When to use 3D Die-Stacked Memory for Bandwidth-Constrained Big Data Workloads

Jason Lowe-Power || Mark D. Hill || David A. Wood

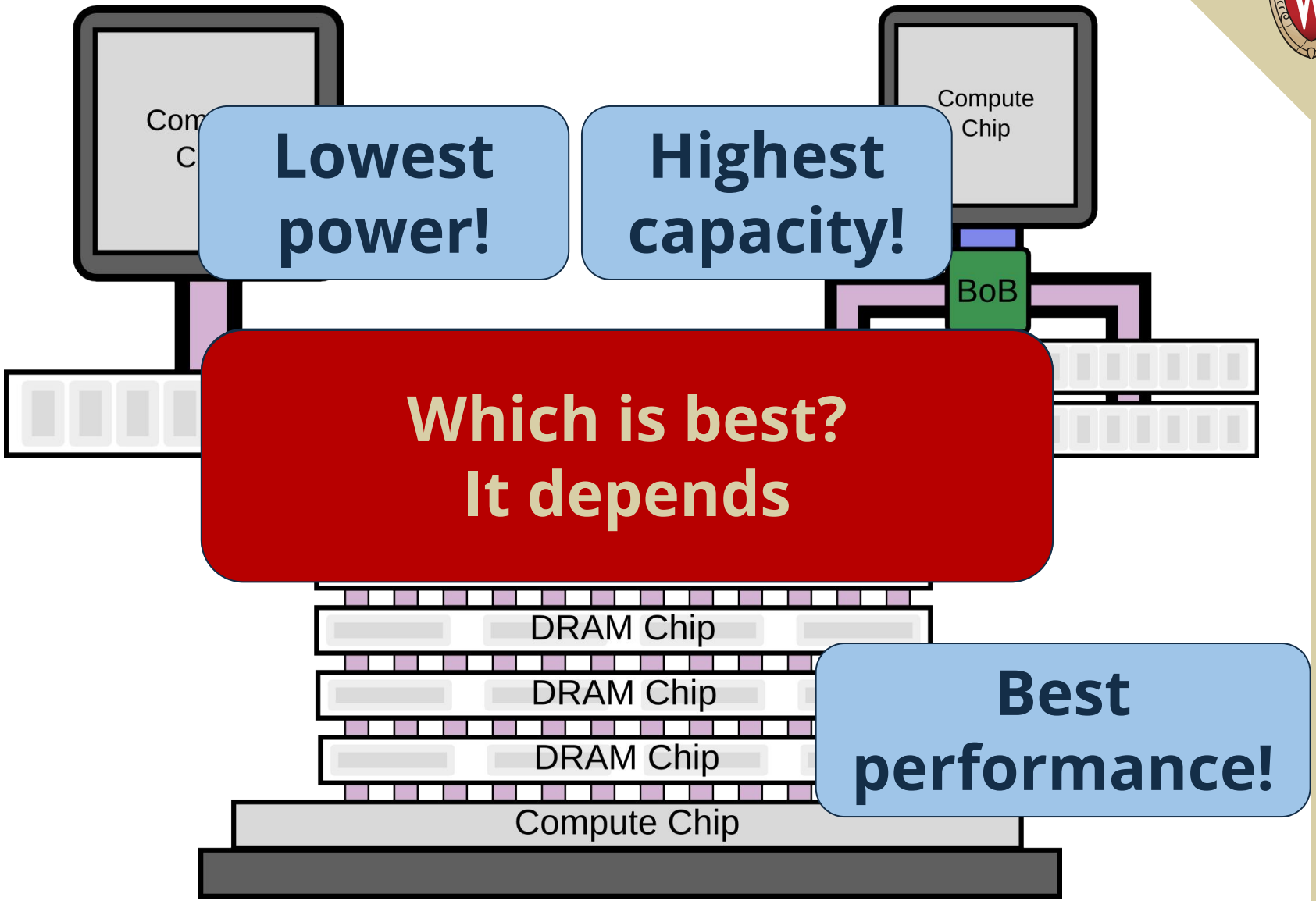
Big Data == Big Memory

GEODE

redis

Low latency → Real-time

What is the performance for 16 TB system?


