PBS GridWorks

Select, Place, and Vnodes: Exploiting the PBS Professional Architecture on Cray Systems

Bill Nitzberg, Ph. D.

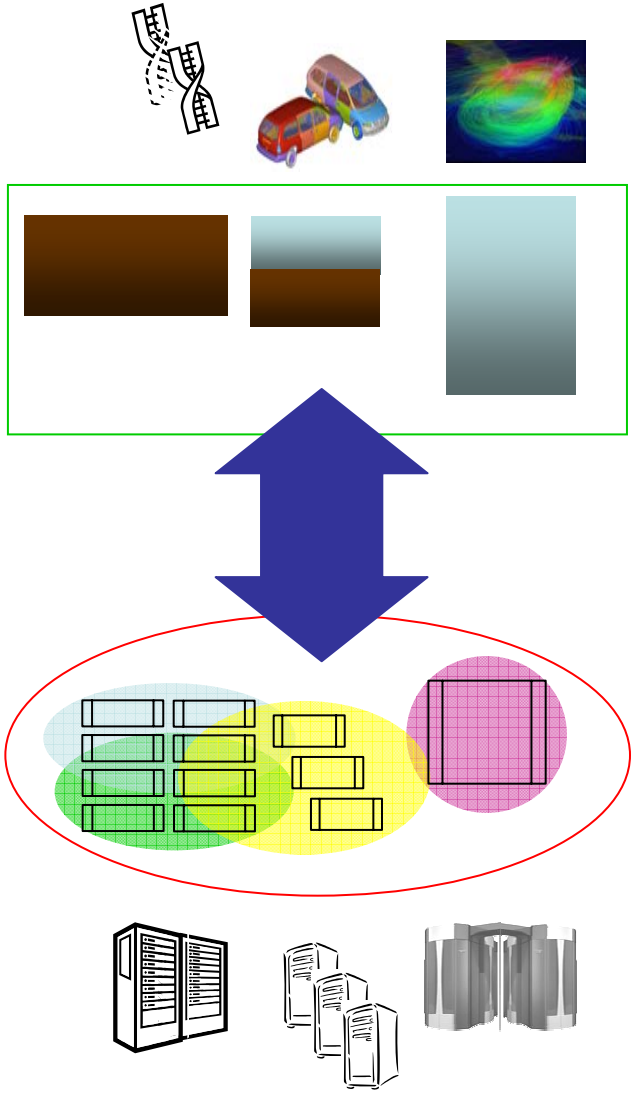
CTO, PBS GridWorks

Altair Engineering, Inc.

May 4, 2009

PBS GridWorks

One-Slide Summary™



Virtualized job requirements

MPI, OpenMP, SMP

Jobs select resources in "chunks"

- E.g., one MPI rank

Topology via grouping

PBS matches chunks & vnodes optimizing placement

Topology via placement sets

Machine resources in "vnodes"

- E.g., one node/blade/socket

Virtualized machine resources

SMP, Cluster, NUMA

Altair Overview

A global software and technology company focused on

- enterprise analytics,
- product development, and
- advanced computing.

Altair

Global Expertise: 45 Offices in 17 Countries

Seattle, USA
Mtn View, USA
Los Angeles, USA
Austin, USA
Dallas, USA

Mexico City, Mexico

Toronto, Canada
Windsor, Canada

Detroit, USA
Boston, USA
Milwaukee, USA
Atlanta, USA

Sao Paulo, Brazil

Lund, Sweden
Gothenburg, Sweden
Coventry, UK
Manchester, UK
Boeblingen, Germany
Cologne, Germany
Hanover, Germany
Munich, Germany
Paris, France
Sophia Antipolis, France
Torino, Italy
Milan, Italy

Moscow, Russia

Delhi, India
Pune, India
Chennai, India
Hyderabad, India
Bangalore, India

Kuala Lumpur, MY

Beijing, China
Shanghai, China

Tokyo, Japan
Osaka, Japan
Nagoya, Japan

Seoul, Korea

Melbourne, Australia

3,800+ Customers World-wide

Local Customer Interface

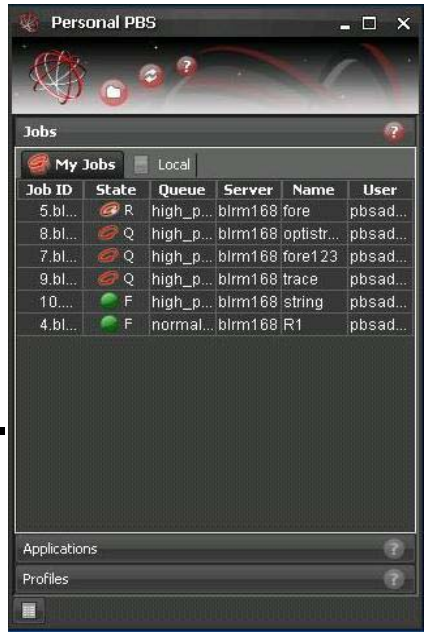
Local Market Knowledge

PBS GridWorks Solution Suite

GridWorks Analytics- accurate reporting, rich analysis

GridWorks Portals- easy to use web based portals, rich domain knowledge

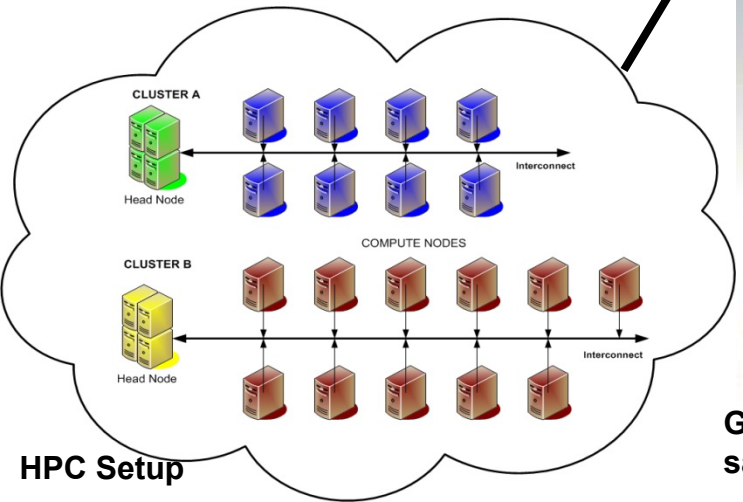
Personal PBS- drag and drop job submission



