Enabling Contiguous Node Scheduling on the Cray XT3

Chad Vizino
vizino@psc.edu
Pittsburgh Supercomputing Center

Cray User Group
Helsinki, Finland – May 2008

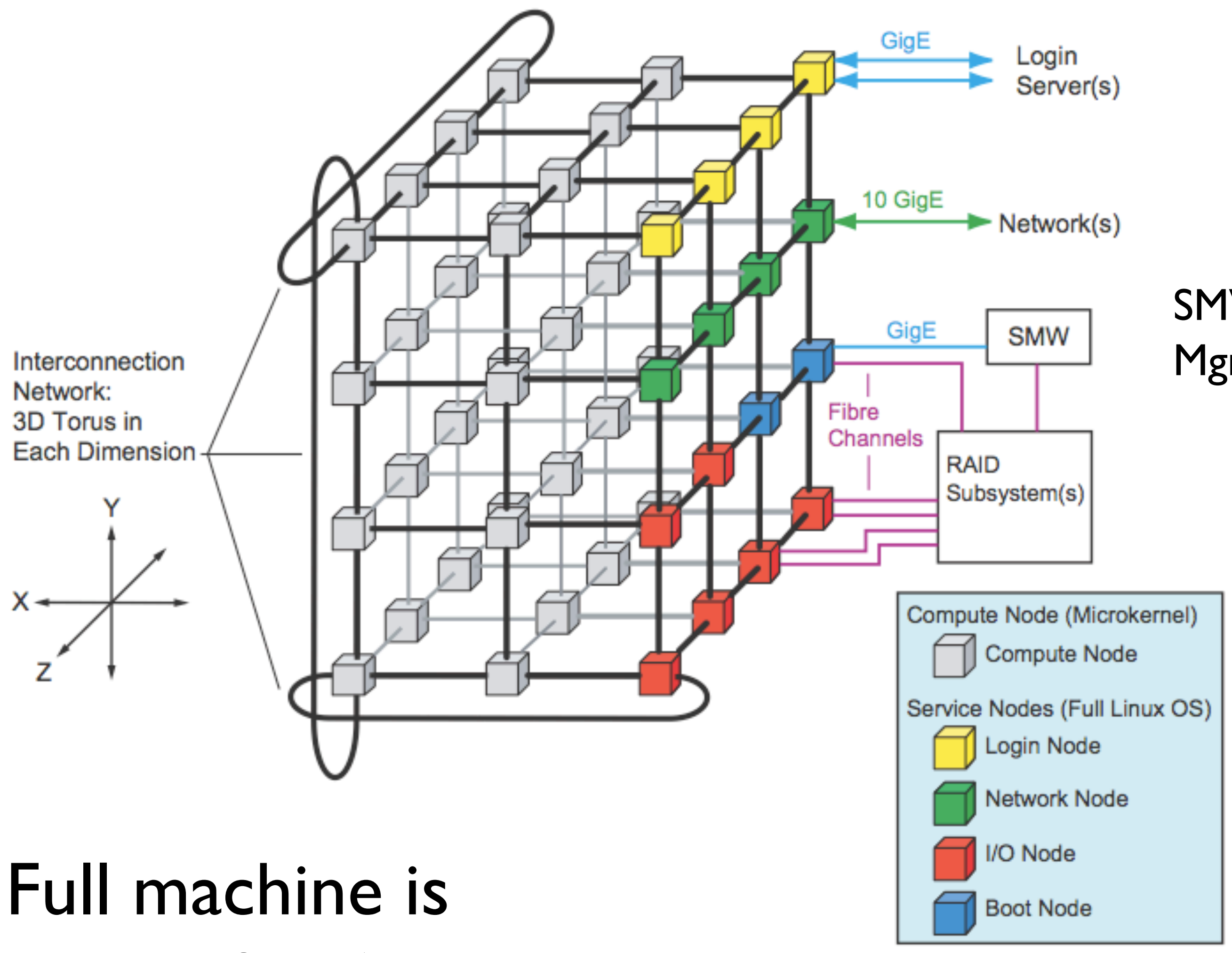
With acknowledgements to Deborah Weisser, Jared Yanovich, Derek Simmel, Abhinav Bhatele, and Adam Liwo

Overview

- Introduction to machine
- Previous work
- System changes to help improve application performance for
  - Topologically unaware codes
  - Topologically aware codes
- Conclusion