Com
C
Compute
Chip
Lowest
power!
Highest
capacity!
BoB
Which is best?
It depends
DRAM Chip
DRAM Chip
DRAM Chip
Compute Chip
Best
performance!

Big Memory Machines

Dell PowerEdge R930

Memory capacity

3 TB (3,072 GB)

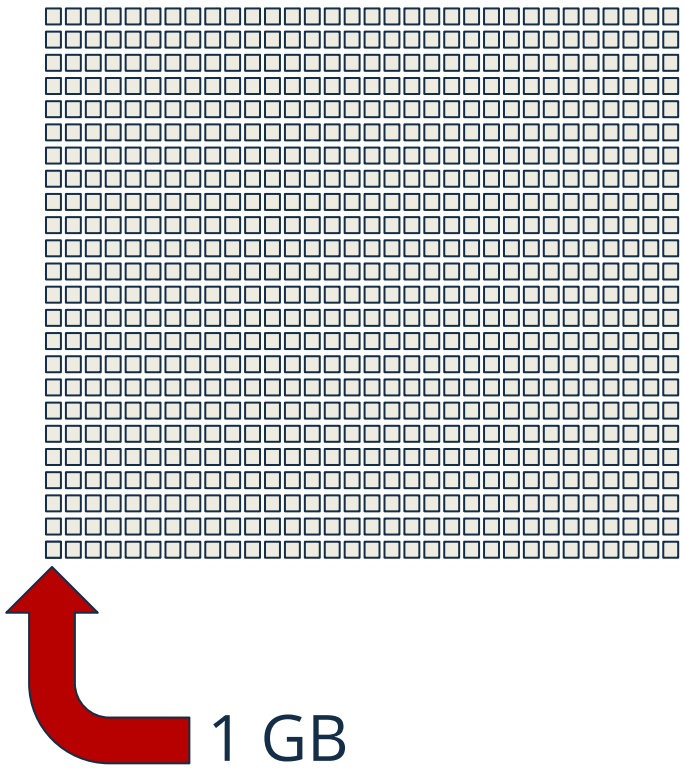
Memory bandwidth

408 GB/s

Processors

64 cores

DRAM (per socket)



1 GB

Amount accessible per second

Amount accessible in 10 ms

Processing 2x–10x faster than data supply

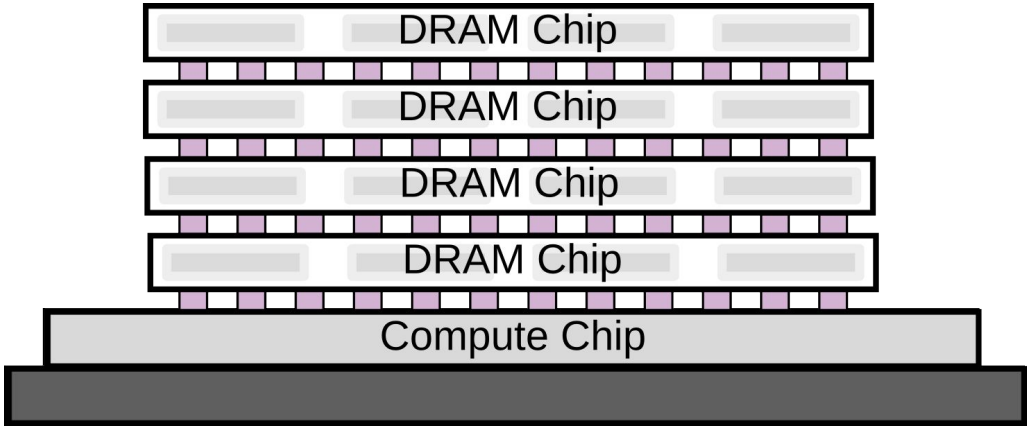
Amount accessible per second

CPU processing in 10 ms

Amount accessible in 10 ms

GPU processing in 10 ms

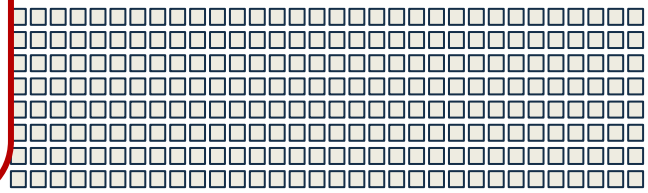
3D Die-Stacking



DRAM (per socket)