PBS GridWorks
Enabling On-Demand Computing™

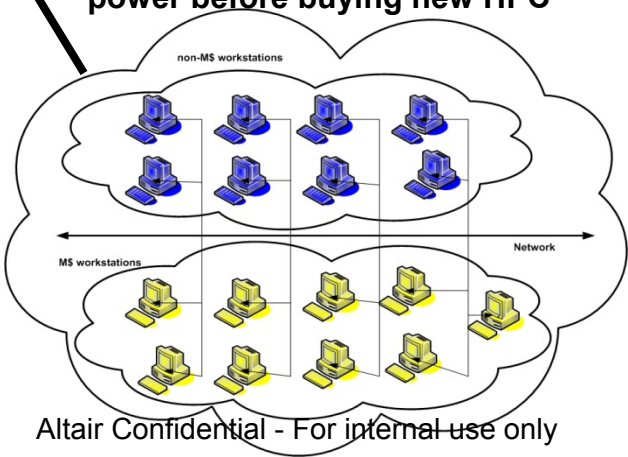
PBS Professional™- Robust (setup & forget), Scalable(10000s of cores), Flexible

Desktop Grid- utilize desktop power before buying new HPC



HPC Setup

Green Provisioning™- save energy, save $$, save the planet



Why?

In the past… hardware was simple:

Symmetric Multiprocessors and "Clusters of "Workstations"

Today… a single system is a heterogeneous mix of technologies:

SMP + cluster + multi-core + NUMA + FPGA + …

The PBS architecture provides A Higher-level Abstraction to easily and automatically access unique qualities and raw power of today's high performance computers

PBS GridWorks

Abstract Application-level Job Submission

Jobs select application requirements, not hardware

- E.g., MPI, OpenMP, cpus, memory, licenses, scratch space

Users do not need to know hardware details

→ **improved productivity**

Jobs are not restricted to particular hardware

→ **improved utilization**

Jobs can specify placement, sharing and exclusivity

→ **improved performance**

Add / change / replace hardware with minimal user impact

→ **lower maintenance**

Abstract Virtualized Hardware & Resources

Virtual nodes (vnodes) represent all types of hardware

- E.g., SMP, MPP, Sockets, NUMA, /tmp, NAS devices, software licenses, …

Single representation for heterogeneous resources

→ **greater expressive power**

Separation of Host – MOM - node

→ **greater scalability**

All PBS Professional features for all architectures

→ **agility to deliver more features, faster**

Topology-aware scheduling with placement sets

→ **increased performance & reduced fragmentation**

select & place

PBS GridWorks

Jobs Select Resources in "Chunks"

qsub -l select=128

Application with 128 chunks == simplest 128-way MPI

Default is 1 chunk == 1 MPI rank with 1 OpenMP thread on 1 cpu

Chunk: "natural" unit of allocation (for the application)

- E.g., memory is shared by all processes in one chunk

MPI: ranks assigned left to right, override default with "mpiprocs=N"

qsub -l select=1:mem=16gb+15:mem=1gb

OpenMP: threads per MPI rank, override with "ompthreads=M"

qsub -l select=8:ompthreads=4

PBS GridWorks

qsub –l select=…

Only specify application requirements (e.g., MPI)

4-way MPI job with 6GB ranks

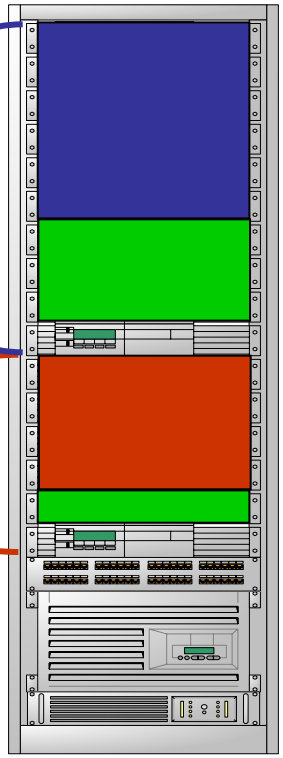
```
qsub -l select=4:ncpus=1:mem=6gb
```

6-way MPI w/ 2 OpenMP threads & 2GB

```
qsub -l select=6:ncpus=2:mem=2gb
```

4-way MPI job with rank 0 of 2 OpenMP & 6GB and ranks 1-3 of 2 OpenMP & 2GB

```
qsub -l select=1:ncpus=2:mem=6gb+3:ncpus=2:mem=2gb
```



PBS GridWorks

Place Modifies Resource Selection

-l place=scatter

- Each chunk is allocated to a separate host

```
qsub -l select=16:ncpus=1:mem=1GB -l place=scatter
```

-l place=pack

- All chunks are allocated from vnodes on the same host

```
qsub -l select=2:ncpus=2:mem=1GB -l place=pack
```

