

Challenges and Opportunities (for OSCAR) at the Mid-Range of Cluster Computing

OSCAR Symposium

HPCS 2004

May 17th, 2004

Dan C. Stanzione Jr.

National Science Foundation, USA

The views and opinions of the speaker do not necessarily represent the views of the NSF and the US Government



Outline

- Near to Mid Term challenges in High Performance Computing and Cyberscience
- What these have to do with OSCAR
- What these don't have to do with OSCAR, but we should do anyway

Science Drivers For High Performance Computing

- That they exist, are plentiful, and significant, is no longer a question
- Examples:
 - Materials Science
 - *Computation-based nanoscale materials design*
 - Genomics
 - *Accessing all genes, all proteins, in real time*
 - Biology
 - *Regulatory networks, protein folding, biodiversity and systems biology*

Science Drivers For High Performance Computing

- More Examples:
 - Cosmology
 - *Energy content of the universe*
 - *Galaxy and Star Birth*
 - Relativity
 - Plasma Science/High Energy Physics
 - Physics of nano-scale electronic structures
 - Biology
- In all cases, massive need for simulation to guide experiments

Science Problems for HPC

- Software Challenges
- Model Challenges
- Cultural Challenges

(Only some of these are OSCAR problems, but all of them are problems the community must face)

Software Challenges

- Trust comes slowly in the scientific community, and large applications are fundamentally hard.
- The large apps that are at the core of much computational science take years to build and an investment of \$10s of millions.
 - (and it's incredibly difficult to fund this kind of sustained effort)
- That's been true for years; but there are many new software challenges arriving...

Software Challenges

- Data Management
- Integration, interoperability
- Theory to ensure accuracy and stability
- Data Mining
- Algorithm Research
 - Massive Dimensions
 - Statistical challenges
 - Multiscale computation

Software Challenges

- Software Engineering Challenges
 - Ensure quality, but don't overconstrain
 - Maintain and extend massive simulation codes
 - Migration of domain-driven software to other applications
- Interface?
- Maintain portability, but don't sacrifice performance or crush architectural innovation...

Model Challenges

- From a system software perspective, we've made little progress in the last 30 years
 - Still fundamentally a UNIX model
 - Is this a good thing, or is it slowing us down?
- From an application software perspective, progress is also glacial
 - New languages? Many have tried...
 - Libraries remain successful approaches
 - Still no science-user-friendly model that allows for cluster innovation.

Cultural Challenges

- Academic/Research culture isn't very good at rewarding the kind of needed efforts
 - Little value placed on software artifacts
 - Less placed on maintenance, migration
- There is a large gap between “proof-of-concept” and “commercially viable”
 - Most science codes are never commercially viable
- Funding agencies (mine included), are not very good at funding the sustained, long term effort scientific software takes

The Opening at the Mid-Range

- (In the US), there are a few massive well-funded supercomputer centers ($> \$5\text{M}/\text{yr}$).
 - Funding will be found to keep these afloat
 - Great science comes out of these centers
- Funding is (relatively) easily available for small clusters ($< \$200\text{k}$); probably thousands of these exist.
- Very little attention is paid to the space in-between

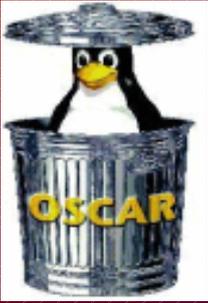
More on the Middle

- The big centers are oversubscribed
- Most Science doesn't happen in the giant multi-month runs of the top few codes;
 - Most science is **lots** of users making **lots** of runs
- Most code development does not happen on just a few large systems
- The next generation of computational scientists can't all be trained in three centers