PSC'S Cray XT3

Cray XT3 Arch. Overview



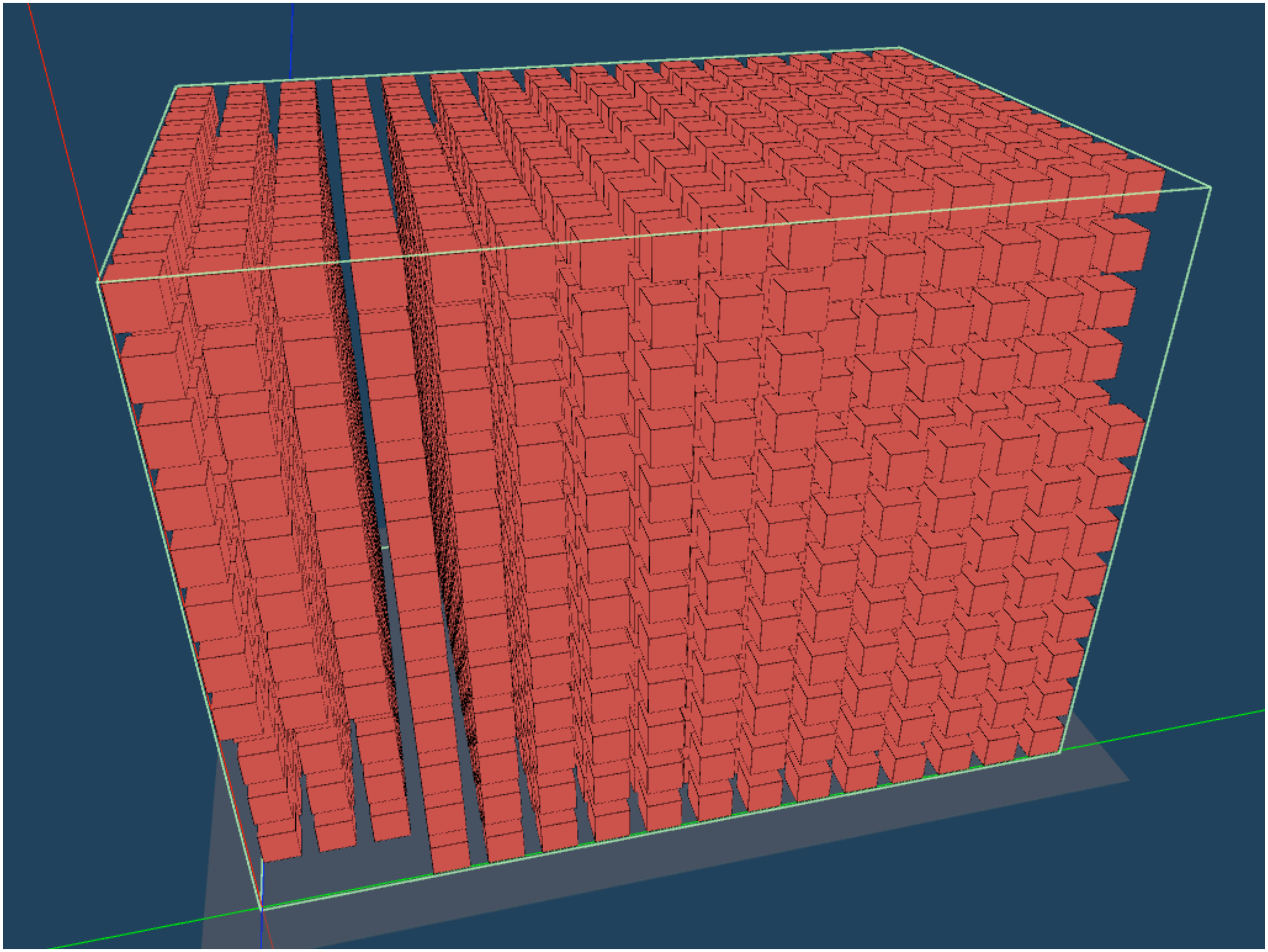
SMW = System Mgmt Workstation

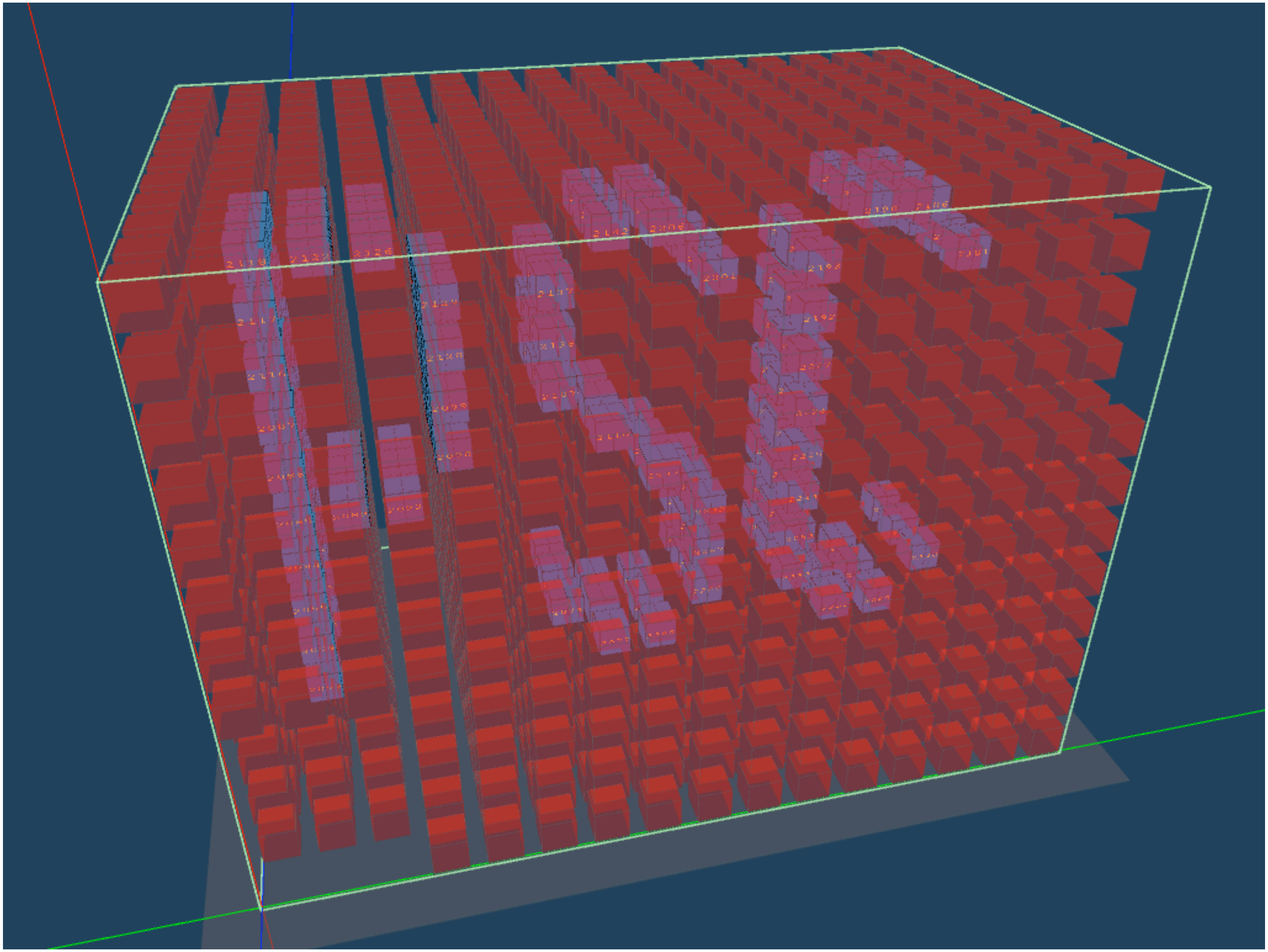
Full machine is 11x12x16

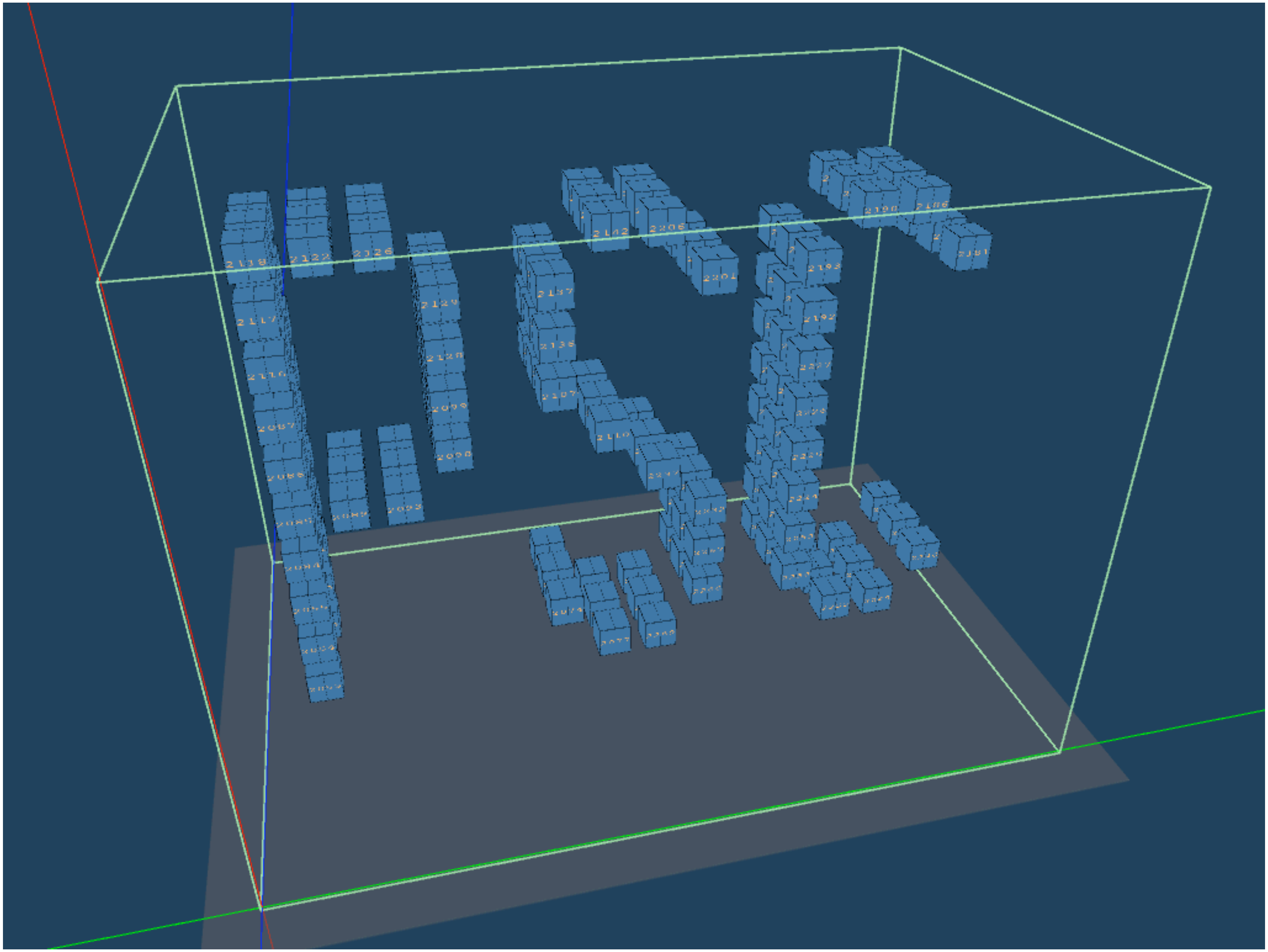
Illustration courtesy Cray, Inc.

Previous Work

- 2005, *Batch Scheduling on the Cray XT3*
  - Batch system on harness system
- 2006, *XT3 Operational Enhancements*
  - Batch system with CPA
  - Graphical monitor






2118
2122
2126
2117
2116
2087
2086
2129
2128
2099
2098
2137
2136
2107
2110
2142
2206
2201
2190
2186
2181
2193
2192

Optimizing Job Placement

- 2006, *Optimizing Job Placement on the XT3*, Weisser et al.
- Quantified effects of job layout on communication intensive codes:

App	P	Placed: Default	Placed: Opt	Prod Ave	Prod Std Dev	Opt vs. Prod Ave
PTRANS	1024	129.3 gb/s	146.5 gb/s	131.2 gb/s	21.2 gb/s	11.7%
DNSmsp	512	316.5 s	296.0 s	310.5 s	9.2 s	4.7%
DNSmsp	192	198.0 s	163.0 s	181.7 s	23.5 s	**10.3%**
NAMD	512	161.4 s	150.1 s	167.1 s	13.1 s	9.8%
NAMD	32	252.7 s	228.3 s	252.0 s	12.2 s	9.4%

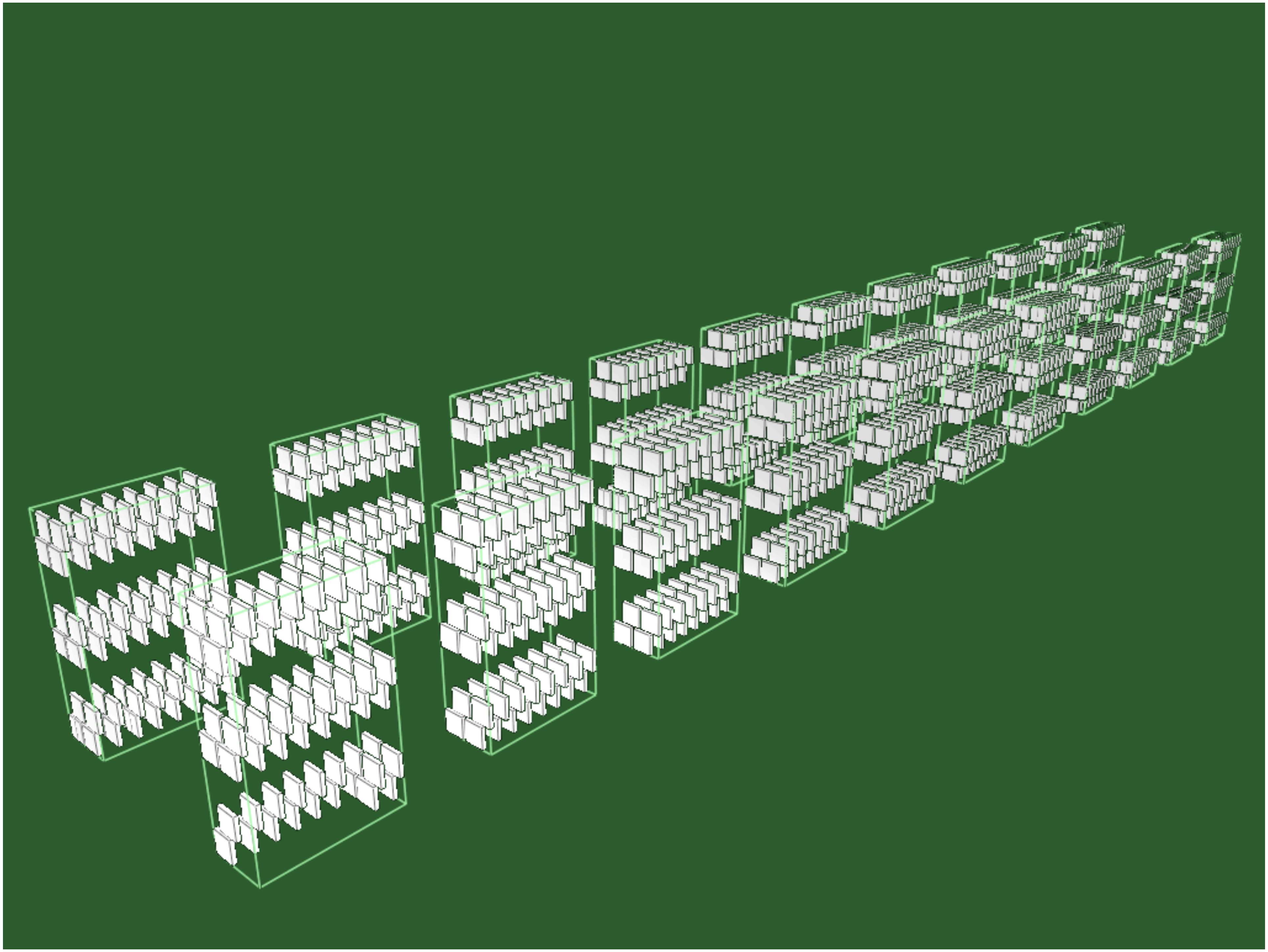
DNSmsp – numerical sim. Code for analysis of turbulence; NAMD – MD code

Outcome

- Jobs assigned in cubes or near-cubes will have lowest communication contention
  - Contiguous is good!
- Assign jobs to processors in directly connected cabinets
  - Assign down one row in X-major order

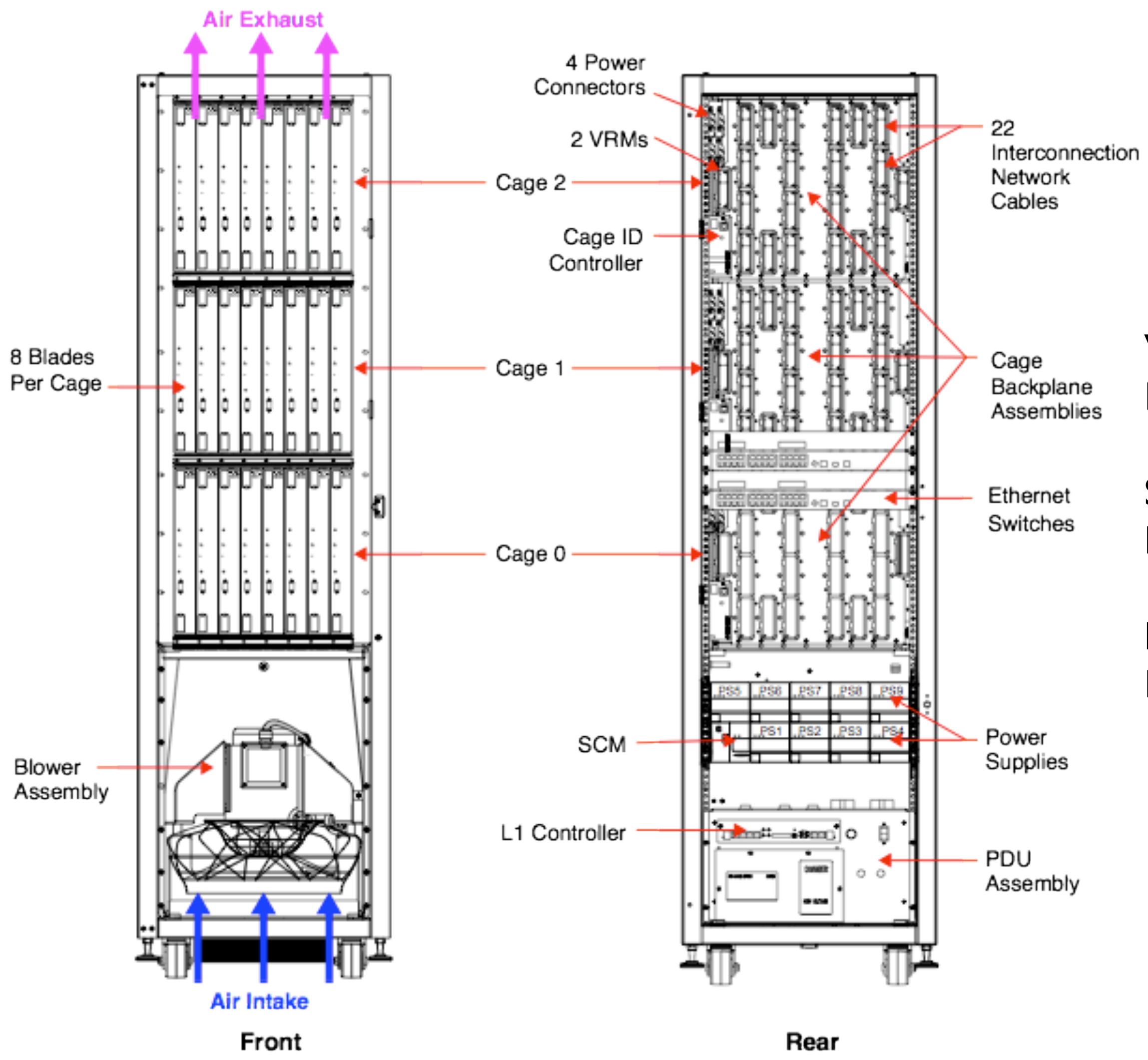
    0 – 2 – 4 – 6 – 8 – 10 – 9 – 7 – 5 – 3 – 1 – 0 …
  - Then down other row

