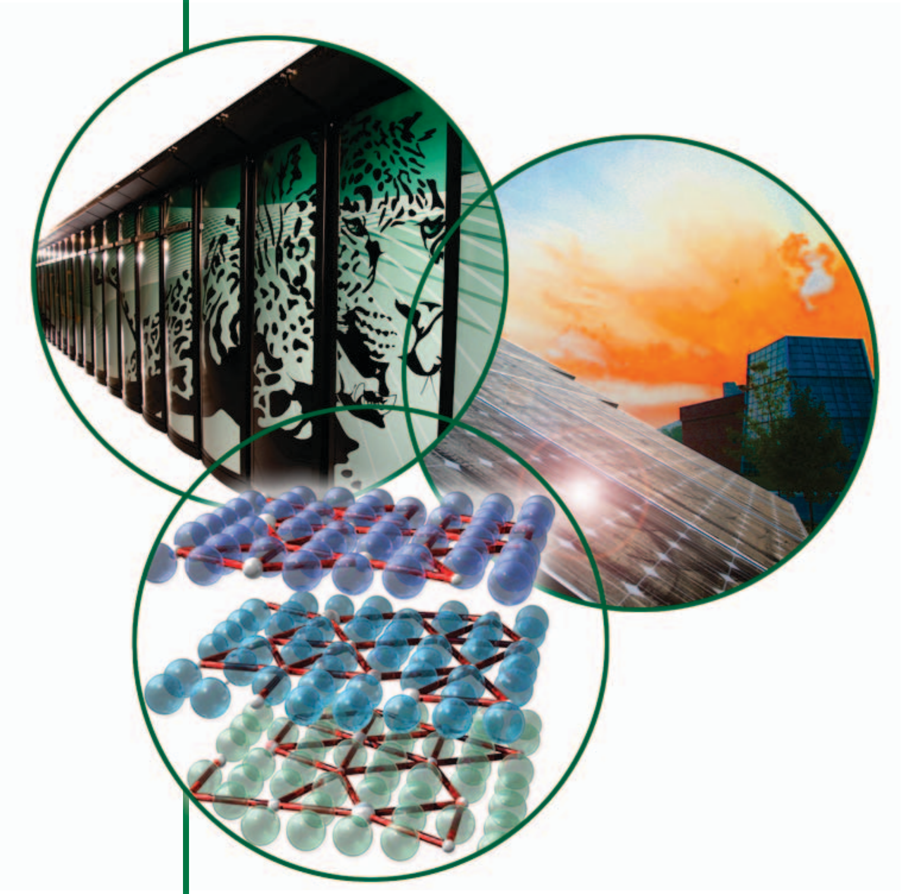
Toward Performance Prediction of Tree-Based Overlay Networks on the Cray XT

Philip C. Roth

Computer Science and Mathematics Division
Oak Ridge National Laboratory



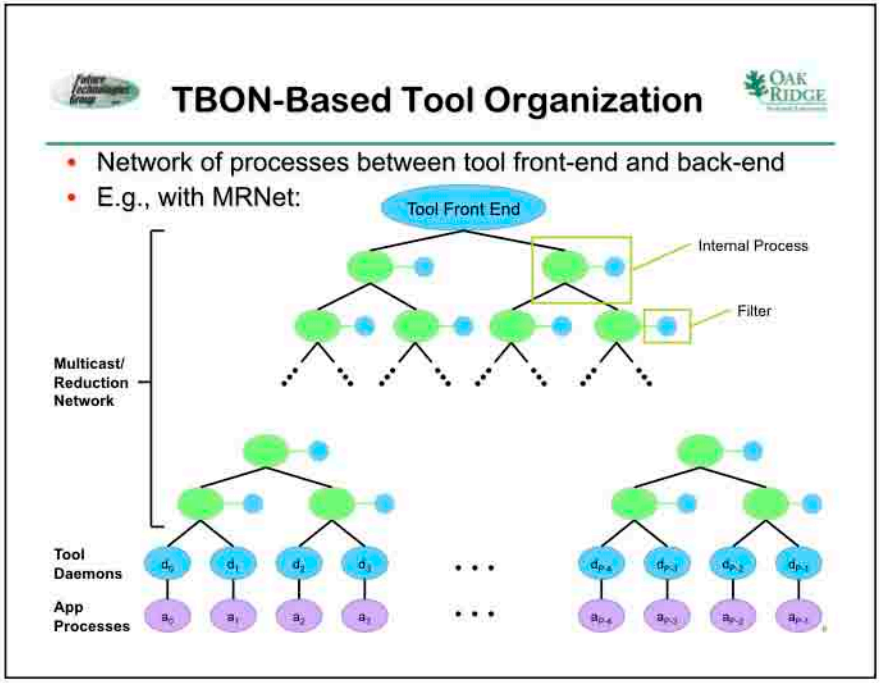
U.S. DEPARTMENT OF ENERGY

Future Technologies Group

OAK RIDGE NATIONAL LABORATORY
MANAGED BY UT-BATTELLE FOR THE DEPARTMENT OF ENERGY

TBONs and MRNet

- A Tree-Based Overlay Network (TBON) like MRNet provides scalable infrastructure for tools and applications
- MRNet's process topology and placement support is extremely flexible (on most platforms)
  - Any tree topology
  - Internal processes on same nodes as application processes, or on distinct nodes

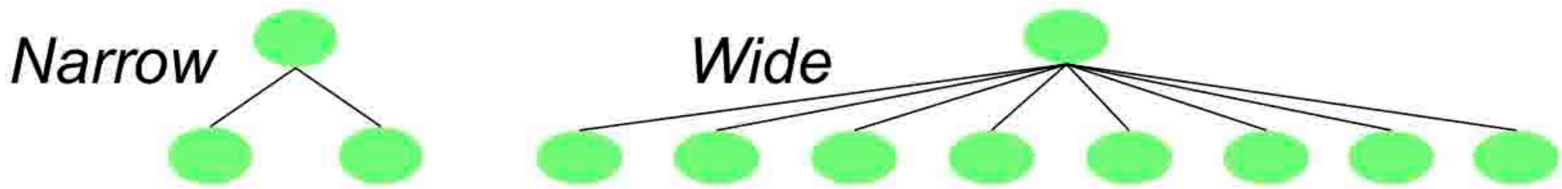


TBON-Based Tool Organization

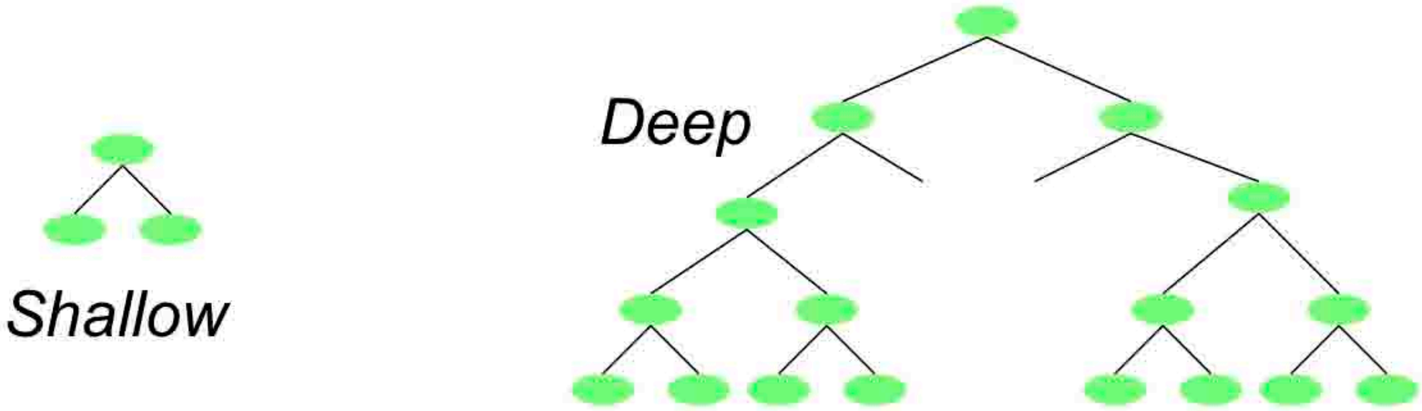
- Network of processes between tool front-end and back-end
- E.g., with MRNet:

TBON Topology Flexibility

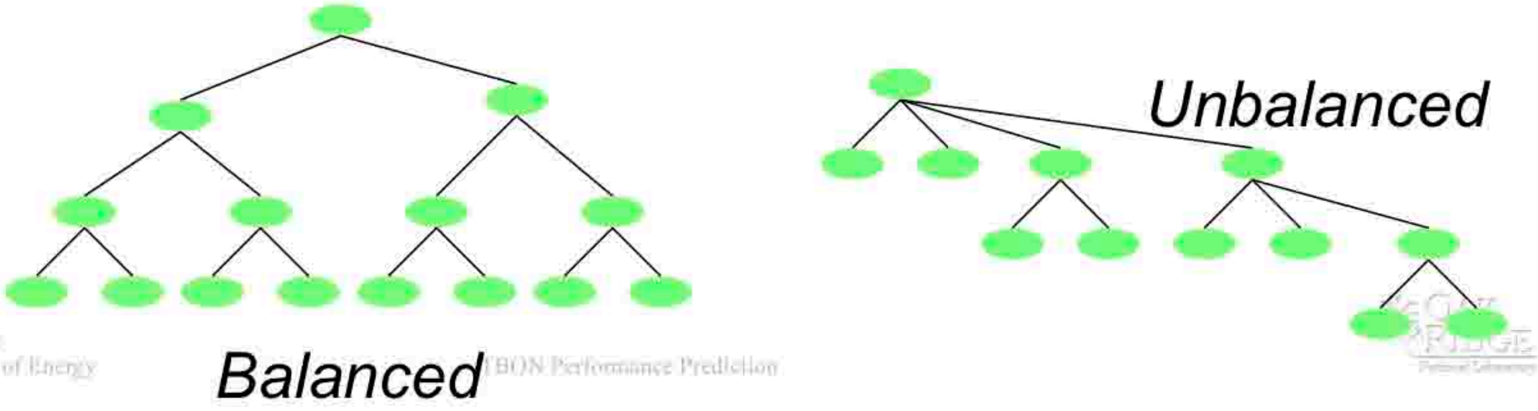
Fanout:



Depth:



Balance:



The Problem With Flexibility

- Flexibility leads to questions identifying “best” process topology and placement
- Interaction of several factors determine “best”
  - Performance (tool and application)
  - System hardware and software
  - Purpose
  - Even economics (e.g., can I afford to request “extra” nodes for MRNet processes given my allocation budget?)
- Decision process often not rigorous – using “rule of thumb”

TBON Performance Prediction

- Goal: Given a node allocation on a leadership class system, to be able to identify “best” MRNet process placement and topoogy

- Several constraints:
  - Tool multicast and reduction requirements
  - Behavior of application under study
  - Other activity on the system
  - System software and hardware

Target Platform

- Cray XT is target platform
  - Jaguar XT4 and XT5 systems at Oak Ridge National Laboratory (ORNL)
  - Hopper XT5 at NERSC
  - Kraken XT5 at ORNL
- Opteron-based nodes arranged in 3D mesh with possibility of torus links

Image courtesy of the National Center of Computational Sciences, Oak Ridge National Laboratory

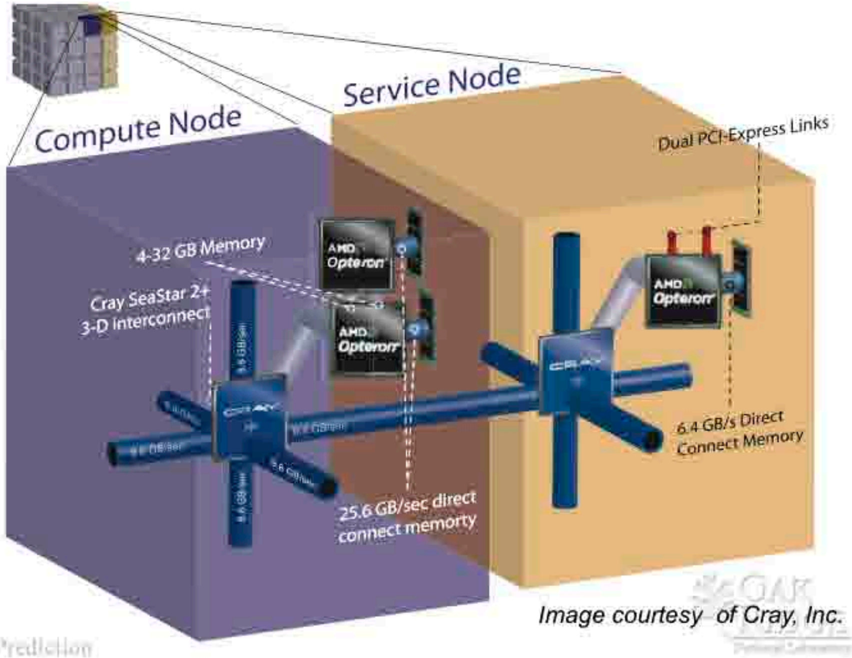
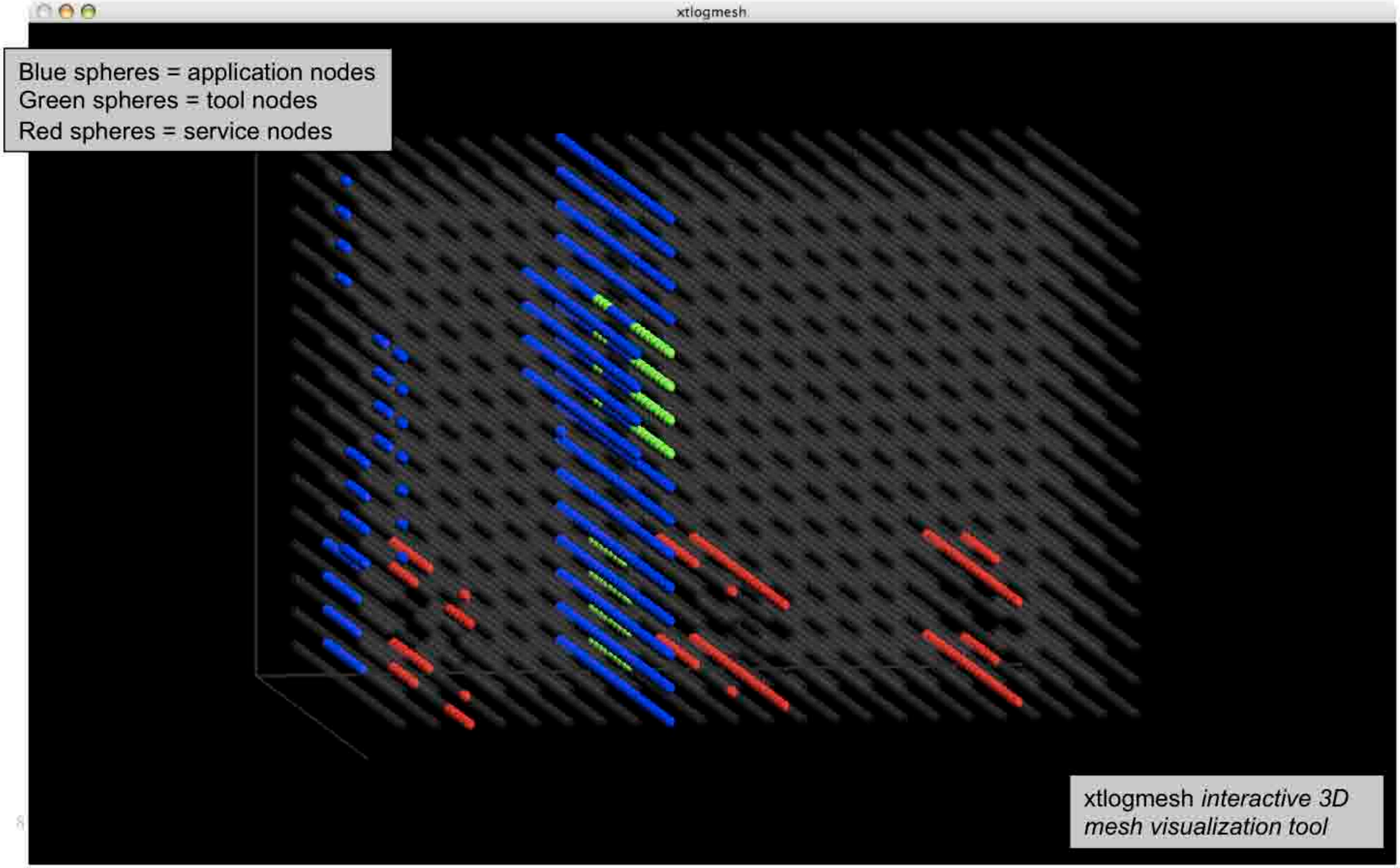


Image courtesy of Cray, Inc.