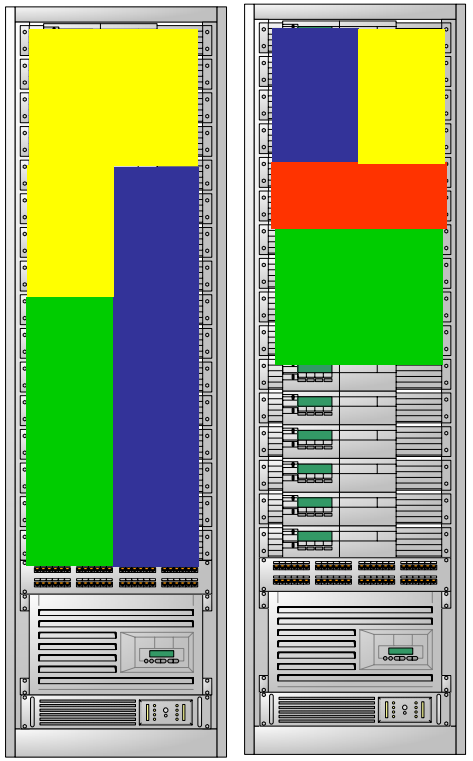
-l place=free

- Resources within the chunk can be taken from any vnode where are available

```
qsub -l select=16:ncpus=1:mem=1GB -l place=free
```

-l place=excl

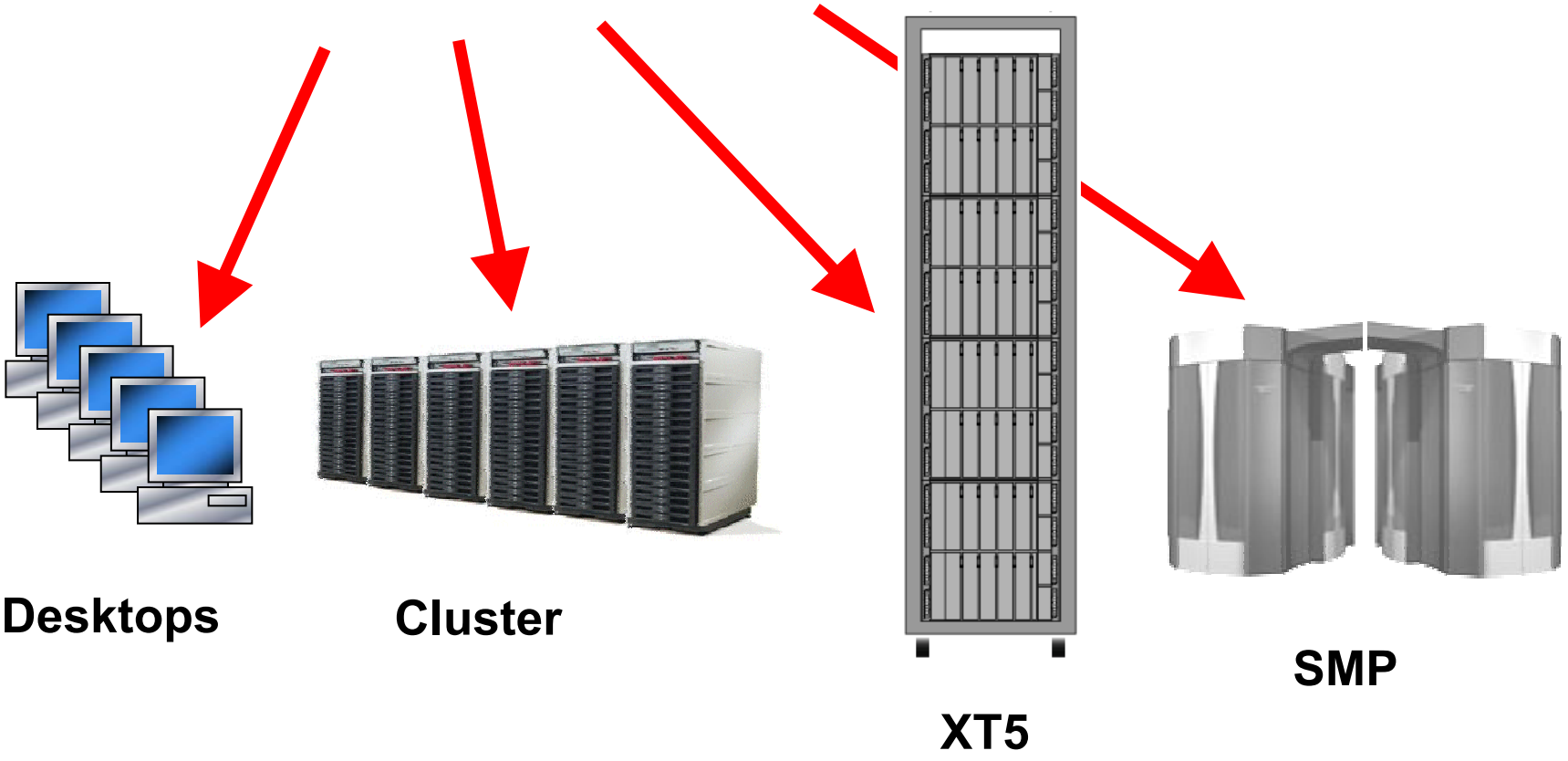
- Allocate vnodes exclusively for the use of this job
- Can be combined with above (e.g., place=scatter:excl)