CRAY XT3



CRAY XT3
COMPAQ

Cray XT3 Compute Cabinet

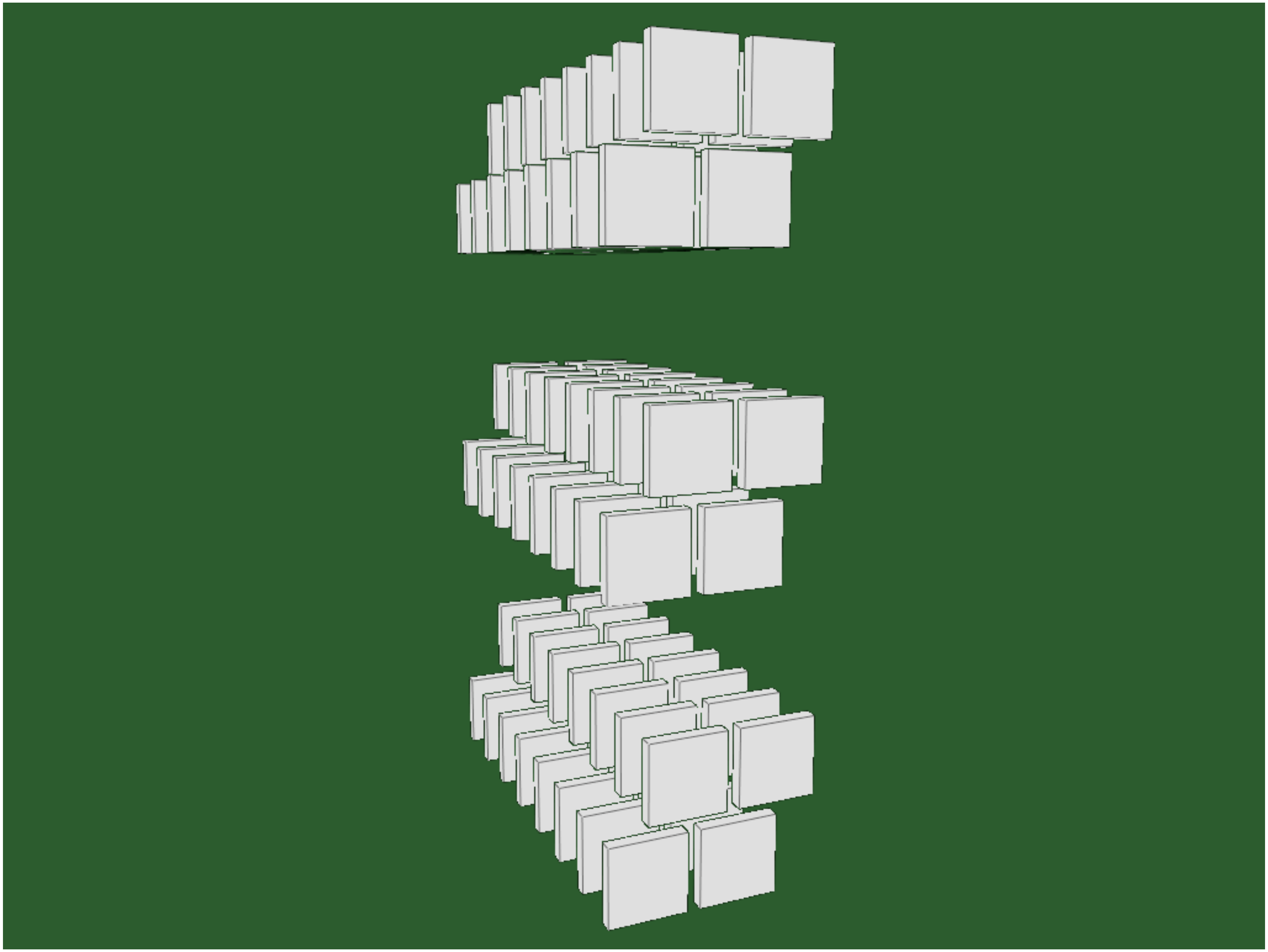


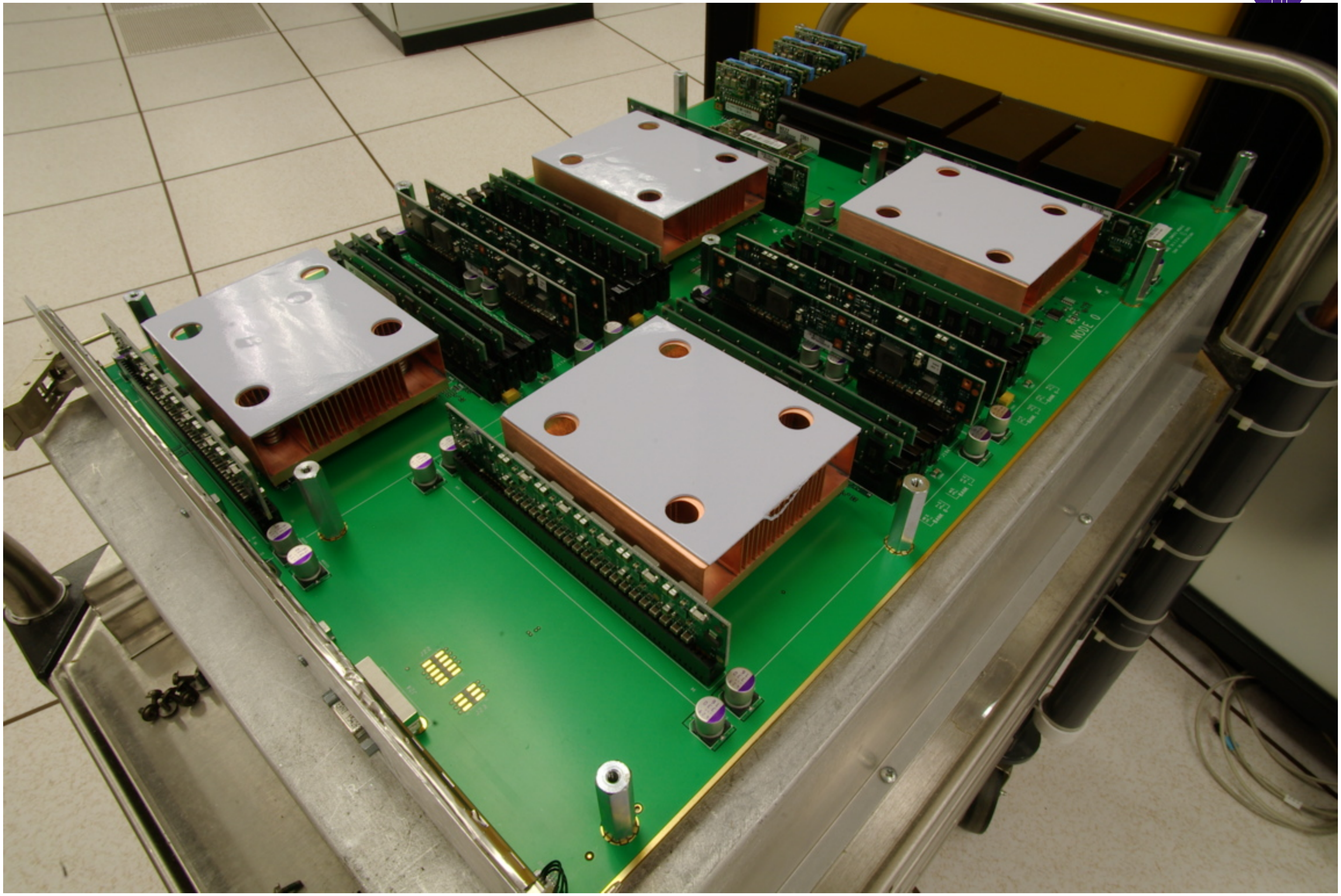
VRM = Voltage Regulator Module

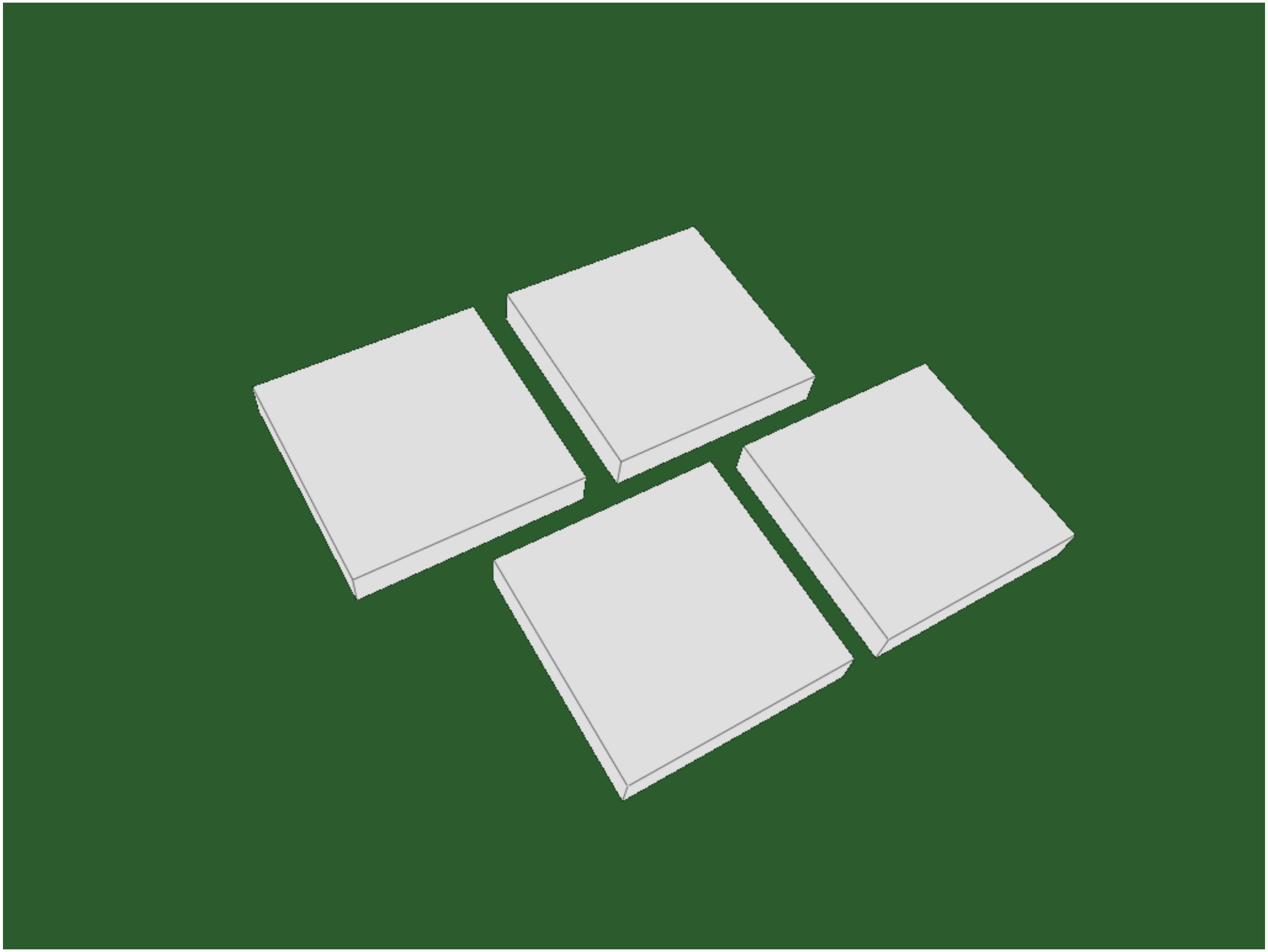
SCM = Shelf Controller Module

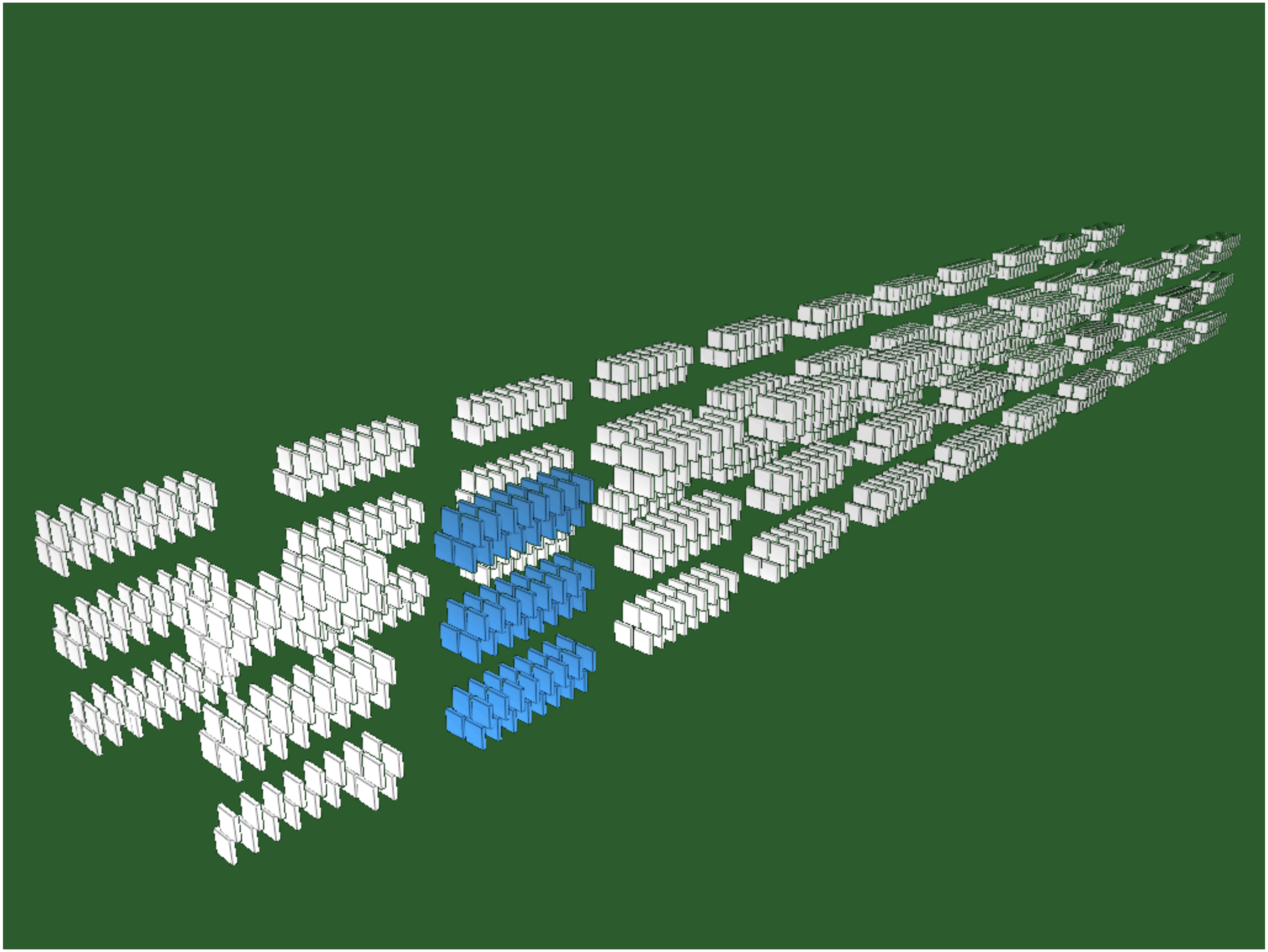
PDU = Power Distribution Unit

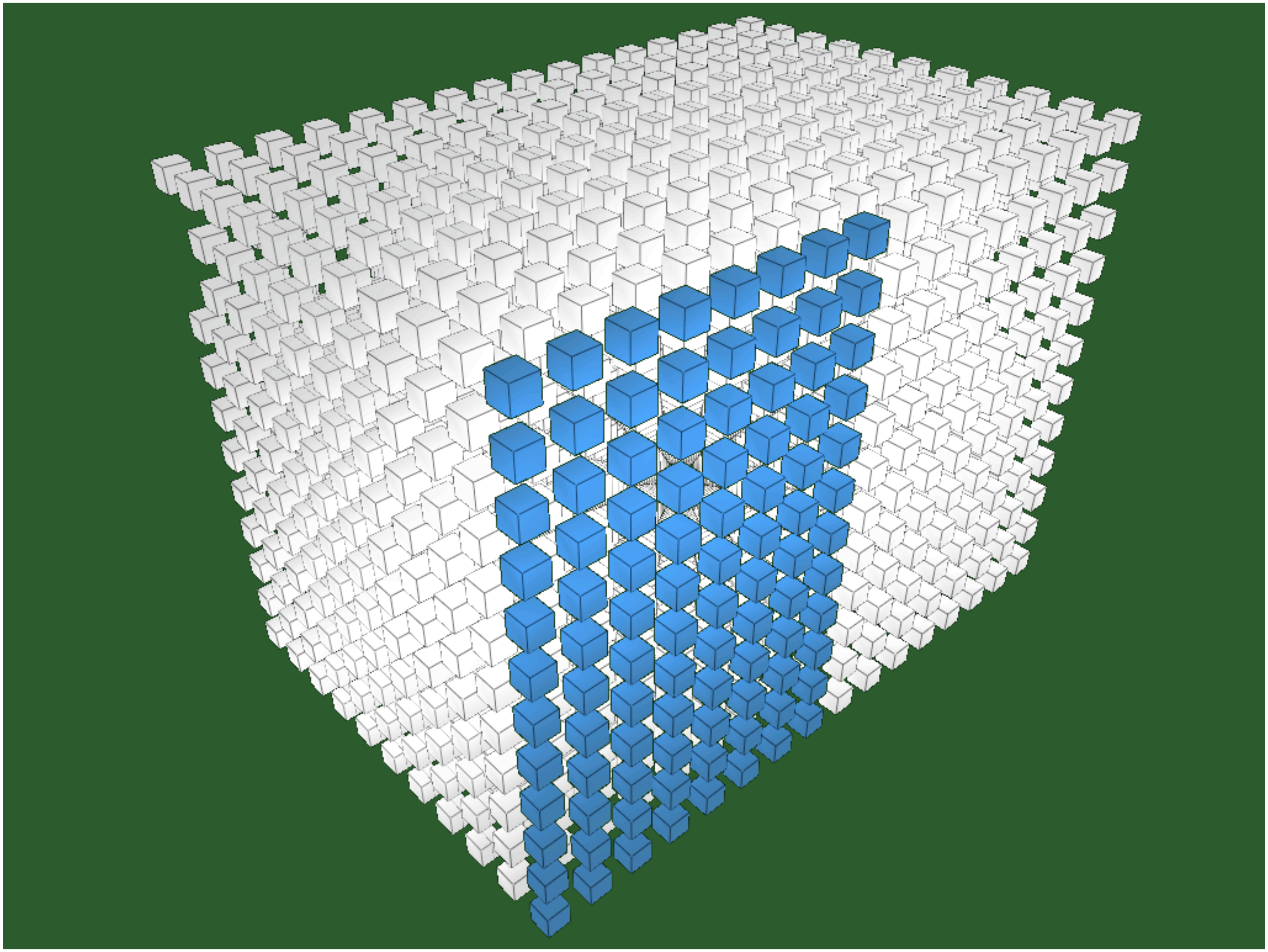
Illustration courtesy Cray, Inc.