Cray XT TBON Process Placement Tests

- Goal: understand Cray XT allocation characteristics & their impact on MRNet-based tool process placement
- Used simple MPI/Portals program to collect node number and position within the XT mesh
  - Earlier generation ORNL Jaguar with dual-core Opterons
- Batch job launched two independent instances of the program:
  - 512 application nodes (1024 processes)
  - 72 tool nodes (enough for balanced 8-way TBON topology assuming front-end is on batch script service node)

Jaguar Placement Trial Example

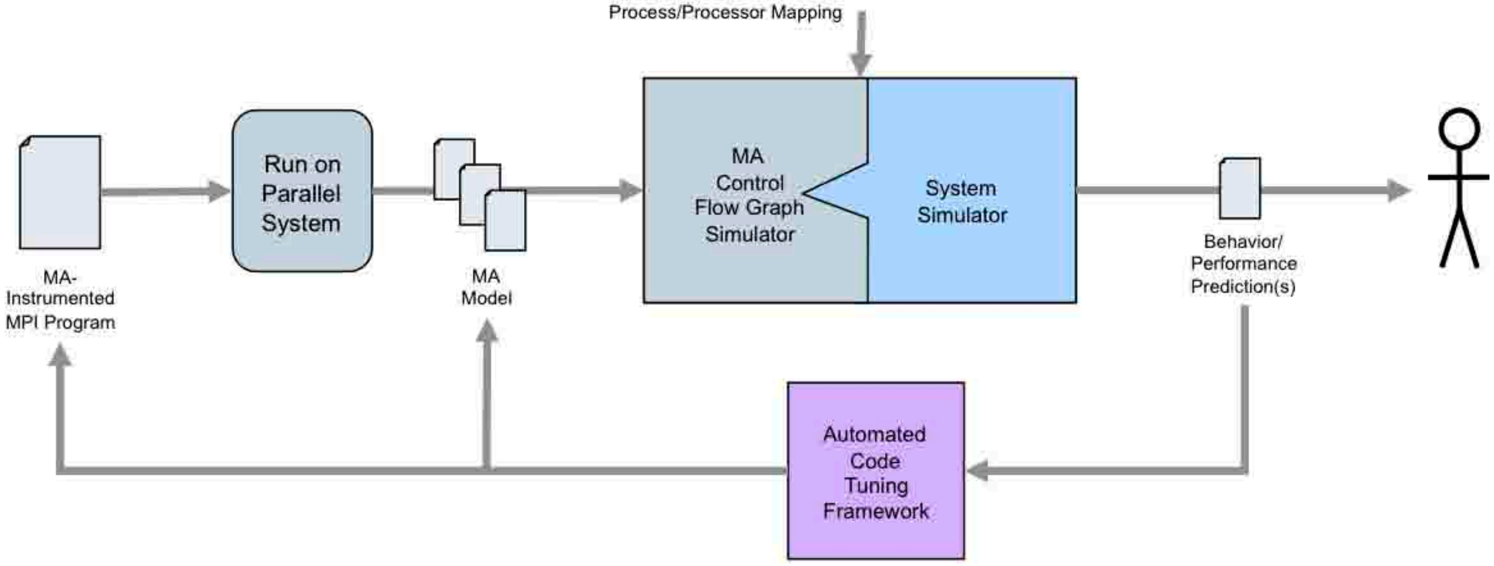


Blue spheres = application nodes
Green spheres = tool nodes
Red spheres = service nodes

xtlogmesh *interactive 3D mesh visualization tool*

Our Approach

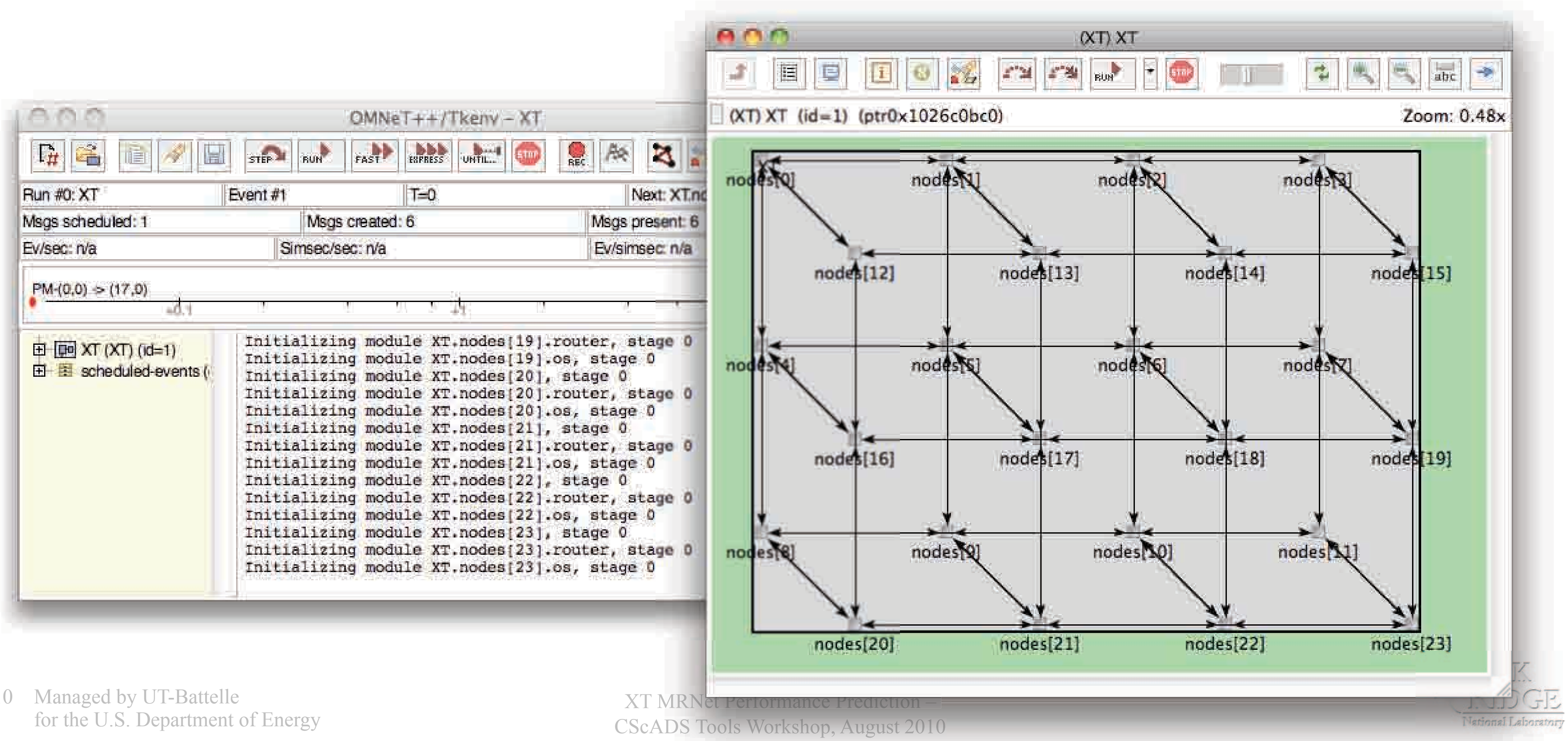
- **Discrete event simulation of XT system nodes running application and MRNet processes**
- **Component of MAST framework: Modeling Assertions, Simulation, and Tuning**


Process/Processor Mapping
MA-Instrumented MPI Program
Run on Parallel System
MA Model
MA Control Flow Graph Simulator
System Simulator
Behavior/ Performance Prediction(s)
Automated Code Tuning Framework

MA-Instrumented MPI Program

System Model

- Node modules connected in 3D torus
- Implemented using OMNeT++ (http://www.omnetpp.org)