□□□□□□□□

Amount accessible per second

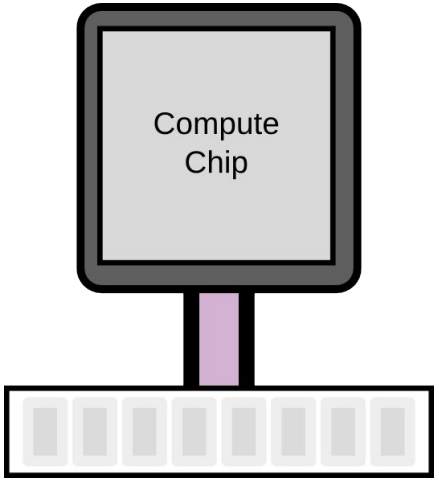


Amount accessible in 10 ms

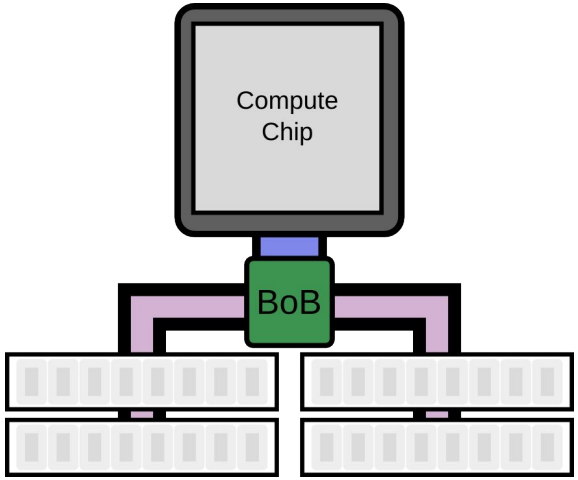
□□□

Data supply to data processing ≈1

Traditional Server



Big-Memory Server

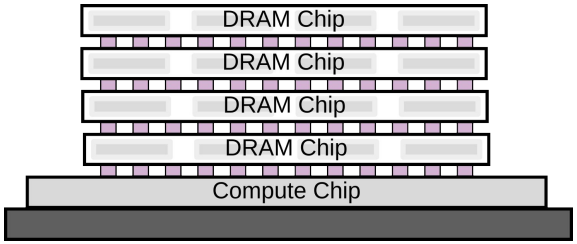


↑ **Higher bandwidth**

↑↑ **Higher capacity**

(compared to traditional)

Die-Stacked Server



Outline

Model and Workload

$$\text{mem modules} = \frac{\text{db size}}{\text{module capacity}} \quad (1)$$

$$\text{compute chips} = \left\lceil \frac{\text{mem modules}}{\text{mem channels}} \times \frac{1}{\text{channel modules}} \right\rceil \quad (2)$$

$$\text{chip bandwidth} = \text{mem channels} \times \text{channel bandwidth} \quad (3)$$

$$\text{chip perf} = min\{\text{core perf} \times \text{chip cores}, \text{chip bandwidth}\} \quad (4)$$

$$\text{chip cores} = \left\lceil \frac{\text{chip perf}}{\text{core perf}} \right\rceil \quad (5)$$

$$\text{mem power} = \text{mem modules} \times \text{module power} \quad (6)$$

$$\text{compute power} = \text{chip cores} \times \text{core power} \times \text{compute chips} \quad (7)$$

$$\text{blades} = \left\lceil \frac{\text{compute chips}}{\text{blade chips}} \right\rceil \quad (8)$$

$$\text{response time} = \frac{\text{percent accessed} \times \text{db size}}{\text{chip perf} \times \text{compute chips}} \quad (9)$$

$$\text{power} = \text{mem power} + \text{compute power} + \text{blades} \times \text{blade power} \quad (10)$$