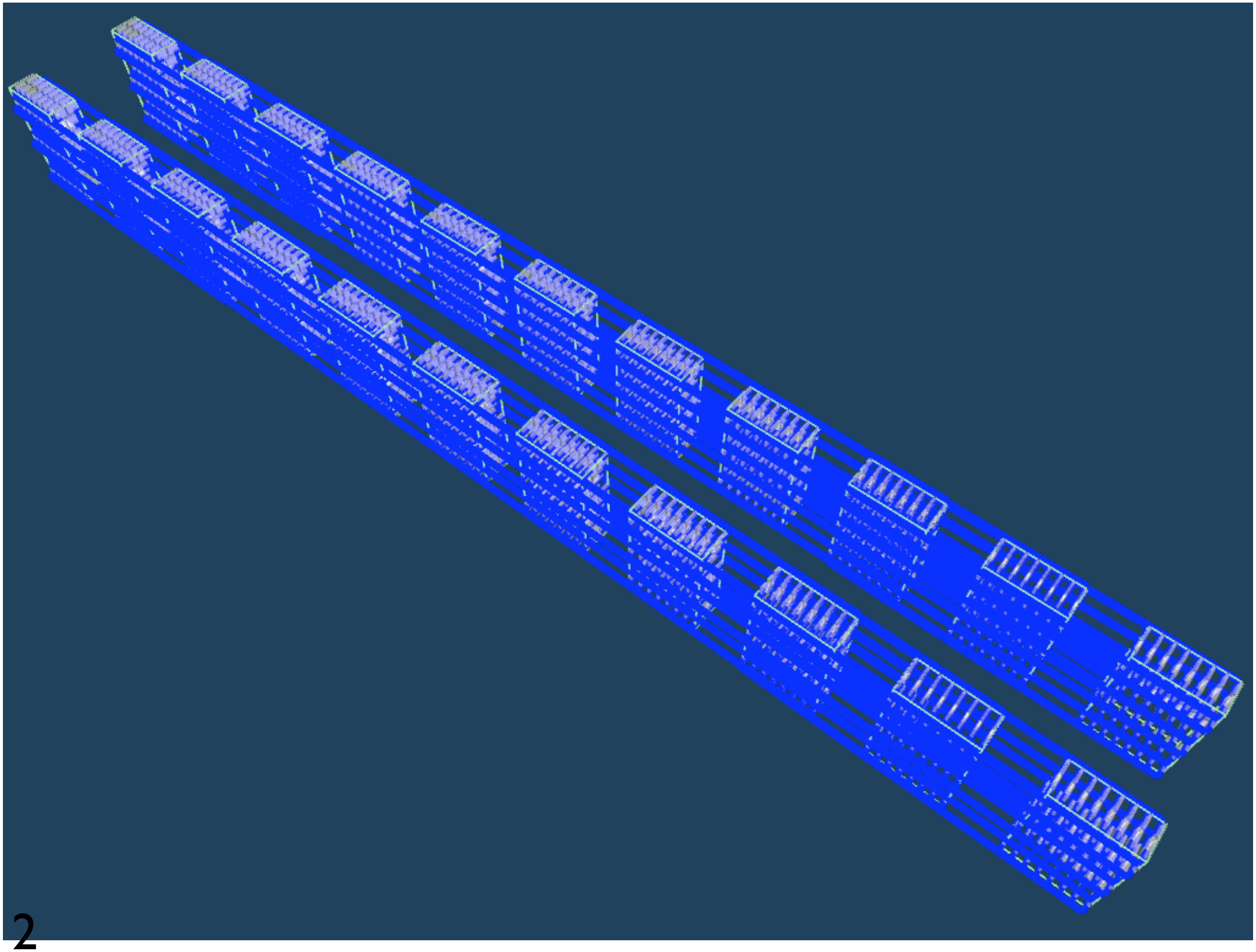


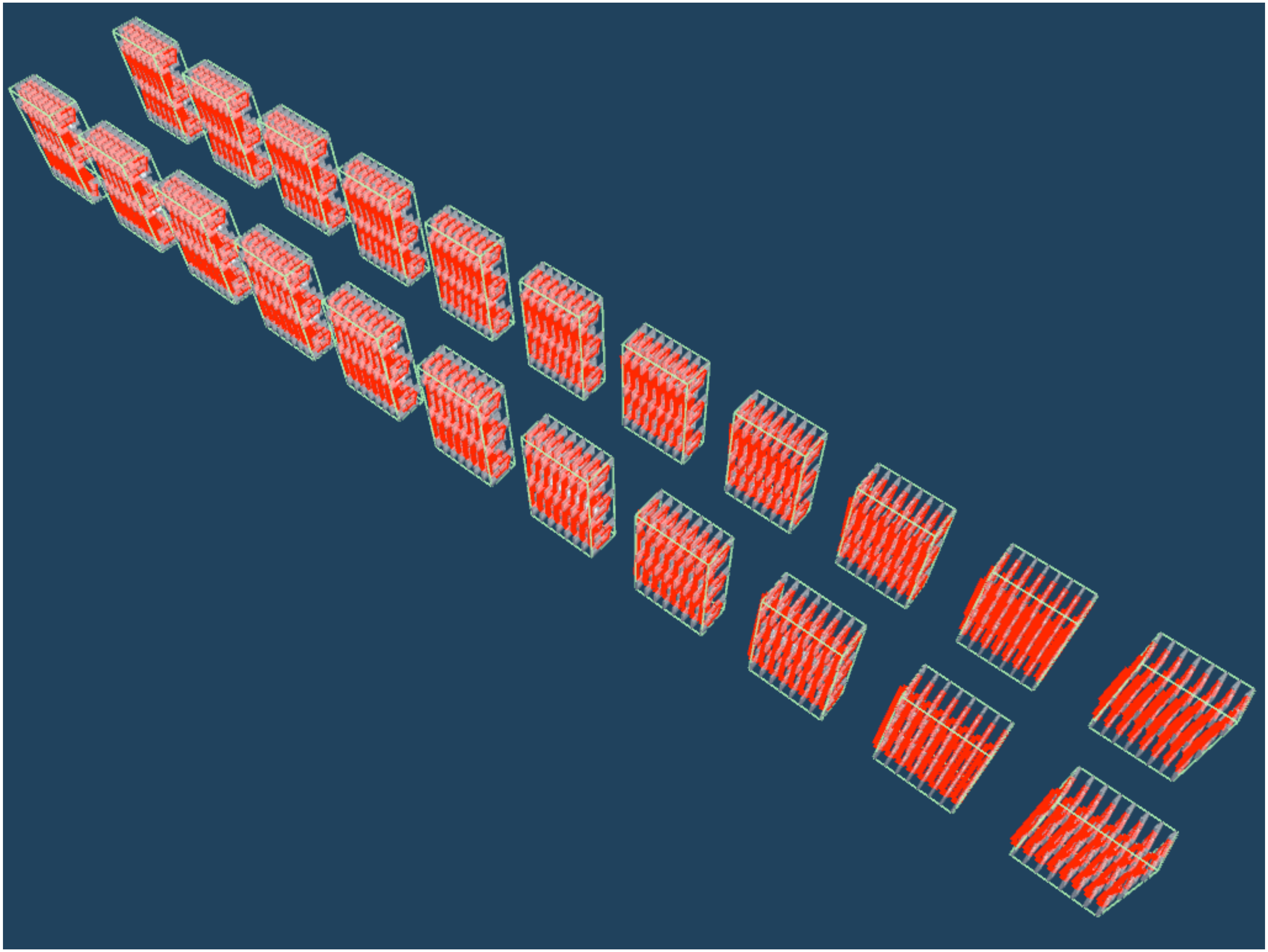


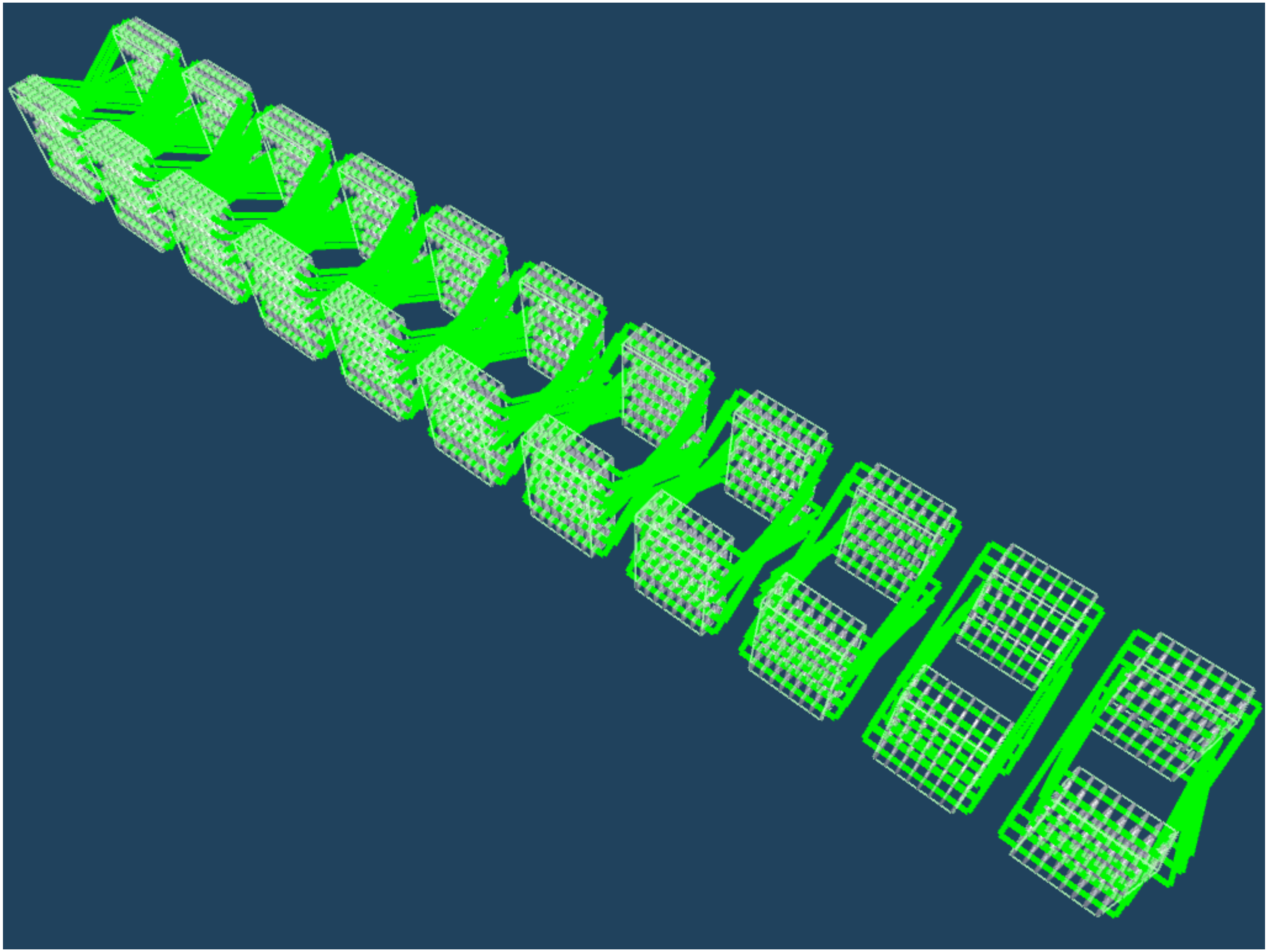


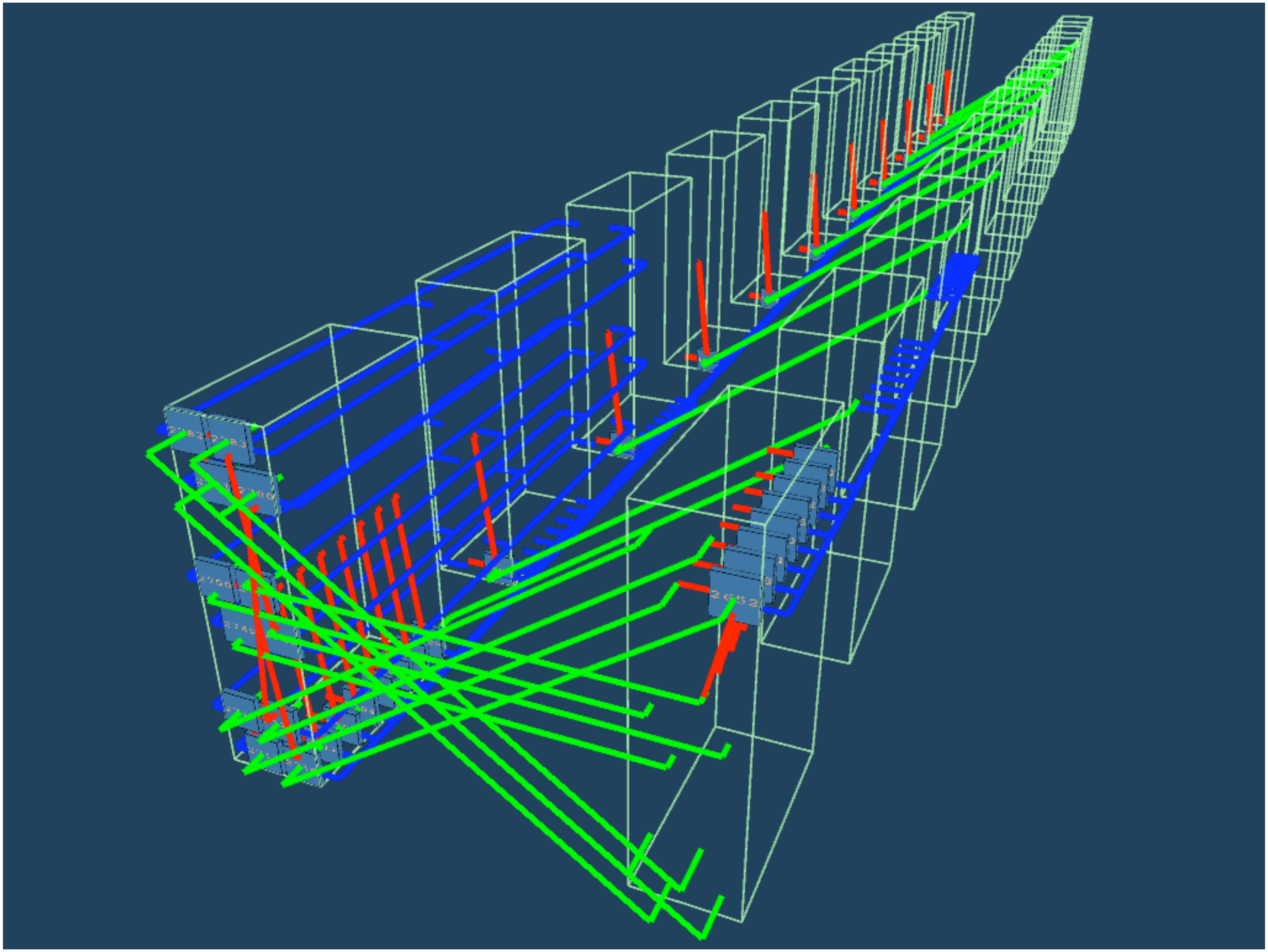











2782
2783
2780
2750
2749
2652

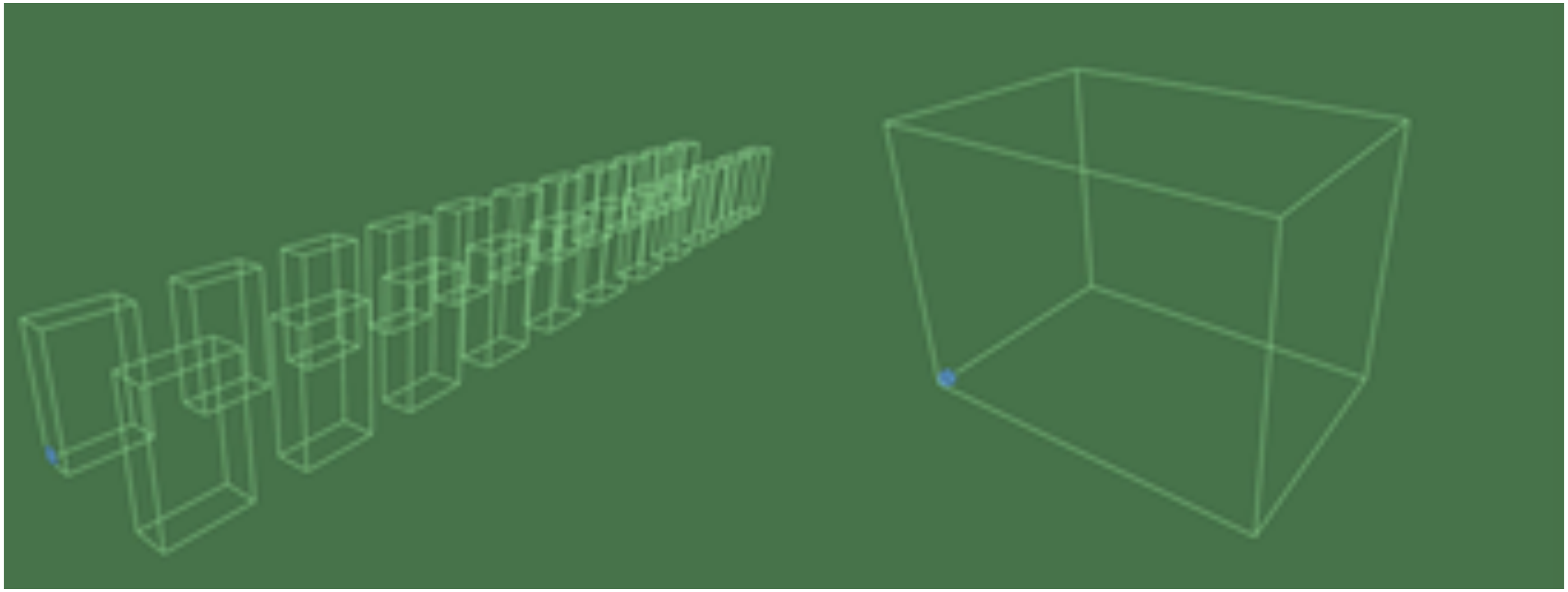
Outcome

- Jobs assigned in cubes or near-cubes will have lowest communication contention
  - Contiguous is good!
- Assign jobs to processors in directly connected cabinets
  - Assign down one row in X-major order
    - 0 – 2 – 4 – 6 – 8 – 10 – 9 – 7 – 5 – 3 – 1 – 0 …
  - Then down other row

System changes to benefit all jobs

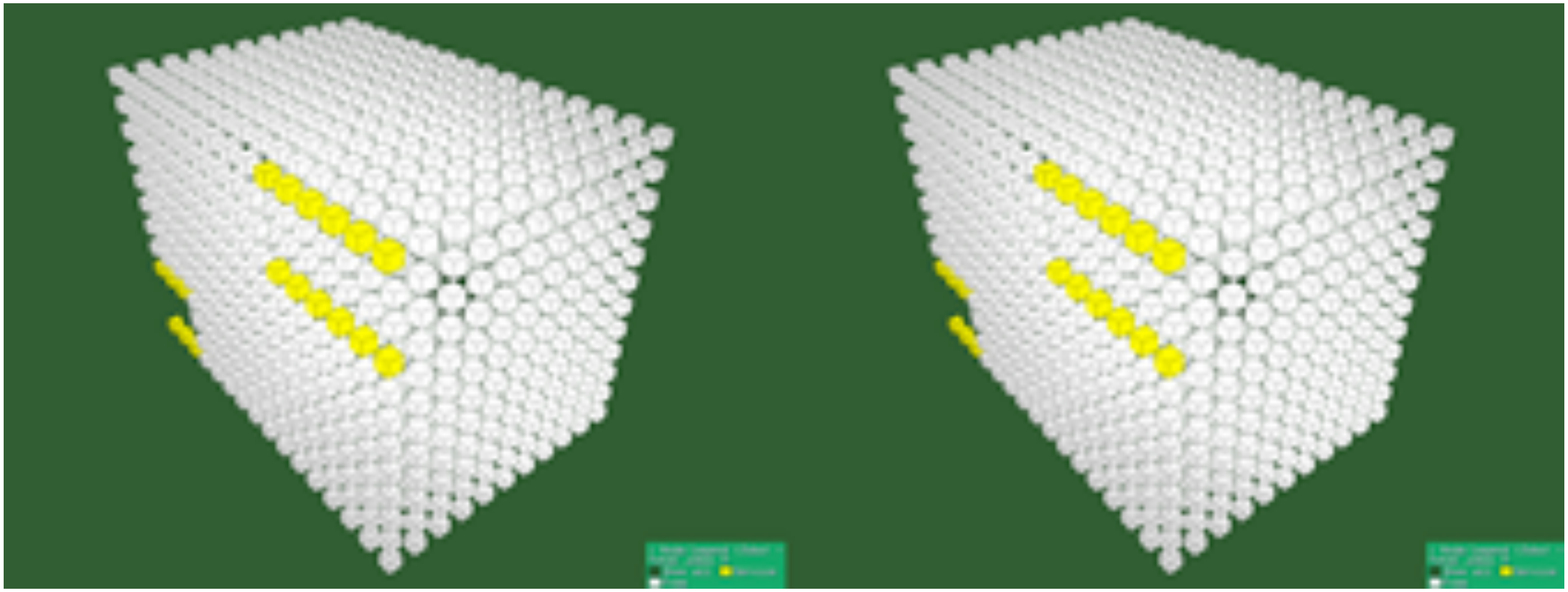
- Code scheduler to assign free nodes according to predefined “optimal” order
- Order free list according to an “optimal” order mask
  - Doesn’t slow down scheduler
  - Fragmentation
- Pass list to CPA via pbs_mom
  - CPA does not get to do node selection

Optimal order mask



[See
http://staff.psc.edu/vizino/cug2008/opt_nid_list_mask.mov
for animation.]

Optimized vs. Numeric Order



[See http://staff.psc.edu/vizino/cug2008/cubic_vs_planar.mov for animation.]

What if users know what they want?

- How do you give users a shape?
  - System issues
    - Batch system
    - Node ids
- How can they tell what they got?