Run #0: XT | Event #1 | T=0 | Next: XT.nc

Msgs scheduled: 1 | Msgs created: 6 | Msgs present: 6

Ev/sec: n/a | Simsec/sec: n/a | Ev/simsec: n/a

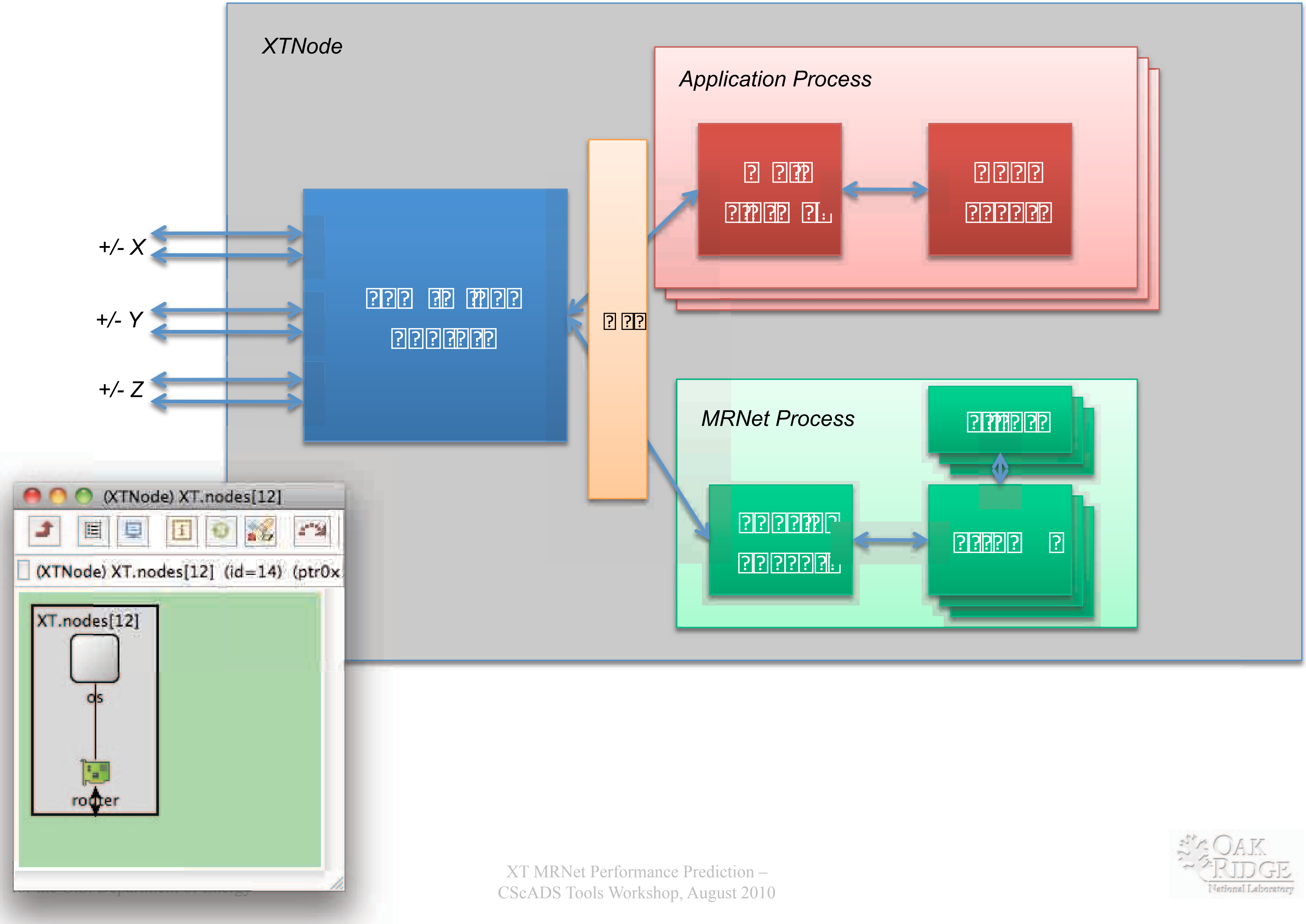
PM-(0,0) -> (17,0)

XT (XT) (id=1)
scheduled-events (

```
Initializing module XT.nodes[19].router, stage 0
Initializing module XT.nodes[19].os, stage 0
Initializing module XT.nodes[20], stage 0
Initializing module XT.nodes[20].router, stage 0
Initializing module XT.nodes[20].os, stage 0
Initializing module XT.nodes[21], stage 0
Initializing module XT.nodes[21].router, stage 0
Initializing module XT.nodes[21].os, stage 0
Initializing module XT.nodes[22], stage 0
Initializing module XT.nodes[22].router, stage 0
Initializing module XT.nodes[22].os, stage 0
Initializing module XT.nodes[23], stage 0
Initializing module XT.nodes[23].router, stage 0
Initializing module XT.nodes[23].os, stage 0
```

(XT) XT (id=1) (ptr0x1026c0bc0) Zoom: 0.48x

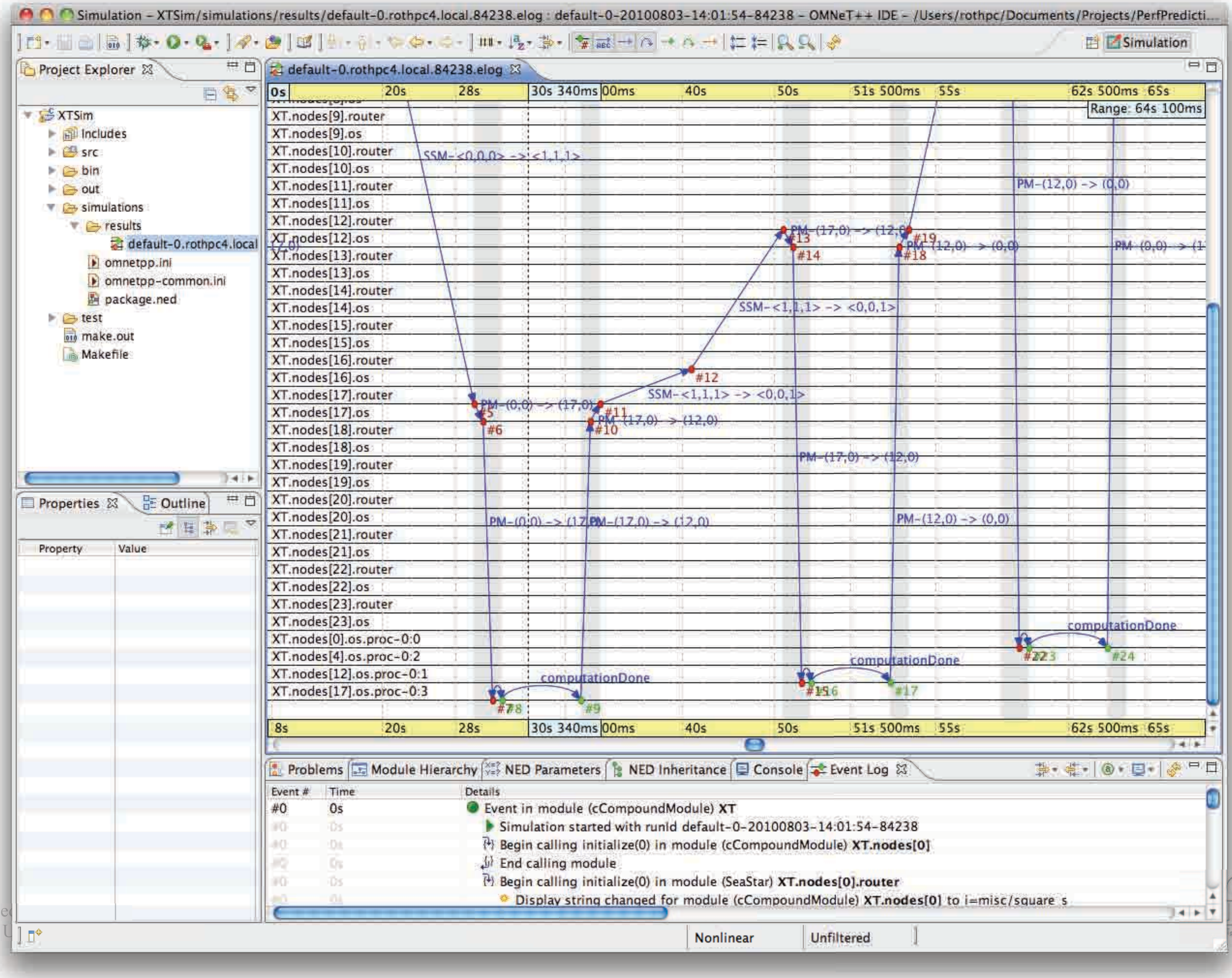
Node Model



(XTNode) XT.nodes[12]