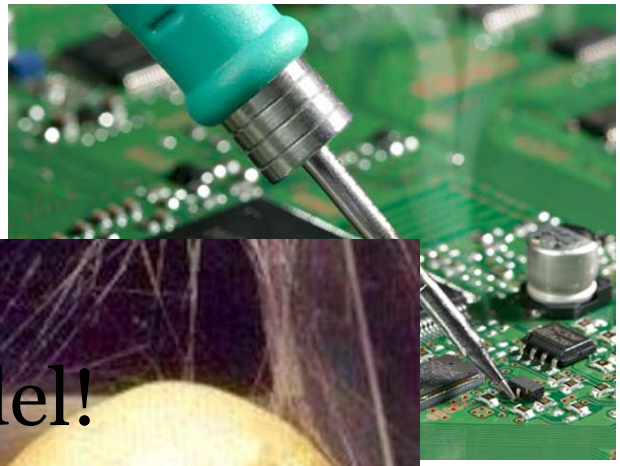
Model results

Discussion

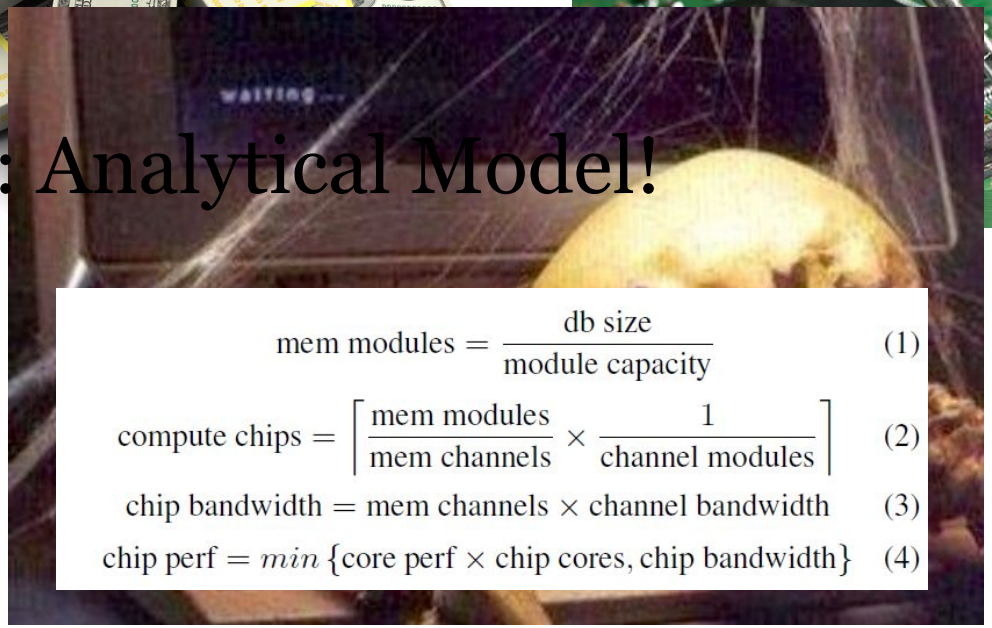
Evaluation

~~Option 1: Build the hardware~~

~~Option 2: Simulation~~



Option 3: Analytical Model!



$$\text{mem modules} = \frac{\text{db size}}{\text{module capacity}} \quad (1)$$

$$\text{compute chips} = \left\lceil \frac{\text{mem modules}}{\text{mem channels}} \times \frac{1}{\text{channel modules}} \right\rceil \quad (2)$$

$$\text{chip bandwidth} = \text{mem channels} \times \text{channel bandwidth} \quad (3)$$

$$\text{chip perf} = min\,\{\text{core perf} \times \text{chip cores}, \text{chip bandwidth}\} \quad (4)$$