PBS GridWorks

One Select → Any Hardware

qsub -l select= 16 : ncpus=2 : mem=2gb



vnodes

PBS GridWorks

"Virtual" Node

"Natural" resource container

hal[0]: ncpus = 2
hal[0]: mem = 1968448kb
hal[0]: fpga = True
…

Canonical system is one vnode per host

- NUMA systems have >1 vnode per host
- PBS on XT5 today has <1 vnode per host

Resources can be shared at any level

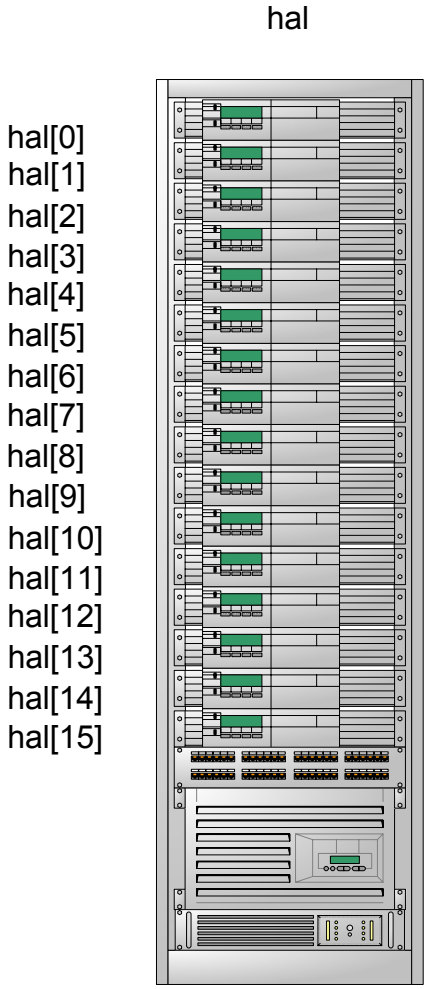
- Global, cluster-wide, rack-wide, host, blade, …

Exclusive and shared allocations

- Shared vnodes → cpu-bound jobs
- Exclusive access → memory-bound jobs

Enables simulation of huge systems for test

- 50,000 vnodes on my laptop



placement sets

PBS GridWorks

Placement Sets are Sets of vnodes

Placement sets are defined by multi-valued string resources

- hal[1] resources_available.router=board0,Top-01,Rack-01
- hal[2] resources_available.router=board0,Top-01,Rack-01
- …
- hal[316] resources_available.router=board48,Top-01,Rack-02
- hal[317] resources_available.router=board48,Top-01,Rack-02
- …

Placement sets can be hierarchical and overlapping

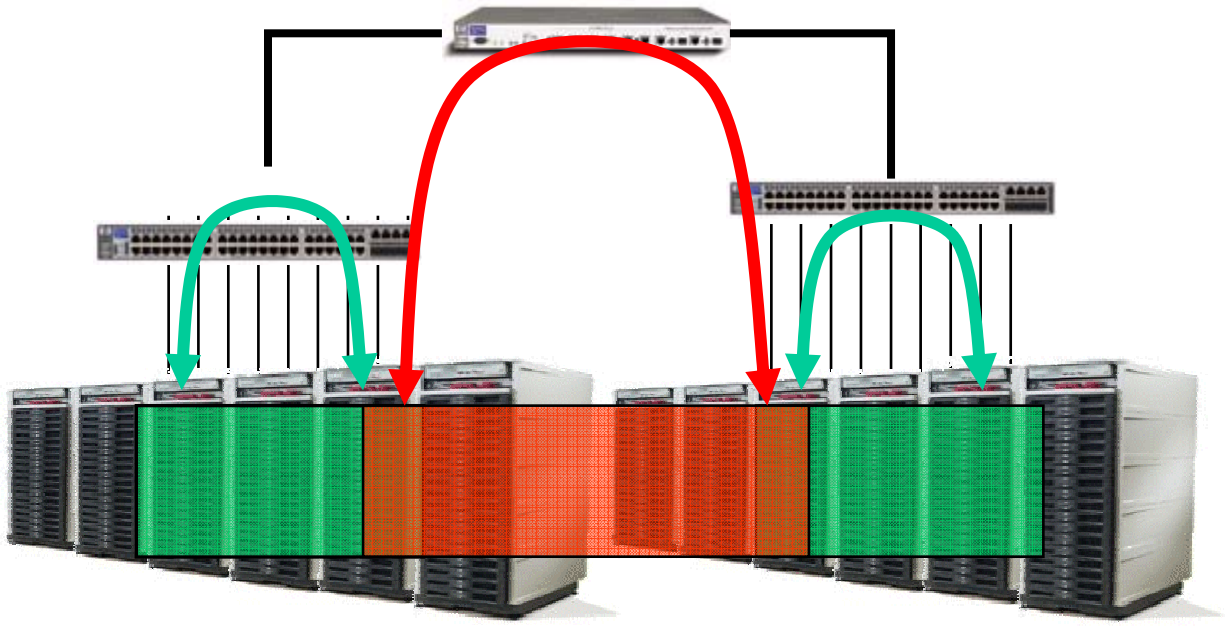
- Same technology works intra and inter machine
- Customer tunable – you can create/modify any topology