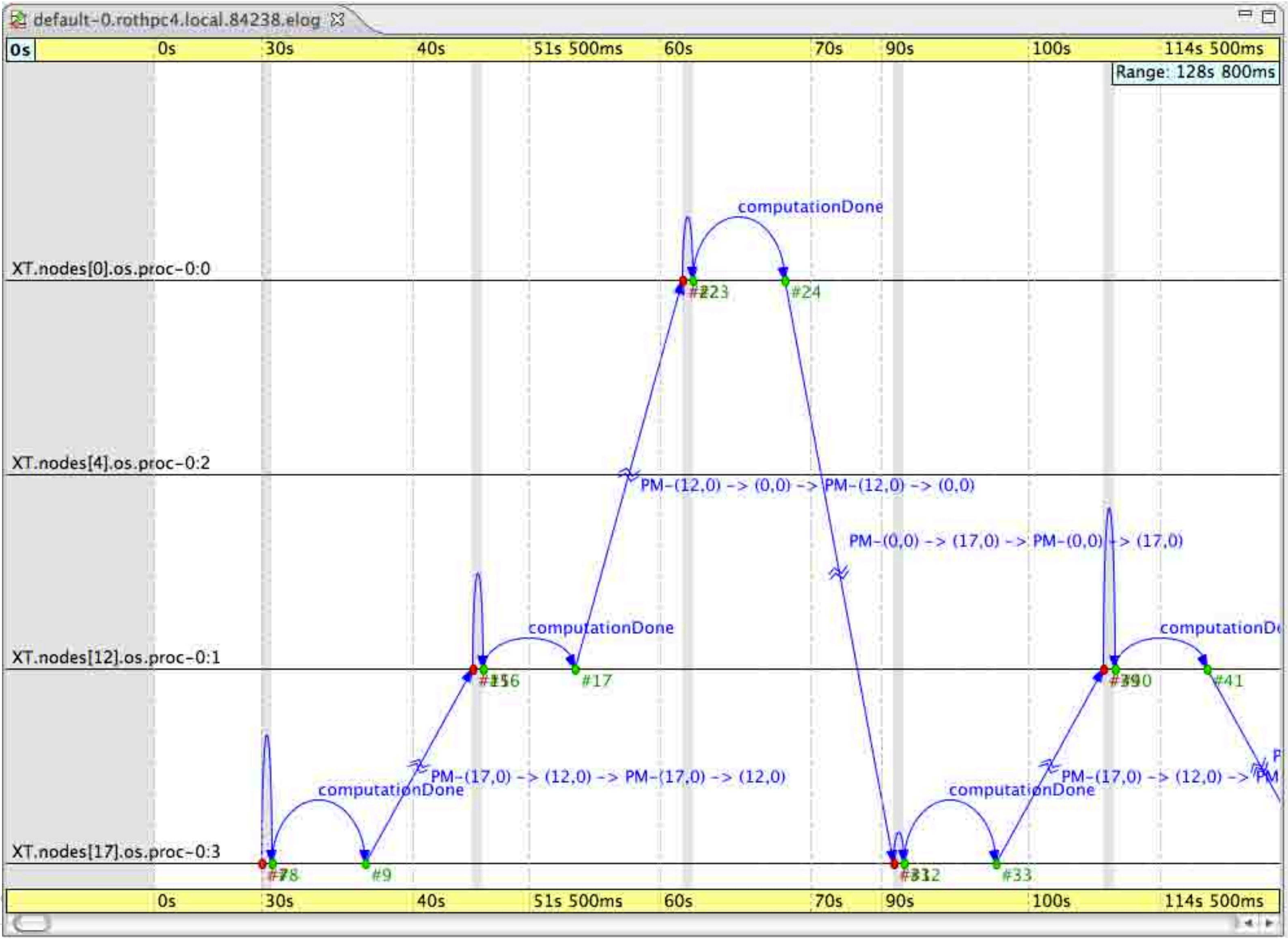
(XTNode) XT.nodes[12] (id=14) (ptr0x

Simulated MPI Hot Potato Activity



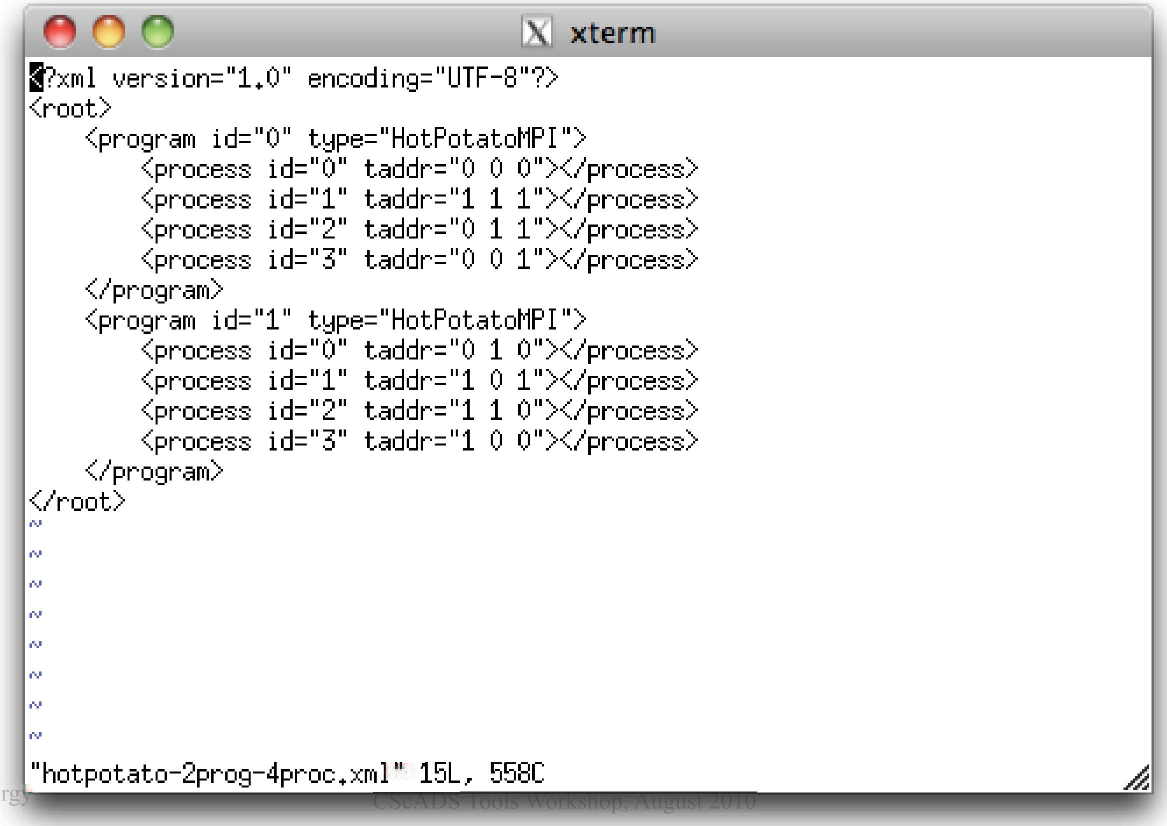
```
Event #   Time   Details
#0        0s     Event in module (cCompoundModule) XT
#0        0s       Simulation started with runId default-0-20100803-14:01:54-84238
#0        0s       Begin calling initialize(0) in module (cCompoundModule) XT.nodes[0]
#0        0s       End calling module
#0        0s       Begin calling initialize(0) in module (SeaStar) XT.nodes[0].router
#0        0s         Display string changed for module (cCompoundModule) XT.nodes[0] to i=misc/square_s
```

