (UQ): Pareto optimal front in multiobjective genetic algorithm (MOGA) provides an ensemble of force fields to enable UQ**
- **File-based workflow was not scalable for large GA population size**
- ***In situ* MOGA (iMOGA): File I/O bottleneck replaced by piping within each computing node & TCP/IP socket communication across nodes for scalability**





A. Mishra *et al.*, *npj Comput. Mater.* **4**, 42 ('18)

Outline

1. Grid programming
   - Metacomputing—multiscale MD/quantum-mechanical (QM) simulations: Grid-enabled MPI (MPI–G2)
   - Task farm: Grid remote procedure call (Ninf–G)
   - Sustainable & adaptive Grid supercomputing
2. Grid software
   - Globus toolkit
   - Open Grid Services Architecture (OGSA)

Globus Toolkit

- **Globus Toolkit version 2 (GT2): Open source, de facto standard of Grid computing middleware to construct interoperable Grid applications ('97)**
  - **Define & implement protocols, application program interfaces (APIs) & services**
  - **Provide solutions to authentication, resource discovery & resource access**
- **Globus Toolkit version 3 (GT3): OGSA-compliant standard ('02)**

http://www.globus.org

Outline

1. Grid programming
   - > Metacomputing—multiscale MD/quantum-mechanical (QM) simulations: Grid-enabled MPI (MPI–G2)
   - > Task farm: Grid remote procedure call (Ninf–G)
   - > Sustainable & adaptive Grid supercomputing
2. Grid software
   - > Globus toolkit
   - > Open Grid Services Architecture (OGSA)

Open Grid Services Architecture (OGSA)

- **OGSA = Definition of a service-oriented infrastructure**
- **Service: A network-enabled entity with a well-defined interface that provides some capability**

1. **Align Grid computing with industrial initiatives in service-oriented architecture & Web services**
2. **Provide a framework within which to define interoperable & portable services**
3. **Define a core set of standard interfaces & behaviors**
4. **Implemented in the OGSA-based Globus Toolkit 3**
   http://www.globus.org