Scheduler optimizes by fitting jobs into “smallest” set available

- Can also directly specify, e.g., fit job into a 16x16x16 cube

PBS GridWorks

Example Placement for Cluster

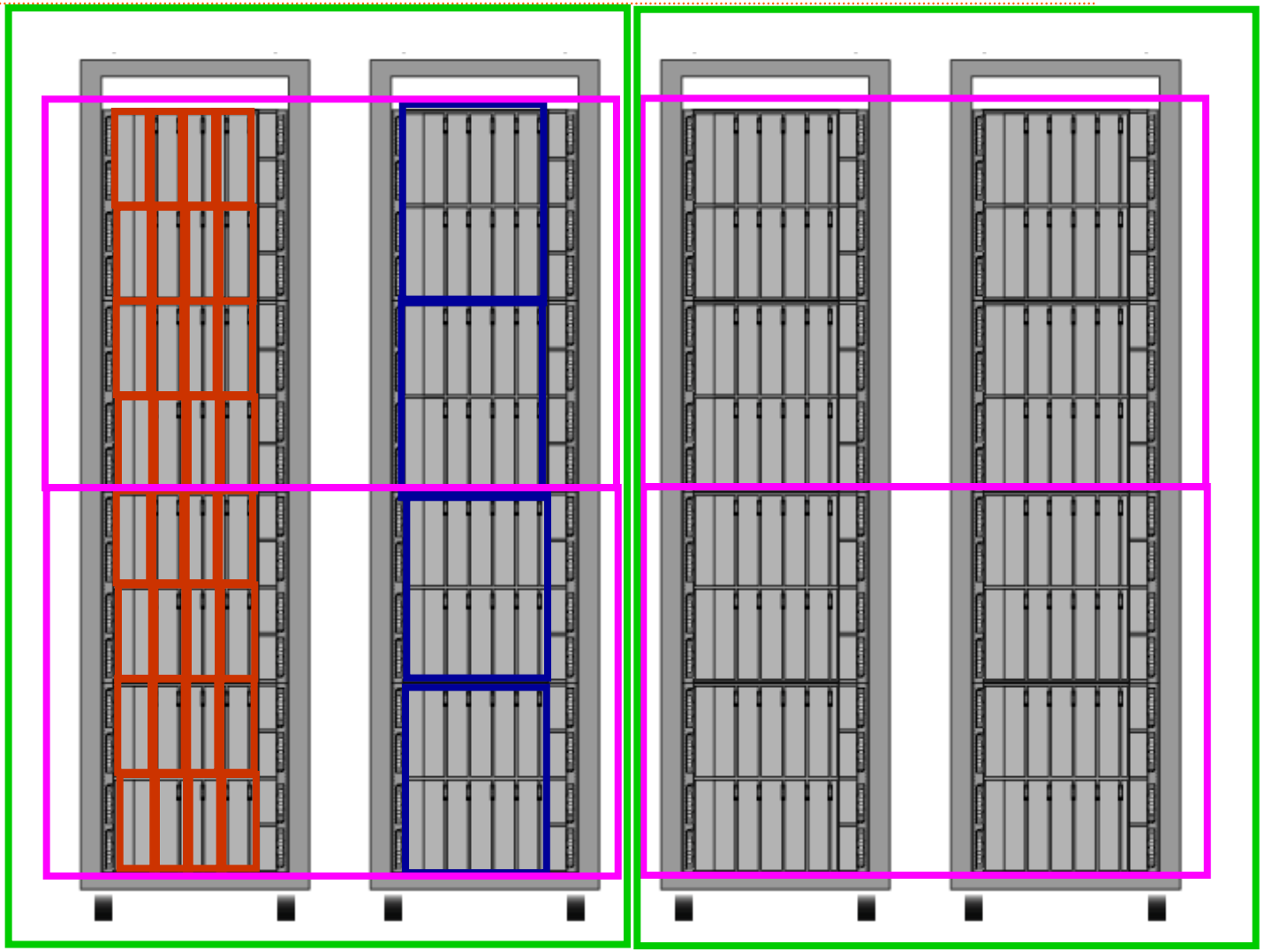


Automatically keep MPI processes near each other

Reduce interference and contention

PBS GridWorks

Example Placement on K-ary N-tree



Cray Platforms

PBS GridWorks

First: 100% Backward Compatibility

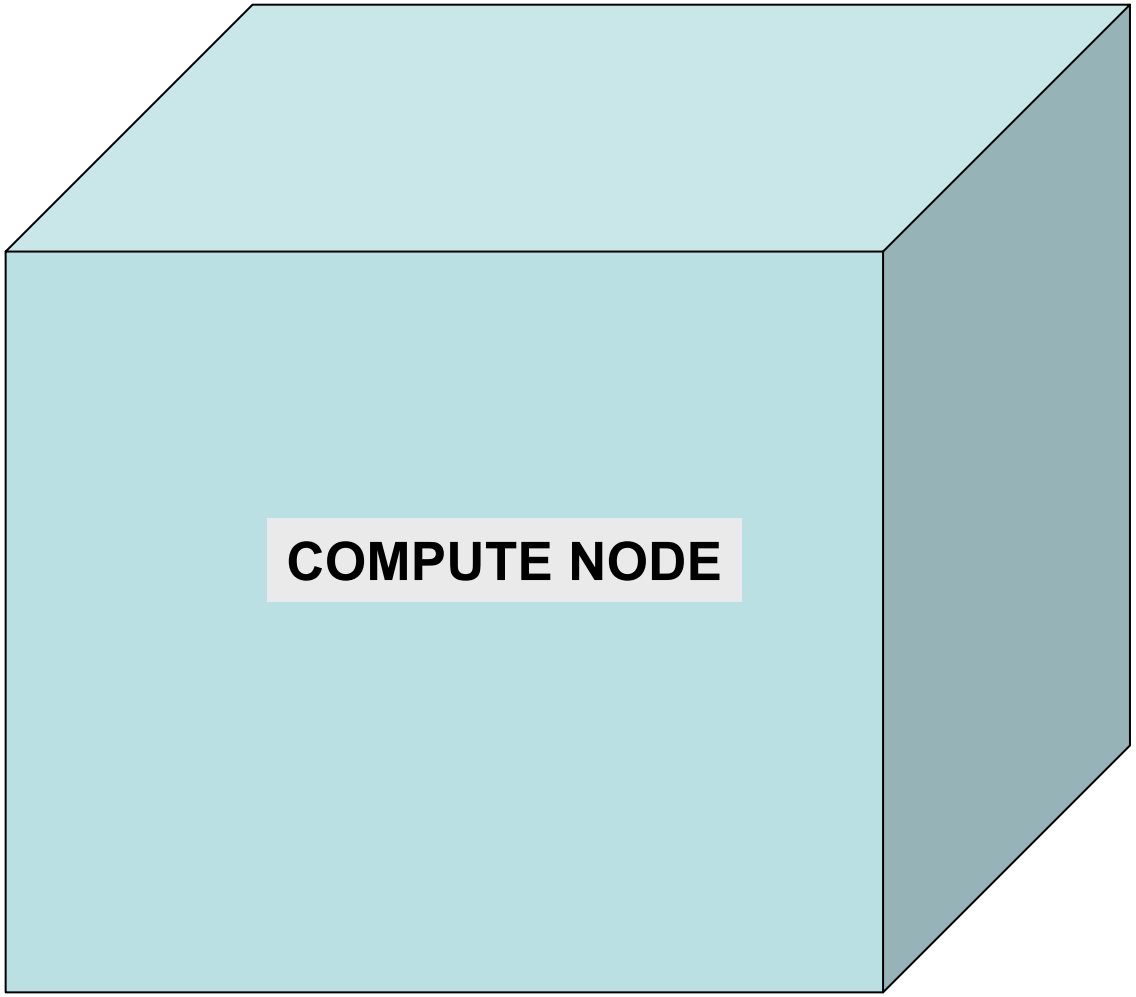
Our new “Hooks” capability provides the perfect mechanism to convert old “mpp*” syntax to new select & place

mppwidth	# of MPI ranks: (mpiprocs)
mppdepth	# of OpenMP threads (ompthreads)
mppnppn	# of MPI ranks per compute node
mpphost	Host (host)
mpparch	XT4 | XT5 (arch)
mppnodes	list of nodes (vnode)
mppmem	Memory (mem)
mpplabel	Any PBS resource for matching