Simulated MPI Hot Potato Activity, Filtered



Workload Specification

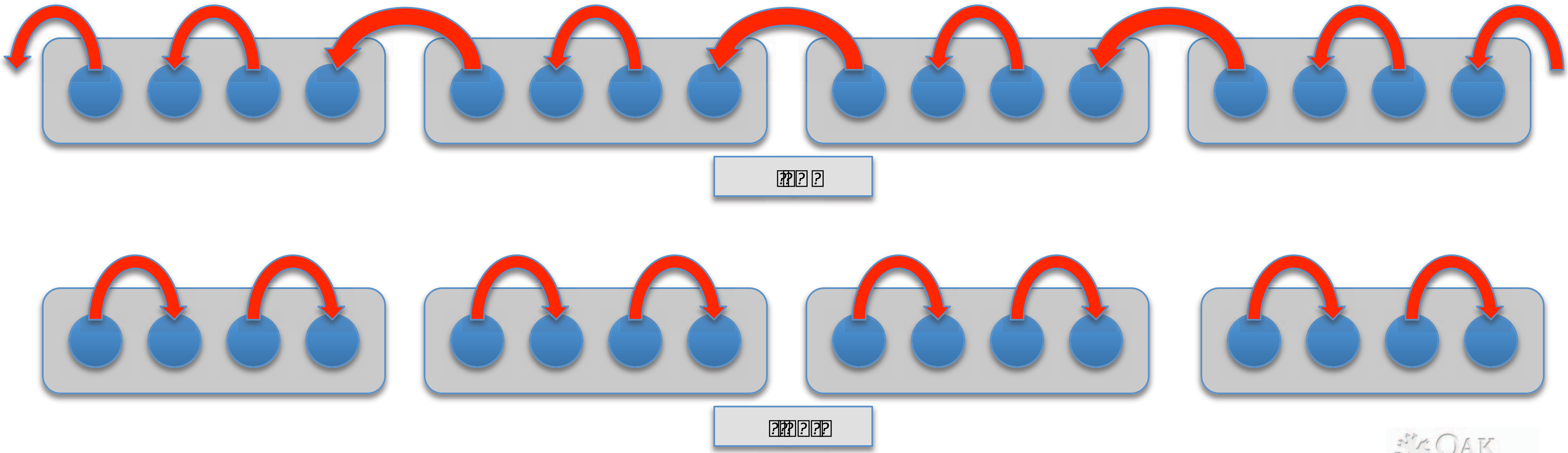
- XML file
- Multiple parallel programs per file, including type and associated attributes like “input”
- Mapping of processes to system nodes



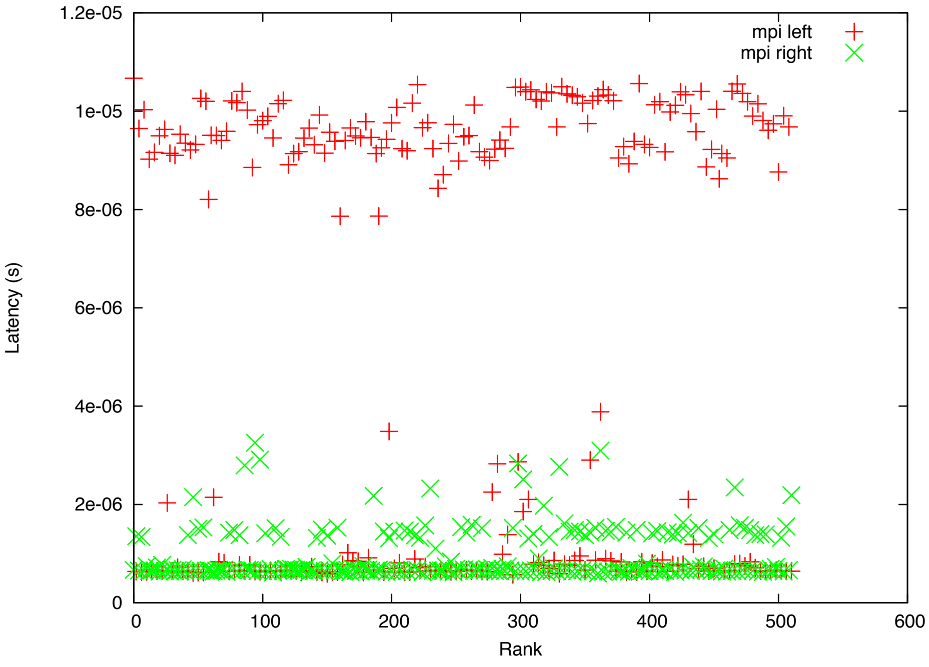
```
<?xml version="1.0" encoding="UTF-8"?>
<root>
    <program id="0" type="HotPotatoMPI">
        <process id="0" taddr="0 0 0"></process>
        <process id="1" taddr="1 1 1"></process>
        <process id="2" taddr="0 1 1"></process>
        <process id="3" taddr="0 0 1"></process>
    </program>
    <program id="1" type="HotPotatoMPI">
        <process id="0" taddr="0 1 0"></process>
        <process id="1" taddr="1 0 1"></process>
        <process id="2" taddr="1 1 0"></process>
        <process id="3" taddr="1 0 0"></process>
    </program>
</root>
~
~
~
~
~
~
~
~
"hotpotato-2prog-4proc.xml" 15L, 558C
```

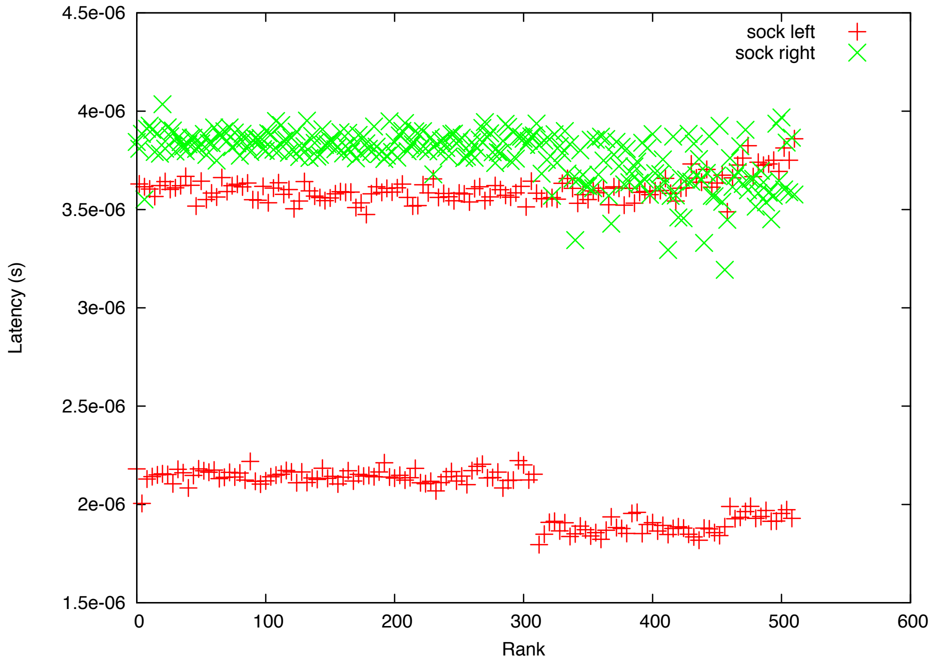
Model Parameterization

- Measuring process-to-process latency and bandwidth
  - MPI, Sockets
  - Fully populated nodes, one process per node
- Pairs of processes
  - Even ranks first pair left, then right