Funding Agency Impetus

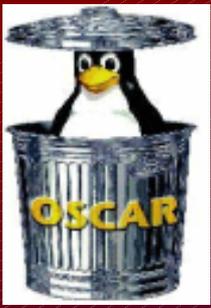
- NSF has been told **repeatedly** to expect no new funding before 2009
- NSF is under significant pressure to increase award size and duration
- NSF is under some pressure to more strongly address interdisciplinarity (match society's problems)
 - Canadian agencies (NSERC, SSHRC) see similar trends
- There simply **will be** fewer awards in the future.
- *There will be dramatically fewer small, single investigator awards: there will be more multi- and inter- disciplinary teams*

The message: Computational Science is likely to proceed primarily not at the very small or very large, but somewhere in between



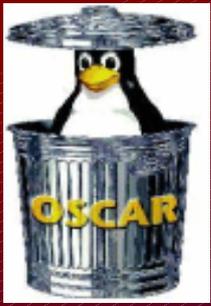
Opportunities for OSCAR

- Best practices - OSCAR as technology transfer
- Move up the abstraction chain
- Attack the mid-range
- Clusters of clusters; and on up to grids (not necessarily the same thing at every scale)
- Novel Cluster Uses
- OSCAR as education



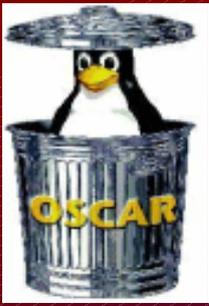
OSCAR – Best Practices

- Original intent, still the best idea
- On a university campus last week, I met 15 cluster admins
 - 3 had CS or ECE people involved
 - 10 more research groups wanted clusters (or cluster access), but were intimidated by the process
- OSCAR brings the best and the stable from the cluster research world to the broader community
 - This is still a crucial mission



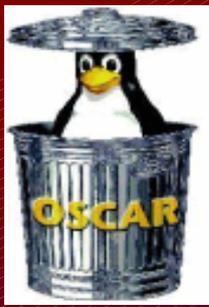
OSCAR and Responsible Tech Transfer

- “Best” however, doesn't mean “every”.
- A broad array of new options for cluster SW appear regularly
 - Monitoring, scheduling, libraries, etc.
- Most have some good ideas, but...
 - A few choices make people feel good; too many leaves them bewildered.
- Continuing to make the right choices is a challenge for OSCAR governance



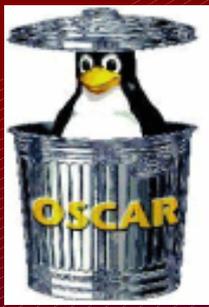
Move Up the Abstraction Chain

- The traditional role of OSCAR has been cluster management.
- A natural expansion path would be to move past the cluster administrator, and address the cluster programmer.
- Become the distributor of Best Practices for Tools and Problem Solving Environments
 - ALICE?, CCA?
 - Template Applications?
- Domain specific OSCARS...