PBS Professional will continue to connect directly to ALPS

PBS GridWorks

Old View: One “Super” Node



PBS GridWorks

Future: Every Compute Node is a PBS vnode

One vnode per node

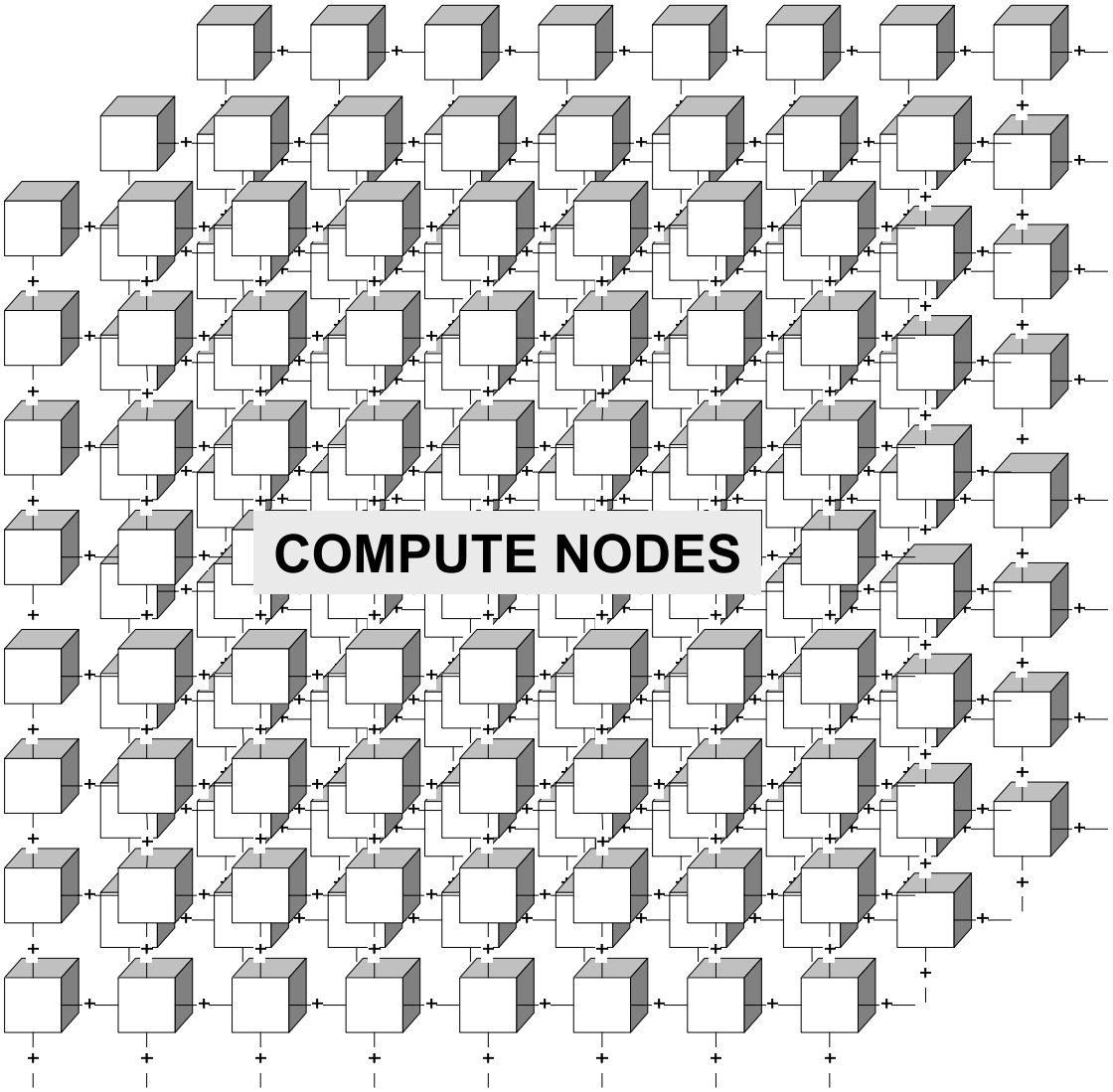
or

One vnode per socket

Address individual nodes

- Partition nodes to queues
- Use specific nodes (for benchmarking or test)