Model Example

Provisioning: 10 ms response time

Data to read: 16,384 GB × 0.20 = **3,276.8 GB**

Bandwidth: 3,276.8 GB ÷ 0.010 s
= **327.680 TB/s**

Chips needed: 327.680 TB/s ÷ 102 GB/s/chip

Power: 458 kW = **3213 chips**

Capacity: 800 TB = **800 blades**

For traditional server

Model details

From the paper

$$\text{mem modules} = \frac{\text{db size}}{\text{module capacity}} \quad (1)$$

$$\text{compute chips} = \left\lceil \frac{\text{mem modules}}{\text{mem channels}} \times \frac{1}{\text{channel modules}} \right\rceil \quad (2)$$

$$\text{chip bandwidth} = \text{mem channels} \times \text{channel bandwidth} \quad (3)$$

$$\text{chip perf} = min\{\text{core perf} \times \text{chip cores}, \text{chip bandwidth}\} \quad (4)$$

$$\text{chip cores} = \left\lceil \frac{\text{chip perf}}{\text{core perf}} \right\rceil \quad (5)$$

$$\text{mem power} = \text{mem modules} \times \text{module power} \quad (6)$$

$$\text{compute power} = \text{chip cores} \times \text{core power} \times \text{compute chips} \quad (7)$$

$$\text{blades} = \left\lceil \frac{\text{compute chips}}{\text{blade chips}} \right\rceil \quad (8)$$

$$\text{response time} = \frac{\text{percent accessed} \times \text{db size}}{\text{chip perf} \times \text{compute chips}} \quad (9)$$

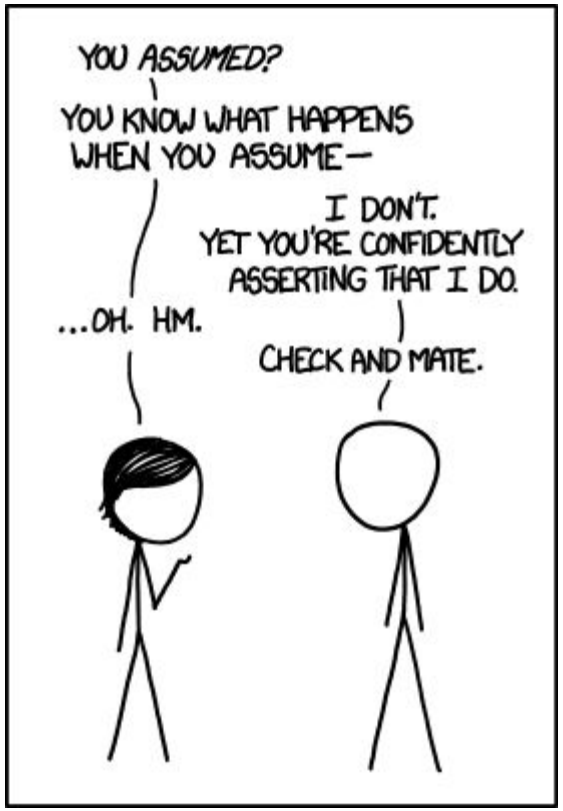
$$\text{power} = \text{mem power} + \text{compute power} + \text{blades} \times \text{blade power} \quad (10)$$

Online

research.cs.wisc.edu/multifacet/bpoe16_3d_bandwidth_model/

Workload Assumptions

- 16 TB data corpus
- Each request accesses 20% of data corpus (3.2 TB)
- One core can process 6 GB/s
- No communication between cores



https://xkcd.com/1339/

Outline

Model and Workload

$$\text{mem modules} = \frac{\text{db size}}{\text{module capacity}} \quad (1)$$

$$\text{compute chips} = \left\lceil \frac{\text{mem modules}}{\text{mem channels}} \times \frac{1}{\text{channel modules}} \right\rceil \quad (2)$$

$$\text{chip bandwidth} = \text{mem channels} \times \text{channel bandwidth} \quad (3)$$

$$\text{chip perf} = min\{\text{core perf} \times \text{chip cores}, \text{chip bandwidth}\} \quad (4)$$

$$\text{chip cores} = \left\lceil \frac{\text{chip perf}}{\text{core perf}} \right\rceil \quad (5)$$

$$\text{mem power} = \text{mem modules} \times \text{module power} \quad (6)$$

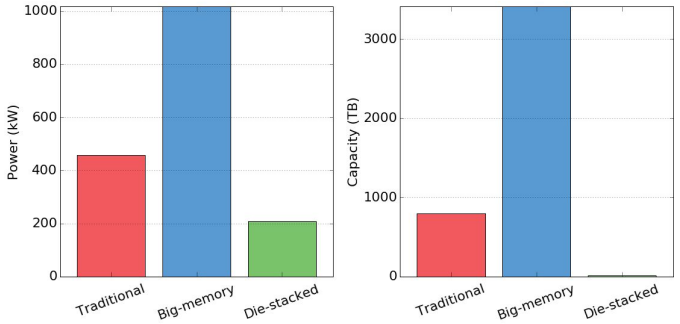
$$\text{compute power} = \text{chip cores} \times \text{core power} \times \text{compute chips} \quad (7)$$