OpenAtom

- *Improving parallel scaling performance using topology-aware task mapping on Cray XT3 and IBM Blue Gene/L*, Bhatele, Bohm and Kale
- Molecular Dynamics code
- Benchmarks used to measure OpenAtom
  - WATER_32M_70Ry
  - WATER_256M_70Ry

OpenAtom Results

- Performance of OpenAtom
- Using MD WATER benchmark
  - Single core per node
  - Times represent time per step in seconds

Processors	32M_70Ry Default	32M_70Ry Topology	% Speedup	256M_70Ry Default	256M_70Ry Topology	% Speedup
512	0.124	0.123	1	5.90	5.37	9
1024	0.095	0.078	**18**	4.08	3.24	**20**

OpenAtom Results

- Using WATER benchmark
  - Two cores per node
  - Times represent time per step in seconds

Processors	32M_70Ry Default	32M_70Ry Topology	% Speedup	256M_70Ry Default	256M_70Ry Default	% Speedup
256	0.226	0.196	13	-	-	-
512	0.179	0.161	11	7.50	6.58	12
1024	0.144	0.114	21	5.70	4.14	27
2048	0.135	0.095	**29**	3.94	2.43	**38**

OpenAtom

- Provide specific shapes:
  - 8x8x16=1024, 8x8x8=512, 8x8x4=256, 8x4x4=128
- Provide node id/dimension mapping

System issues – First Pass

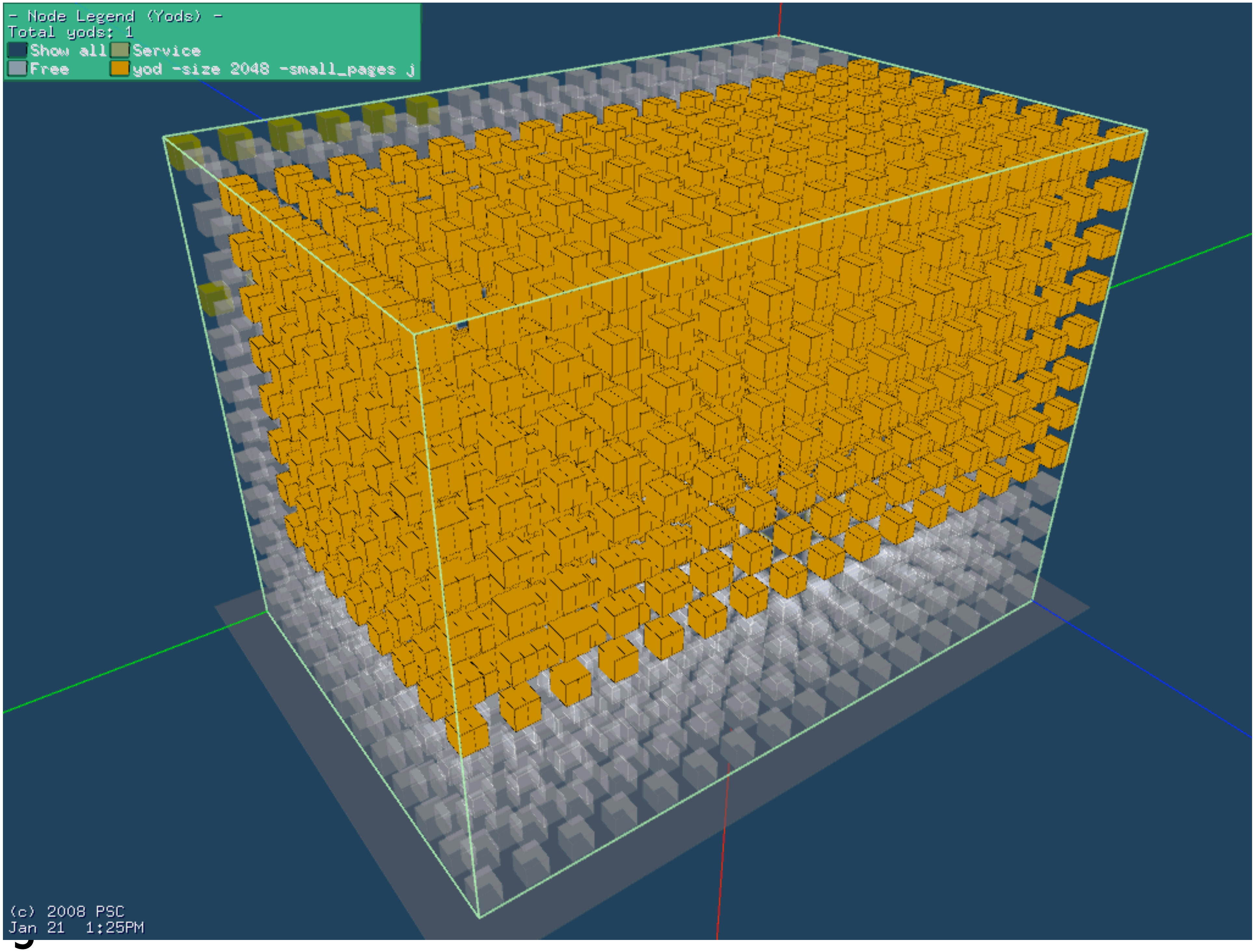
- Provide 4 shape reservations and transition as needed
- Problems
  - Time consuming
  - Admin in the loop while running

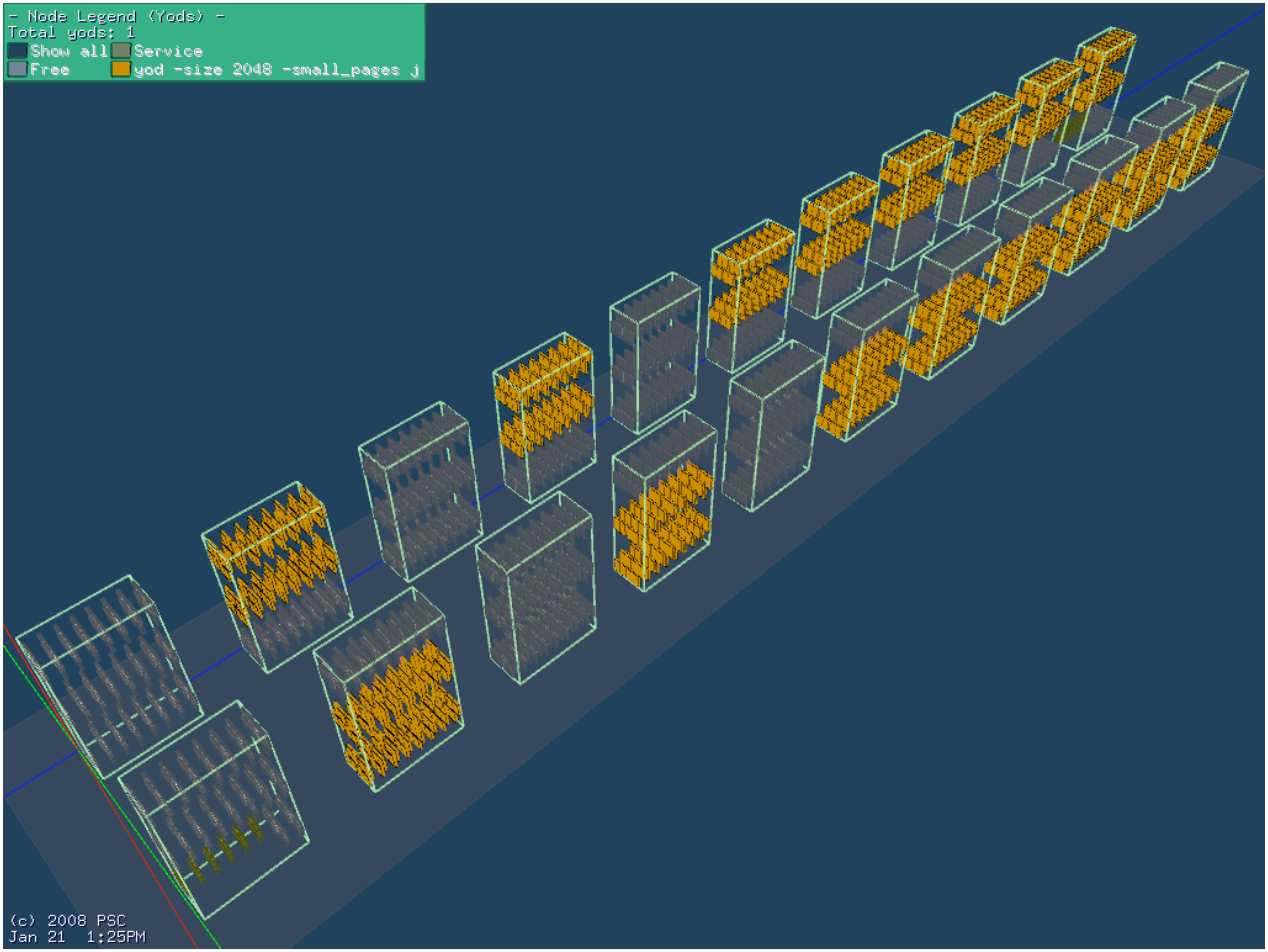
System Issues – Second Pass

- Provide 1 shape reservation
  - 8x8x16=1024
- Provide nid shape lists to yod
  - Developed script to get shape for reservation
    - From SDB
  - yod –list `cat /usr/local/shapes/XxYxZ.rl` …

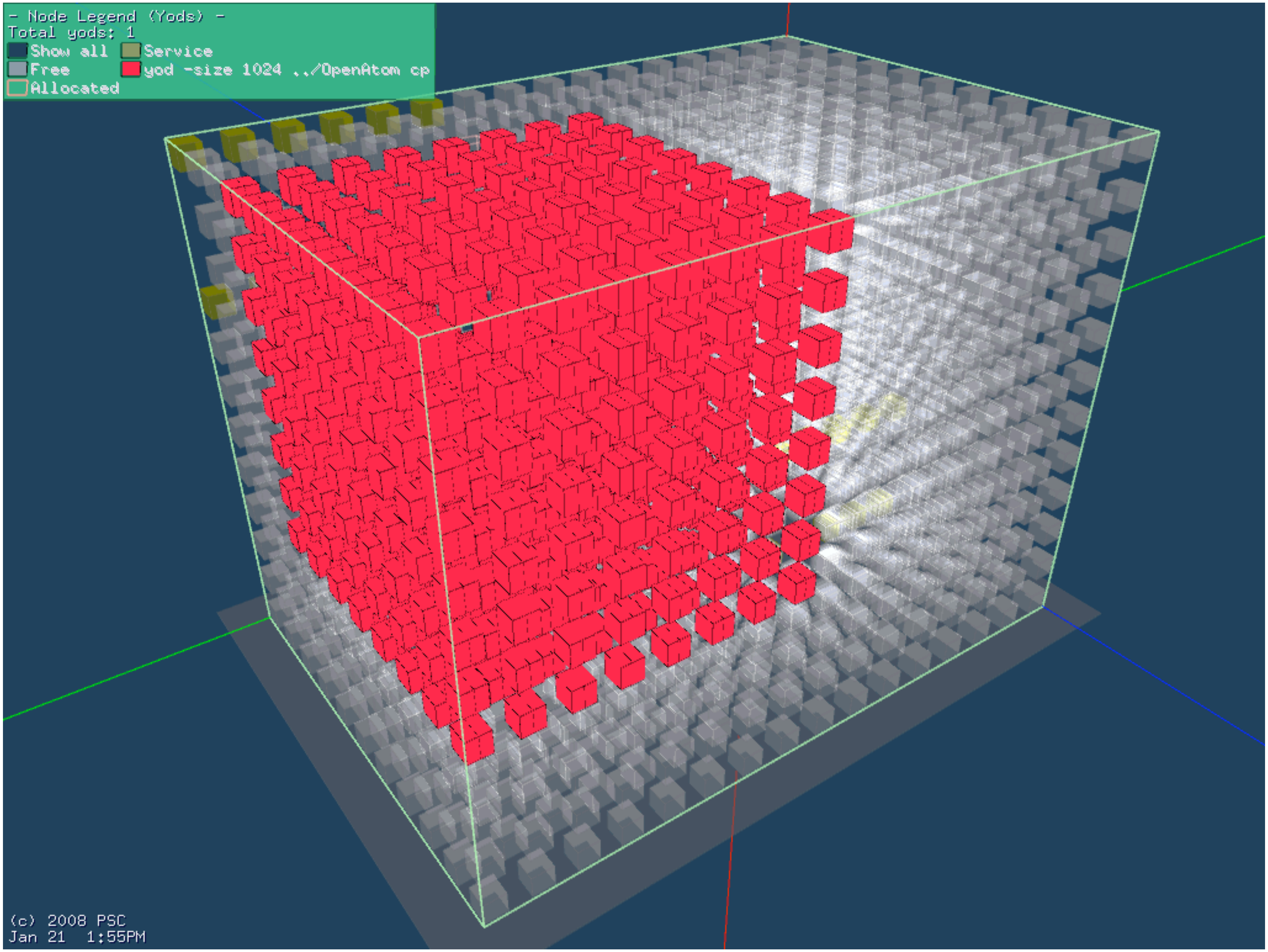
Roadblocks

- Topology information scarce on XT3
  - int rca_get_meshcoord(uint16_t nid, rca_mesh_coord_t *xyz);
- Nodes can be down so must pick around them or not run at all