Address individual nodes/sockets/cores



PBS GridWorks

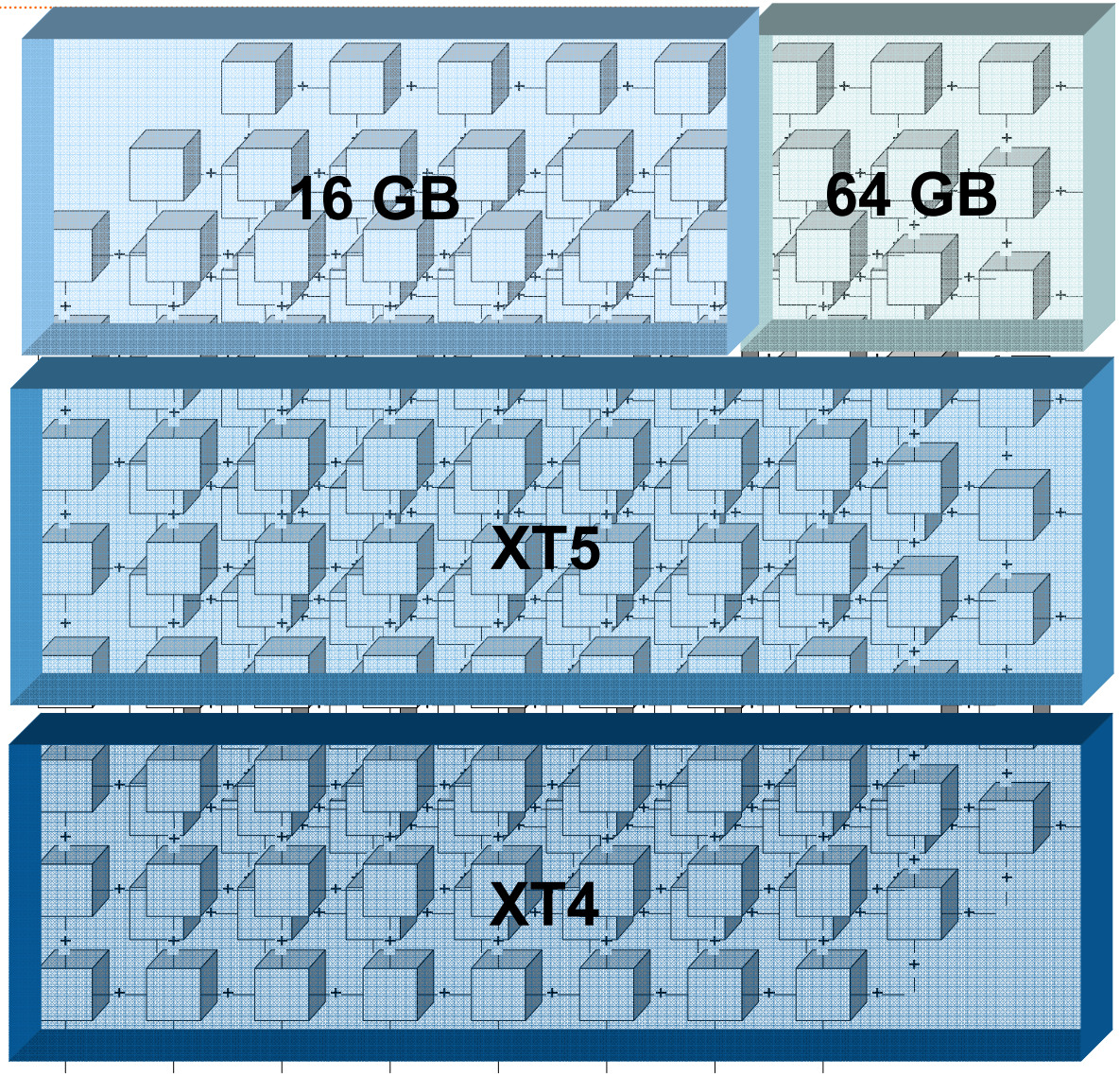
Select Unique Characteristics

Heterogeneous nodes

- Memory
- CPUs
- Architectures

Ready to support

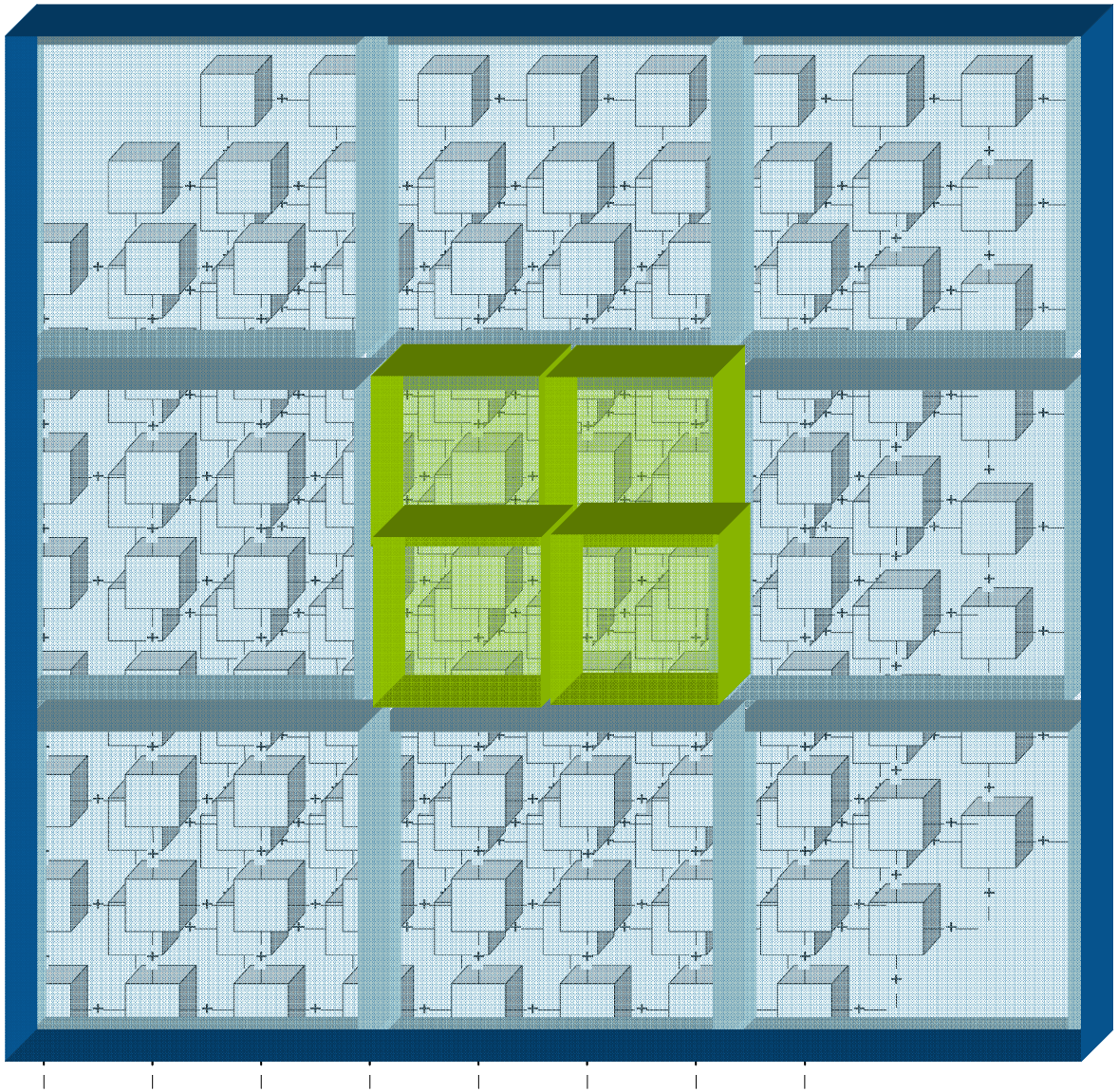
- Accelerators
- NUMA
- Any unique feature