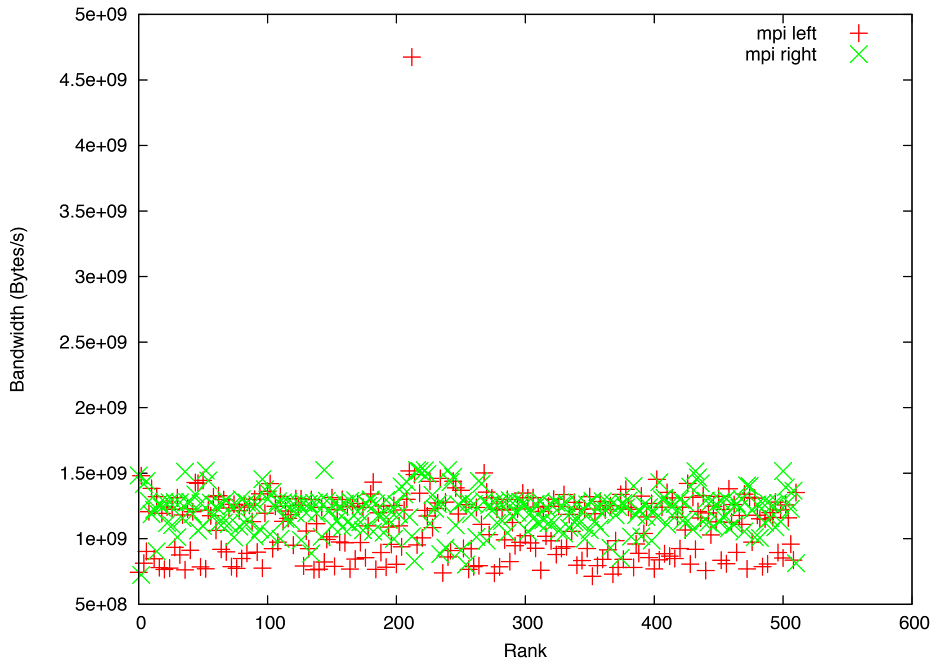


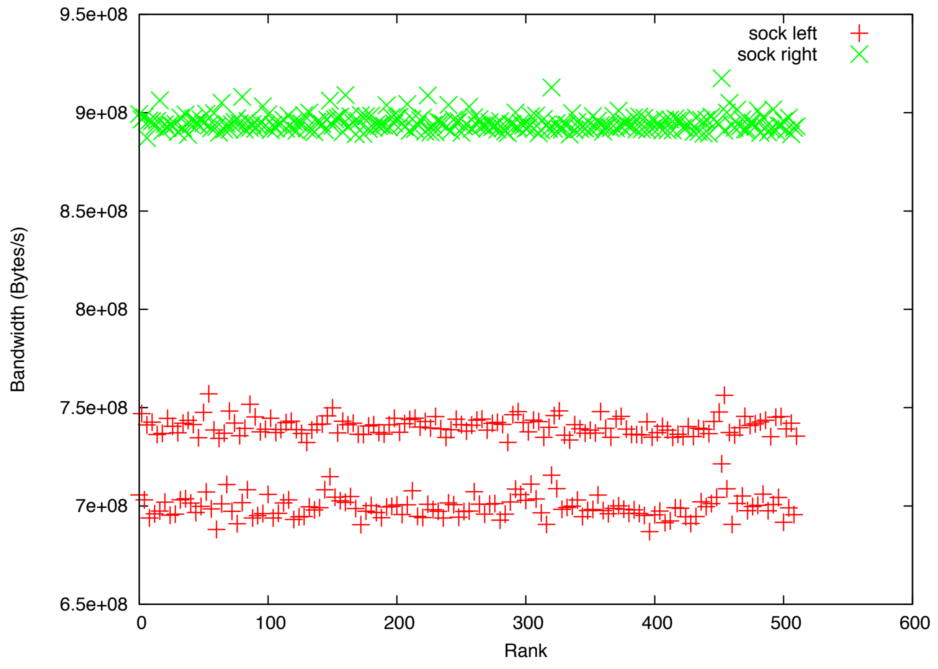
Jaguar XT4 MPI and Socket Latency, Fully Populated Nodes



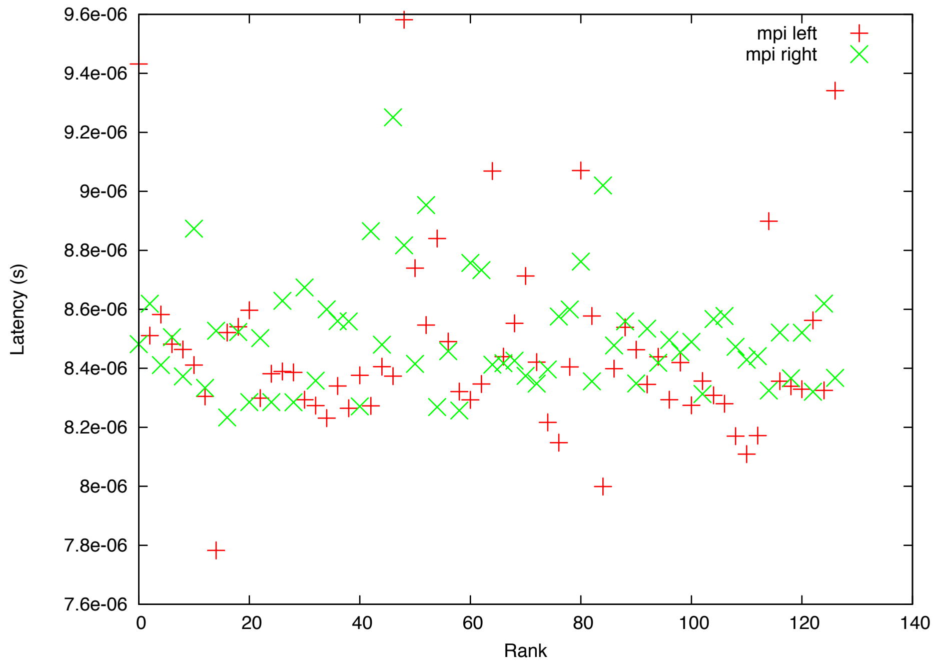


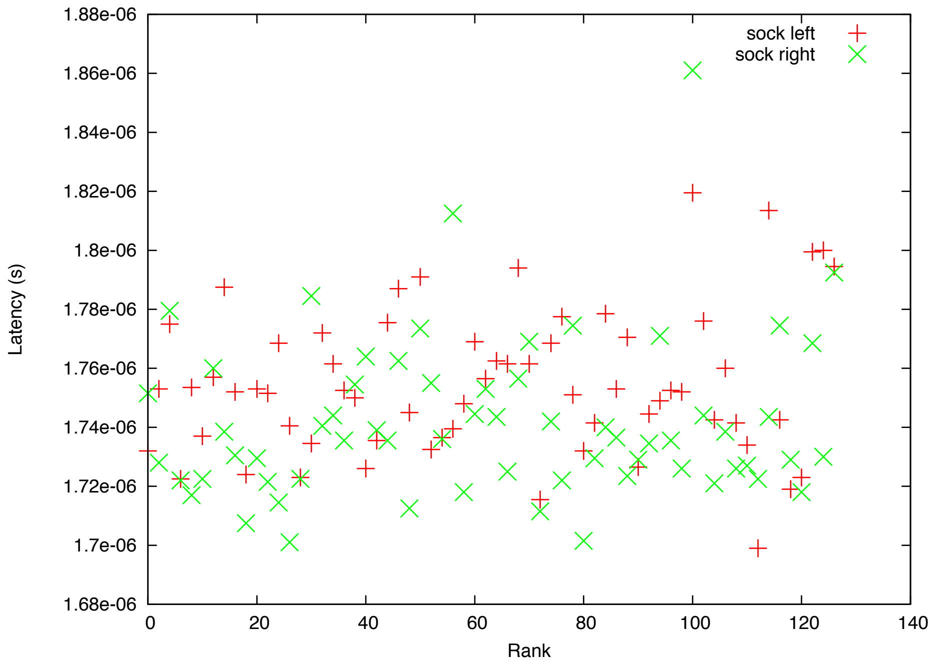
Jaguar XT4 MPI and Socket Bandwidth, Fully Populated Nodes