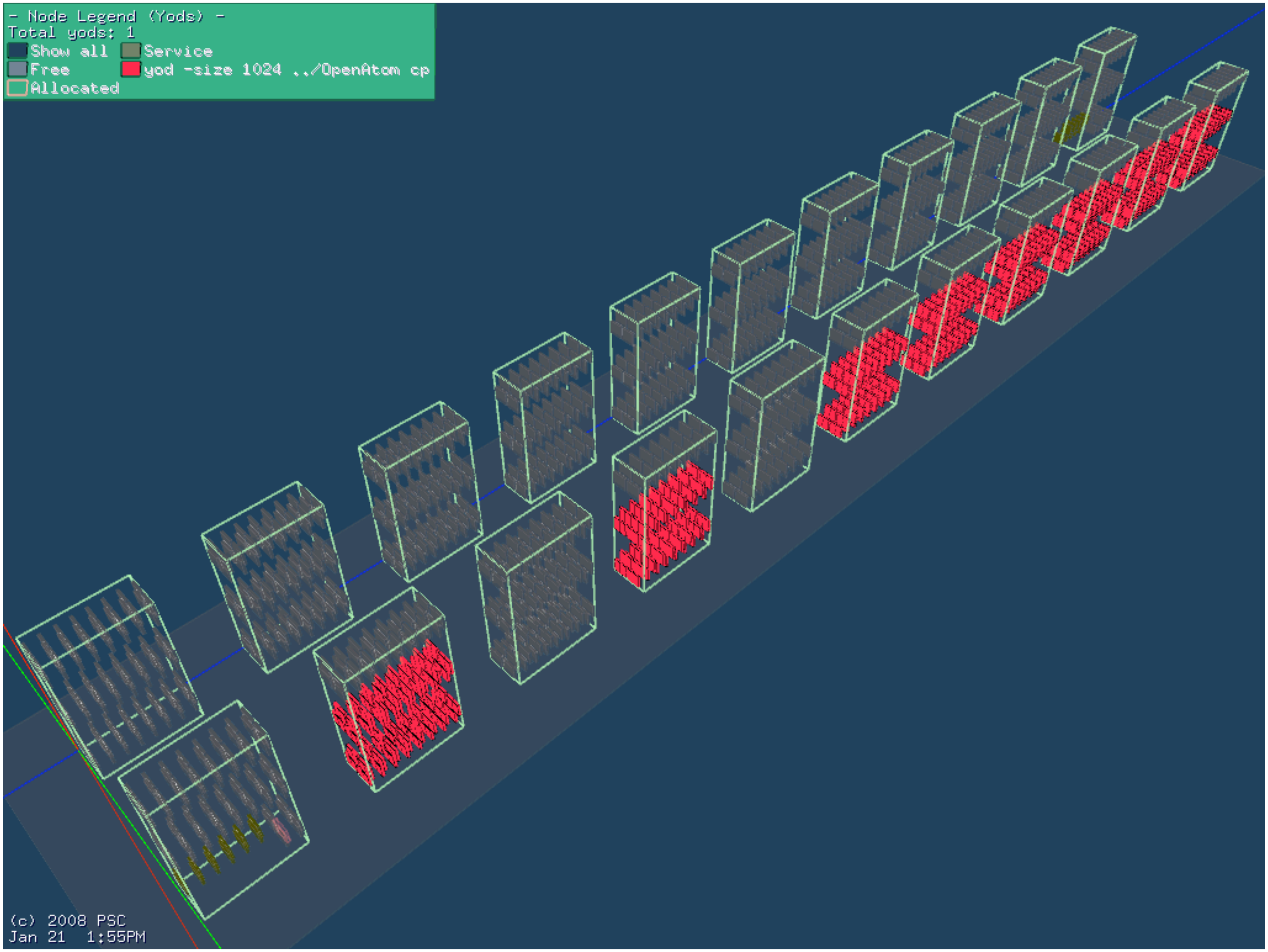

- Node Legend (Yods) -
Total yods: 1
Show all
Service
Free
yod -size 2048 -small_pages j
(c) 2008 PSC
Jan 21 1:25PM



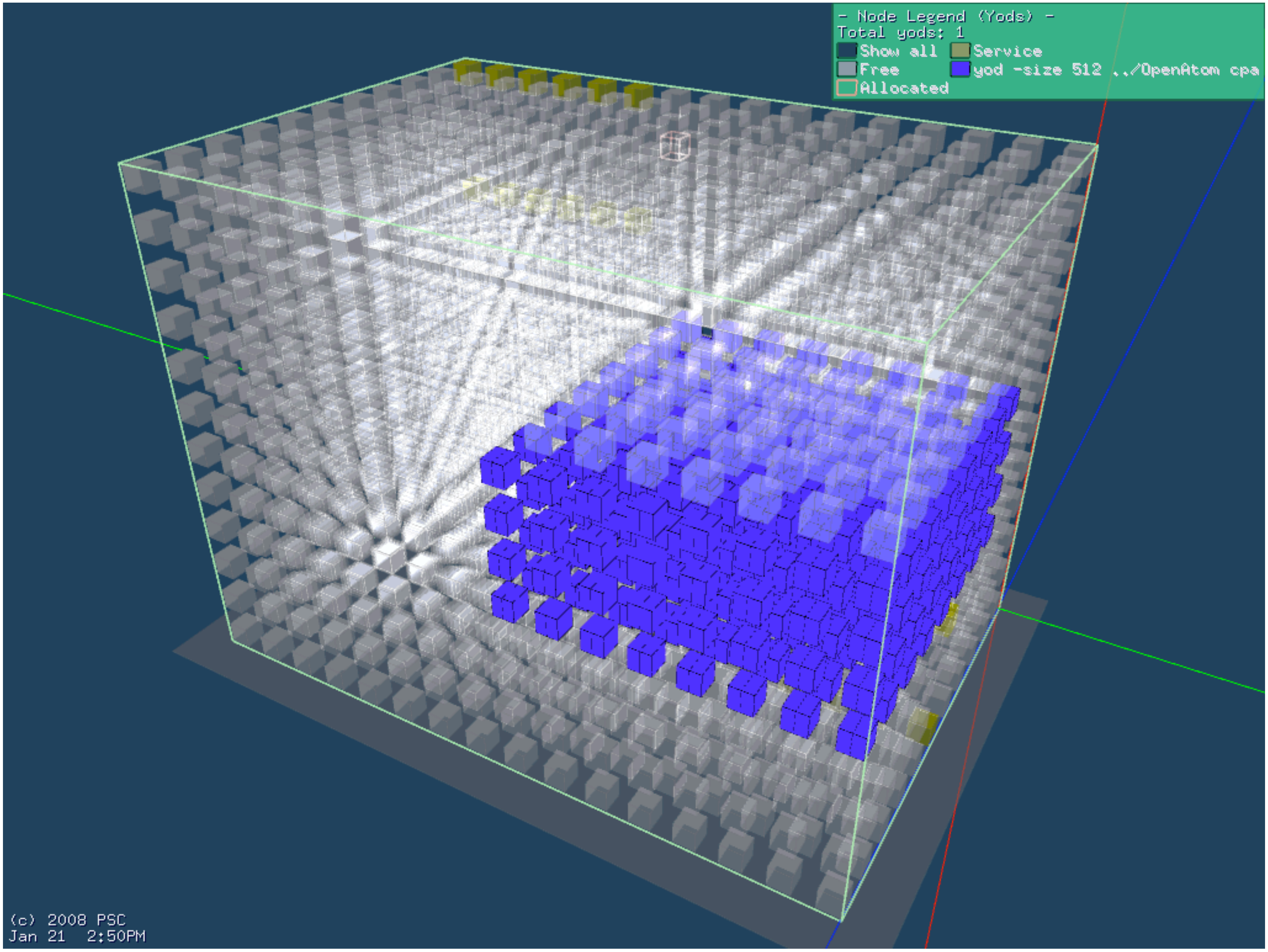
- Node Legend (Yods) -
Total yods: 1
Show all
Service
Free
yod -size 2048 -small_pages j
(c) 2008 PSC
Jan 21 1:25PM

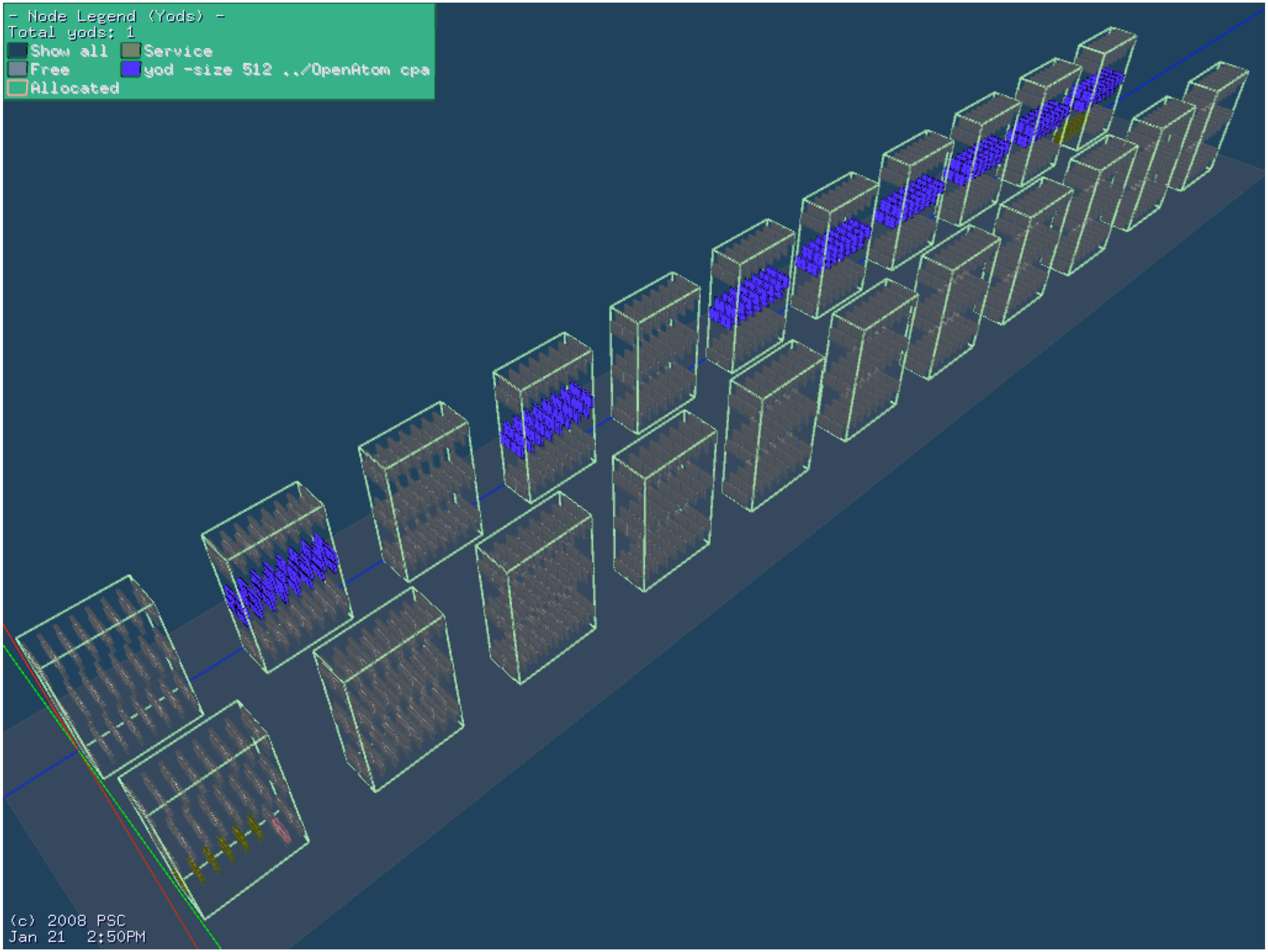


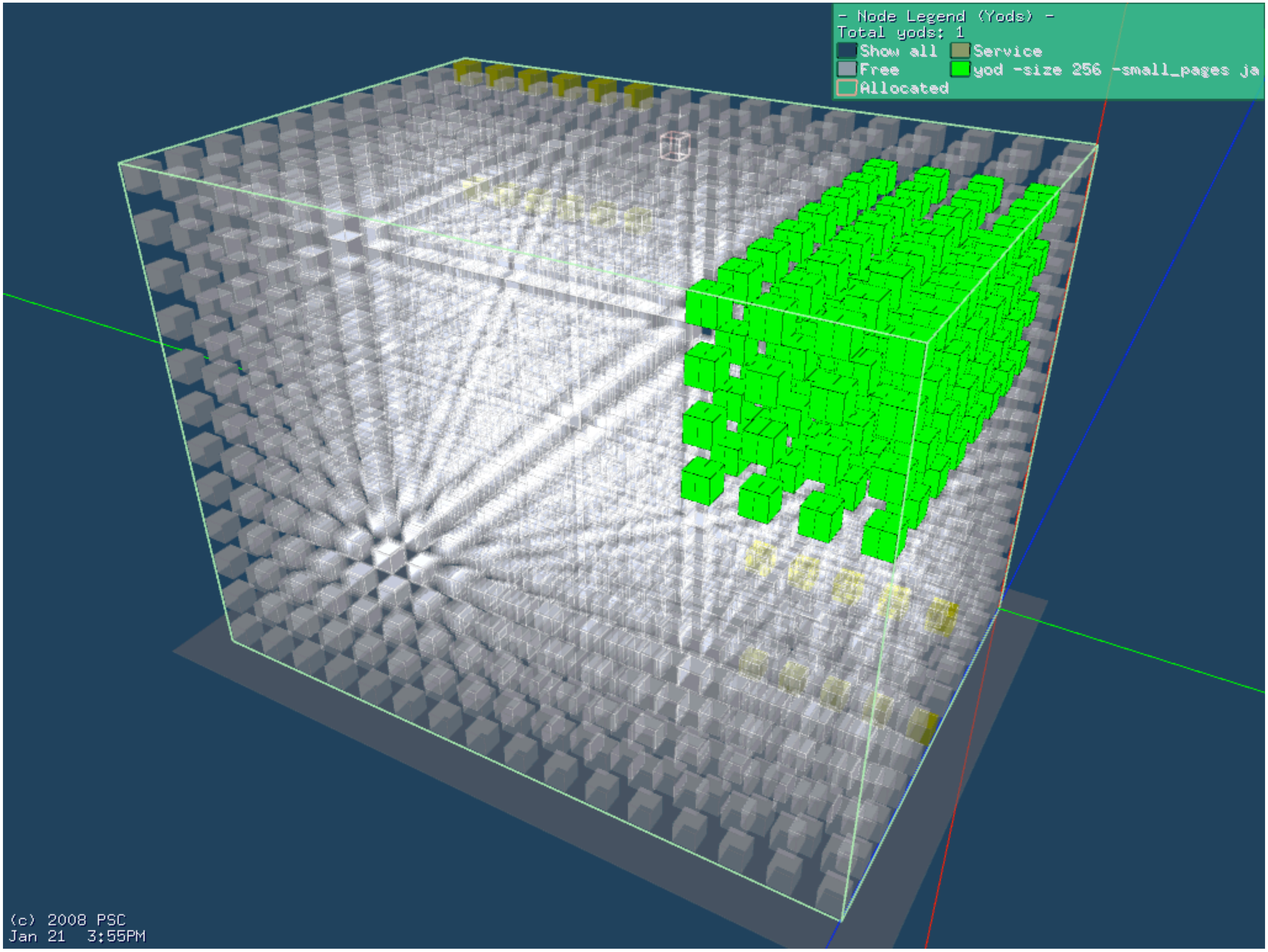
- Node Legend (Yods) -
Total yods: 1
Show all
Service
Free
yod -size 1024 ../OpenAtom cp
Allocated
(c) 2008 PSC
Jan 21 1:55PM