PBS GridWorks

Optimize for Topology

- Create your own sets to suite unique needs
- Topology mapping via “node numbering” too
- Jobs/Queues can directly request placement
- Different queues can have different placements



What Else?

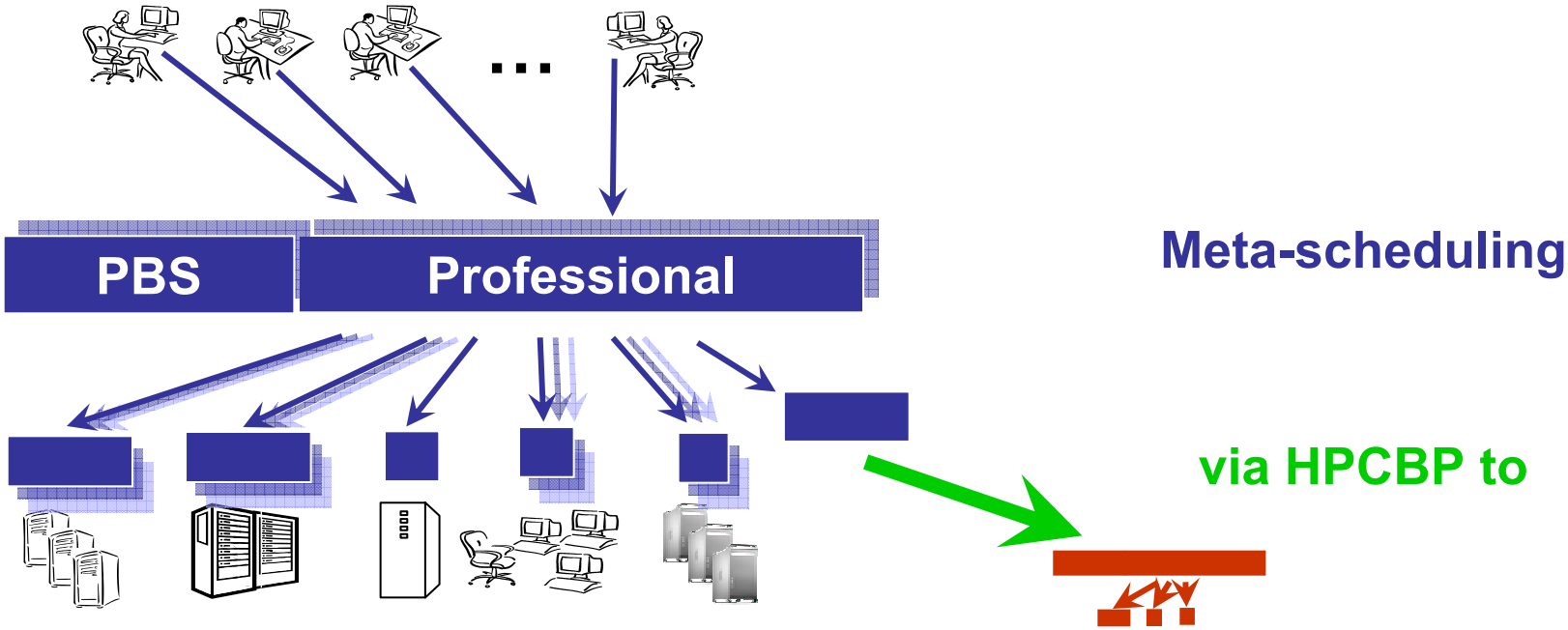
PBS GridWorks

Metascheduling (targeted for Jun 2009 v10.1 release)

No changes needed to Server & Scheduler

- They only deal in “vnodes”

Leverages Open Grid Forum OGSA HPC Basic Profile Standard



PBS GridWorks

Thank You!

e-Compute	e-Render		e-BioChem	Personal PBS
Fail-over	Huge scalability		$restrict_user	Checkpoint / Restart
Scheduling Formula	Standing Reservations		Select & Place	Job Arrays
Backfill	Green Provisioning	**Easy to Use**	Preemption	Eligible Time
License Scheduling	Age-based Scheduling	**Hard to Break**	Dynamic resources	Topology Placement Sets
Job Dependencies	Desktop harvesting	**Do More (with Less)**	Analytics	Multi-core
24x7 On-line Community	Policy-based scheduling	**Keep Track and Plan**	Meta-scheduling	Peer Scheduling
On-demand Licensing	Hooks	**Open Architecture**	MPI integrations	Web Services

PBS Professional®, Personal PBS™, GridWorks Analytics™, e-Compute™, e-Render™, e-BioChem™, Green Provisioning™