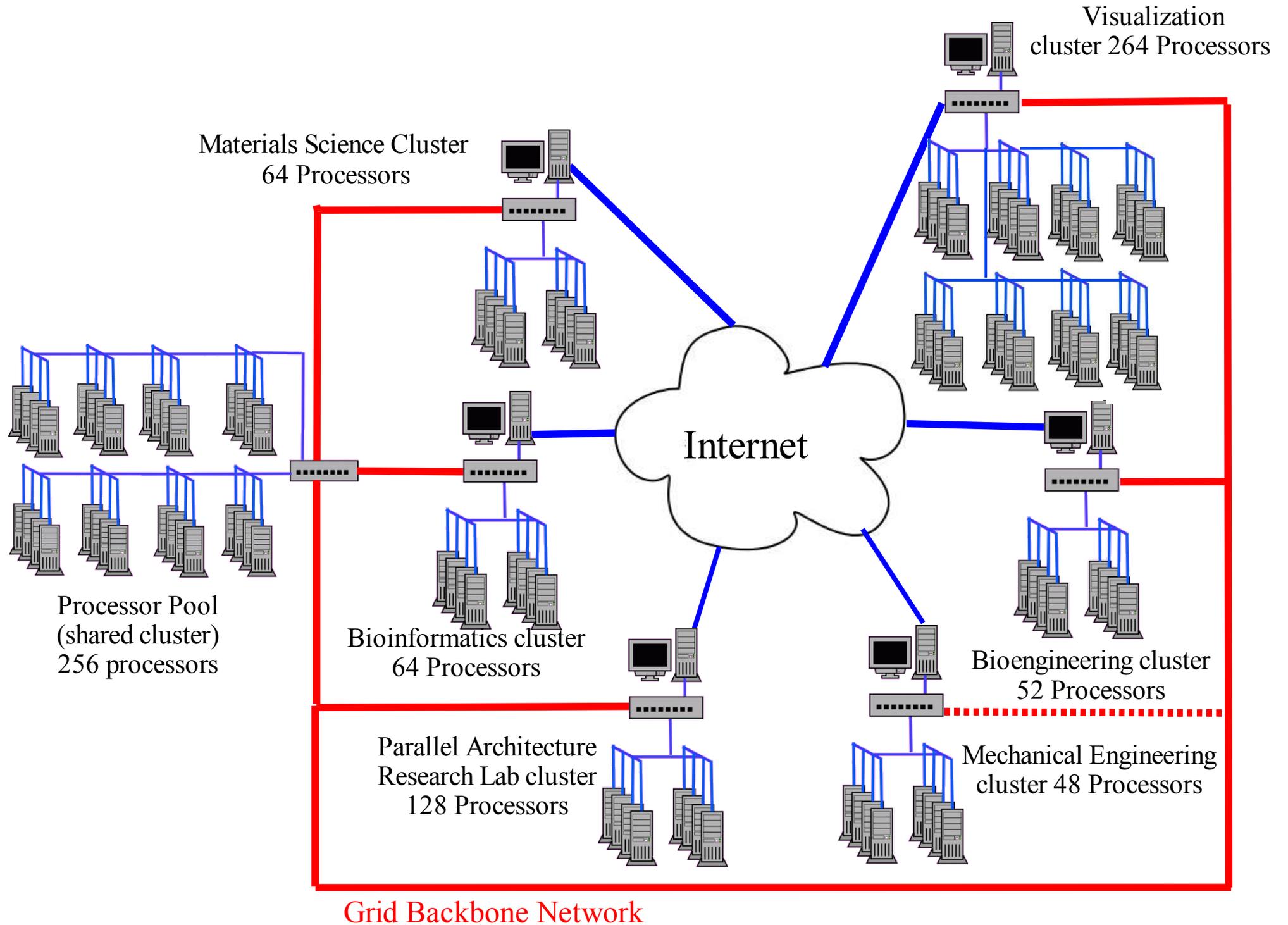
Attack the Mid-Range

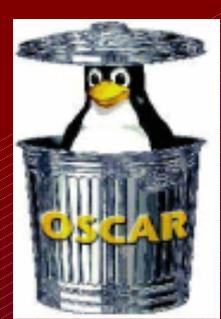
- OSCAR was designed for tens of nodes
- Cluster research now frequently targets thousands of nodes
- It's likely the bulk of systems will range in the hundreds of nodes in the near term
- OSCAR development should focus in this range
 - Some reasonable measures for scalable install, presence of faults, job launch, etc.



Multi-Cluster and Grids

- Another reality of the mid-range is multiple clusters per campus
 - Many small, a few mid-size
- Grids come in many sizes, and they are not self-similar at all scales
 - Grid computing attacks the broad challenges:
 - Heterogeneity: Desktop to supercomputer
 - Global Scale: Many nodes many networks
- A Mid-Range Opportunity for OSCAR: leveraging multiple clusters in reasonable proximity
 - Co-scheduling, shared user spaces