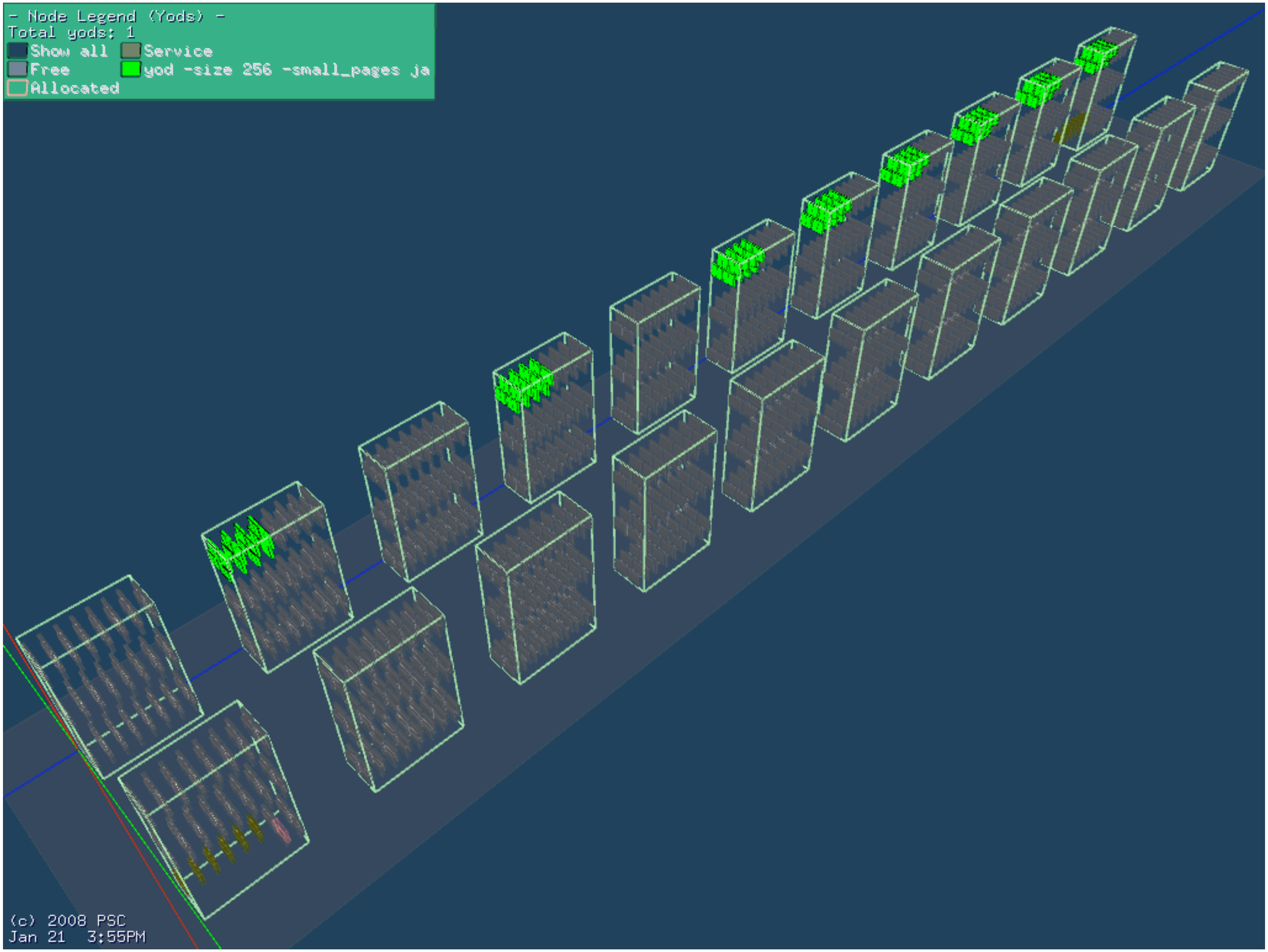


- Node Legend (Yods) -
Total yods: 1
Show all
Service
Free
yod -size 1024 ../OpenAtom cp
Allocated
(c) 2008 PSC
Jan 21 1:55PM


- Node Legend (Yods) -
Total yods: 1
Show all
Service
Free
yod -size 512 ../OpenAtom cpa
Allocated
(c) 2008 PSC
Jan 21 2:50PM


- Node Legend (Yods) -
Total yods: 1
Show all
Service
Free
yod -size 512 ../OpenAtom cpa
Allocated
(c) 2008 PSC
Jan 21 2:50PM


- Node Legend (Yods) -
Total yods: 1
Show all
Service
Free
yod -size 256 -small_pages ja
Allocated
(c) 2008 PSC
Jan 21 3:55PM


- Node Legend (Yods) -
Total yods: 1
Show all
Service
Free
yod -size 256 -small_pages ja
Allocated
(c) 2008 PSC
Jan 21 3:55PM

OpenAtom Conclusions

- 20% speedup for single core
- 38% speedup for dual core
- "…difficulties in obtaining topology information on XT3…"
- "…project that topology-aware mapping should yield improvements proportional to torus size on larger XT3 or XT4 installations."

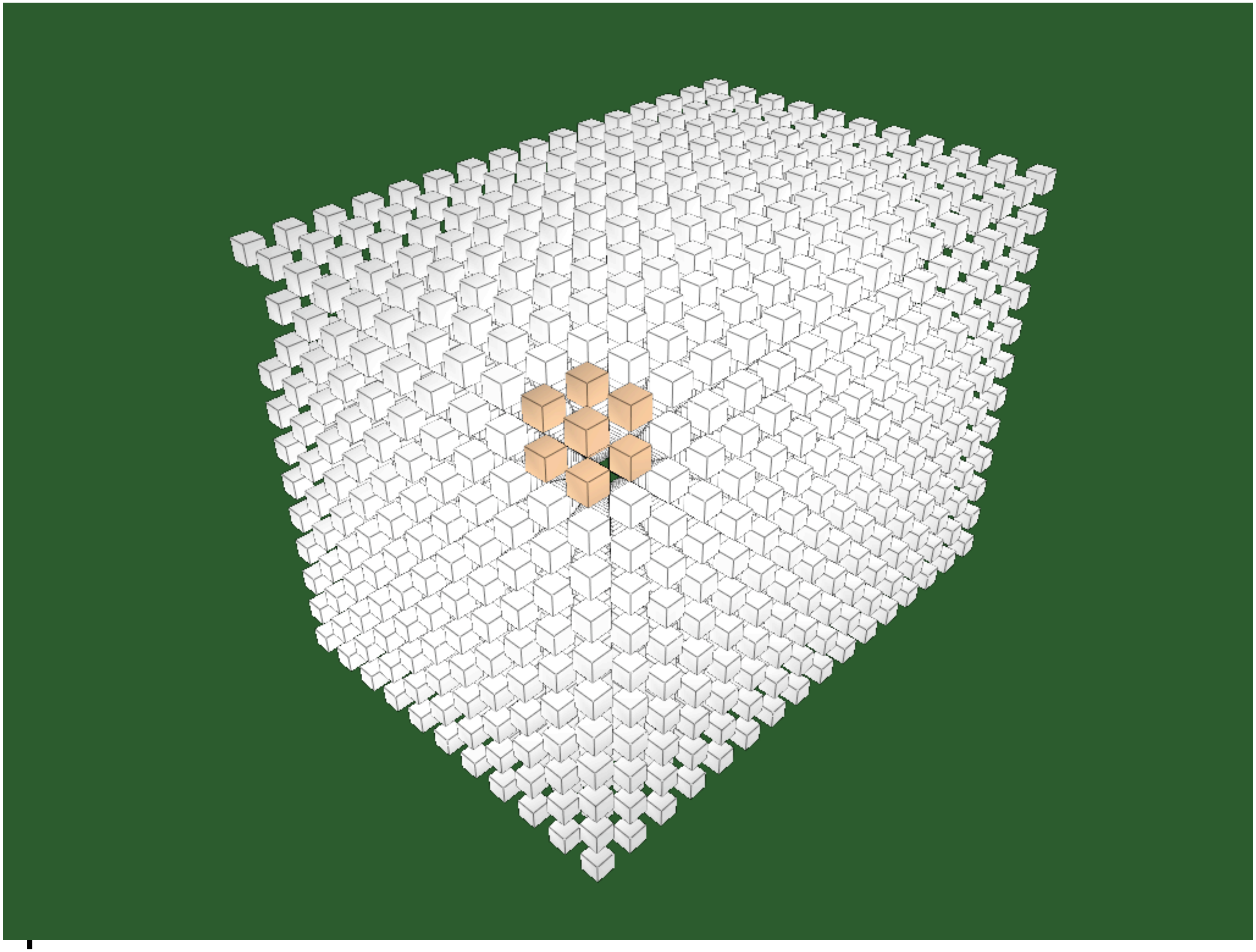
vizino@psc.edu

Another Topo-aware Case

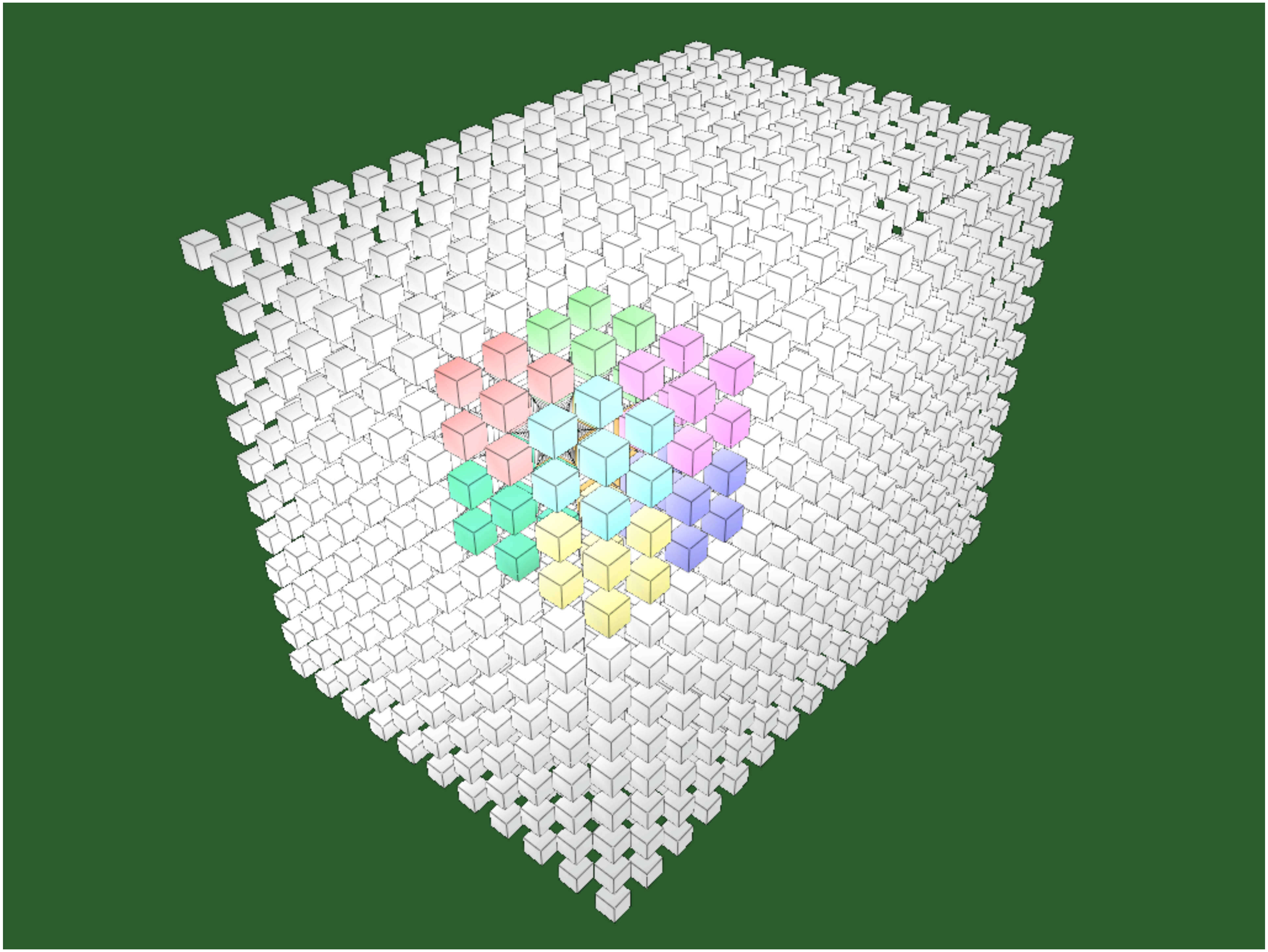
- Adam Liwo, Czarek Czaplewski, Stan Oldziej, and Harold Scheraga
- Molecular dynamics force field simulator for predicting the structure and properties of proteins
- No reference to cite yet

Molecular Dynamics Code

- Goal is to speed up simulations by running many in parallel
- Need to decompose code
  - Coarse-grain level
    - Infrequent communications
  - Fine-grain level
    - Frequent communications
    - Small number of cores



7



Future Work

- Gather more statistics on nid ordering benefit
- Deal with fragmentation issues
- Work into production model
  - Allow users to specify shape reservations
  - Figure out how to make topology information more accessible
- Test under Compute Node Linux

Conclusion

- Contiguous placement can influence code performance
- System changes have provided benefit to both regular and special needs jobs
- Cray provided way to get topology information would help our users