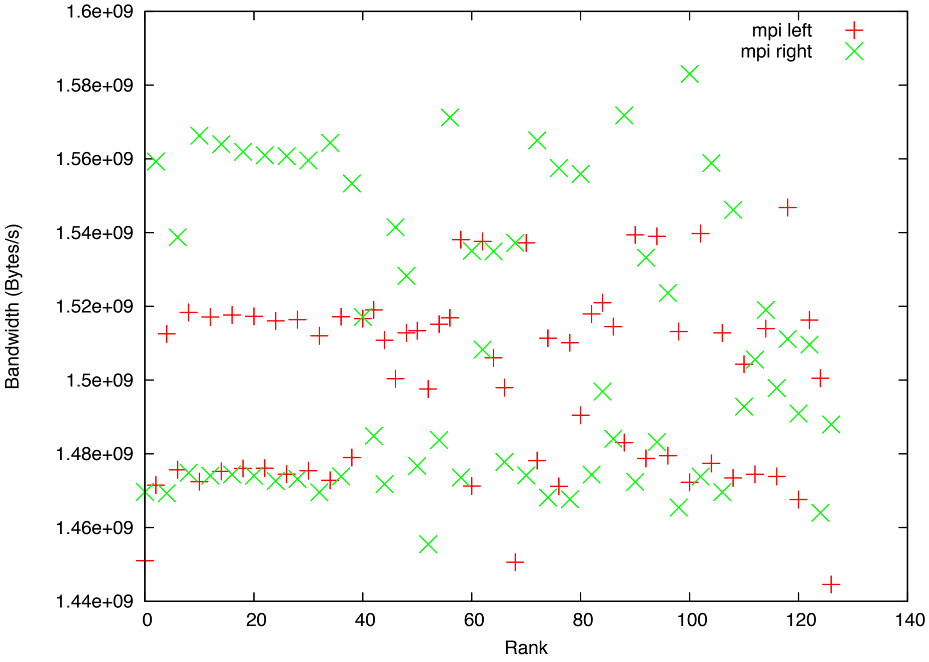


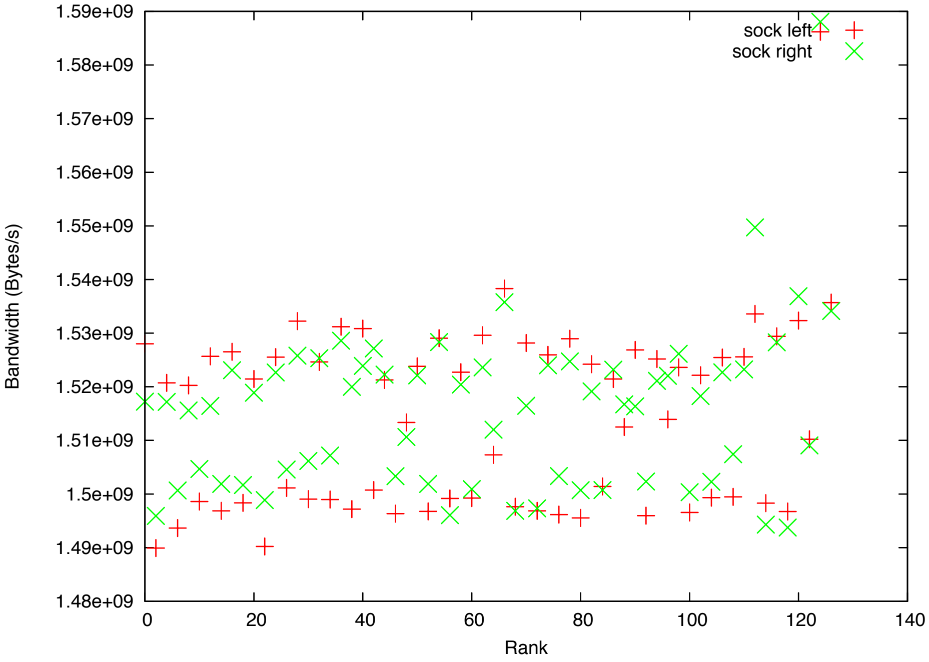
Jaguar XT4 MPI and Socket Latency, One Process Per Node





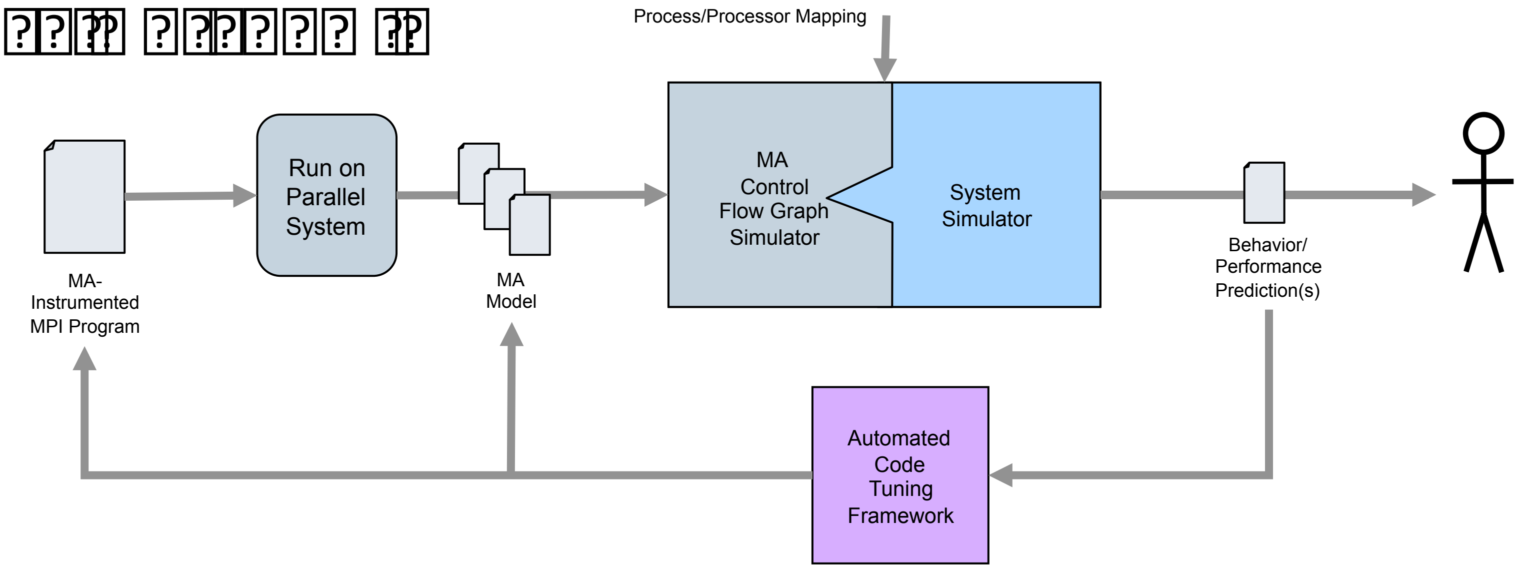
Jaguar XT4 MPI and Socket Bandwidth, One Process Per Node



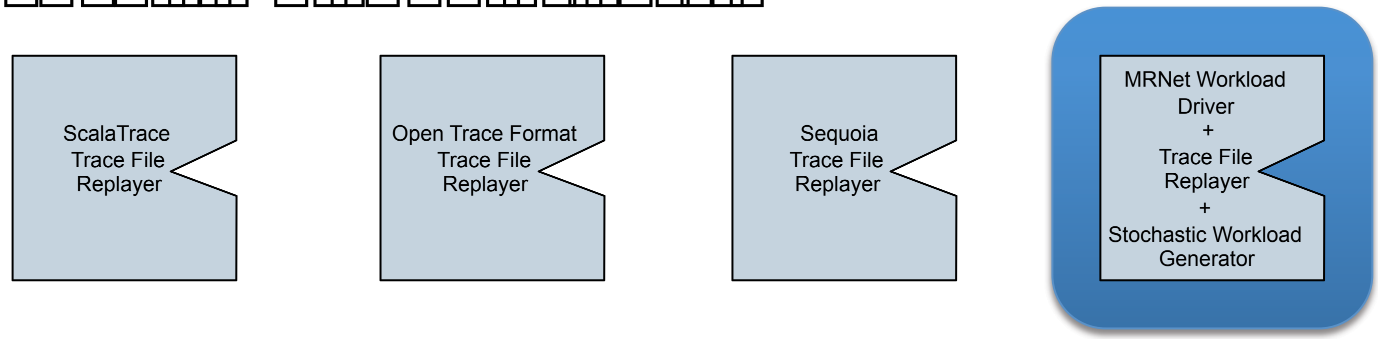


Simulation Flexibility

- [illegible]


Process/Processor Mapping
MA-Instrumented MPI Program
Run on Parallel System
MA Model
MA Control Flow Graph Simulator
System Simulator
Behavior/ Performance Prediction(s)
Automated Code Tuning Framework

- [illegible]


ScalaTrace Trace File Replayer
Open Trace Format Trace File Replayer
Sequoia Trace File Replayer
MRNet Workload Driver
+
Trace File Replayer
+
Stochastic Workload Generator

Status

- Basic XTNode with SeaStar router is implemented
  - Parameterization still in progress as described earlier

- Support for simple MPI-based workloads
  - Hardcoded behaviors (hot potato, 1D exchange)
  - OTF and Sequioa trace readers implemented for previous version, must be resurrected

- Support for TBON processes designed and partially implemented

- Recently adapted model from OMNeT++ 3.2 to 4.1 (changes in simulation time)

Acknowledgements

- This research is sponsored by the Office of Advanced Scientific Computing Research; U.S. Department of Energy. The work was performed at the Oak Ridge National Laboratory which is managed by UT Battelle, LLC under Contract No. De-AC05-00OR22725.
- This research used resources of the Center for Computational Sciences at Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. De-AC05-00OR22725.

Summary

- Predicting TBON performance on Cray XT is highly desirable
  - Matching TBON process topology and placement to tool needs subject to application and system constraints
  - May support online reconfiguration of TBON topology
- Developing simulation-based TBON prediction capability
  - Expect predictions of realistic scenarios soon
  - Easily adaptable to expected future architectures (e.g., GPU-enabled nodes, Infiniband clusters)
  - Embeddable (in theory)

For more information

http://ft.ornl.gov

rothpc@ornl.gov

http://www.paradyn.org/mrnet