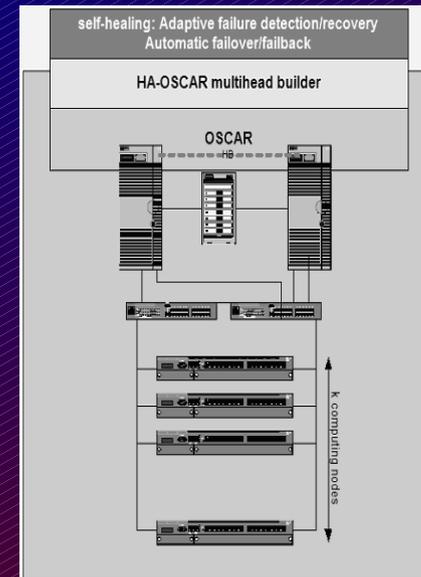
Novel Cluster Uses

- Cluster Architecture is not necessarily dead; OSCAR can be an enabling technology for new ideas in clustering

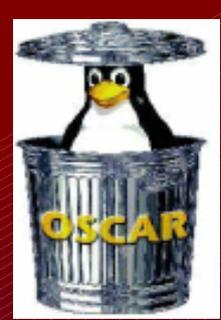


Adaptive Computing Cluster: Clusters plus Reconfigurable Hardware

HA-OSCAR: High-Availability through redundant networks



Distributed Rendering: Clusters plus graphics hardware



OSCAR as an Educational Resource

- There is tremendous demand for education in clustering
- OSCAR makes clusters pop up anywhere
 - This laptop is the head node of my home OSCAR cluster
- How does anyone learn to administer clusters?
 - All the best ideas in one place

Best Practices – Beyond OSCAR

- OSCAR is successful largely because it's a vehicle for distributing cluster best practices
- Now, if only we could do that for the rest of the computing world
 - Supercomputing applications aren't the only hard ones
- Problems facing large parallel applications are largely unsolved in IT in general... why would we solve them in high performance computing first?

Computer Science (and Engineering) at Cross-Purposes

- Training Computer Scientists and training programmers are not the same things.
- The skills of a graduate student include programming; but not necessarily software design (and are absolutely not limited to programming)
- Academic departments prepare researchers
- This is bad for both research and the broader IT profession...

Computer Science: An overloaded degree

- Computer Science degrees are a little like operating systems: they suffer from feature creep (every new topic finds a way in, diluting the rest)
- The problem is **not** that any particular topic is invalid; the problem is the field is broad
- There are best practices to be had here, too!
- The model is the sciences...
 - What's it mean to get an advanced degree in biology? Answer: No such thing:
 - Molecular biology, cellular biology, evolutionary biology, systems biology, not to mention genetics, genomics, ecology, primatology, plant physiology, phylogeny

Solutions:

- Applied Computer Science; Software engineering, or even IT if you must (but not MIS). Focus on:
 - Systems (teaching OO is not a surrogate)
 - Design
 - The Practice of Programming (starting with the book of the same name might not be a bad start).
 - Debugging
- Terminal Master's degrees
 - A dose of entrepreneurship

Solutions:

- Restore the science to computer science
 - For the non-applied, teach research methods, statistics, use of core literature in scholarship
 - More science, more collaboration
- Research PhDs as stewards of discipline; professional degrees for IT.
- Solve the IT software problems, many HPC problems get simpler...
- Nip the burgeoning post-doc movement in the bud, or end up like biology in the bad ways too.

Shameless Plug

- Clusterworld Magazine

<http://www.clusterworld.com>



- 3 month free offer for US customers
(I encourage Canadian customers to negotiate!)
- See OSCAR articles in March/April issues – more to come!

Conclusions

- There is a lot of fertile ground left for OSCAR to cover!
- The best growth path for OSCAR is not at the very large or very small systems, but in the middle, where most computational science happens
- Research in computing systems will continue to flourish even in a flat funding environment, but we may need to operate differently (more and bigger teams!) -- for those of us in computer science and engineering, change our profession to do it.