$$\text{blades} = \left\lceil \frac{\text{compute chips}}{\text{blade chips}} \right\rceil \quad (8)$$

$$\text{response time} = \frac{\text{percent accessed} \times \text{db size}}{\text{chip perf} \times \text{compute chips}} \quad (9)$$

$$\text{power} = \text{mem power} + \text{compute power} + \text{blades} \times \text{blade power} \quad (10)$$

Model results



Discussion

Metrics

Performance

Response time (SLA)

Power

Major component of datacenter cost

Data capacity

Workload size

Performance Provisioning

Goal: Design cluster to meet a *service level agreement* (SLA)

Get matches

10 ms

Sort

50 ms

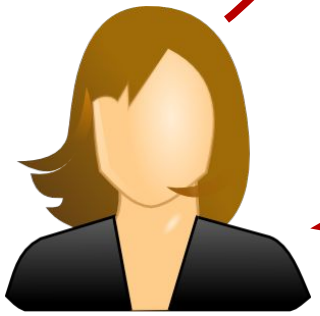
Ads

100 ms

. . .

50 ms

50 ms

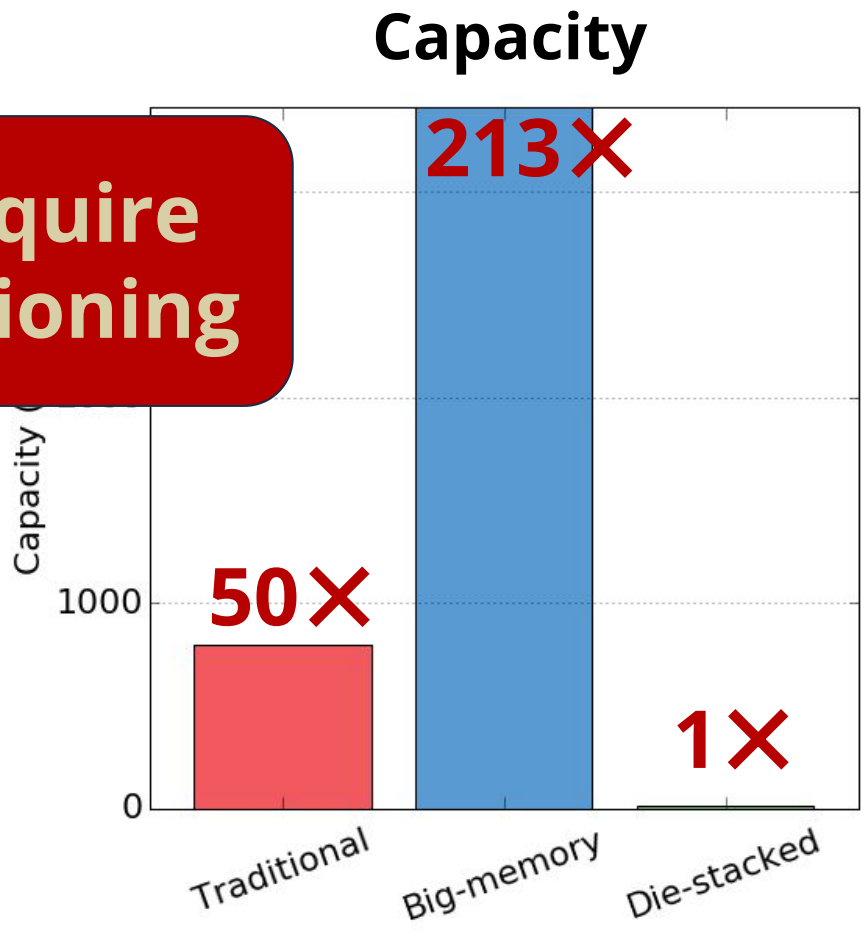


500 ms

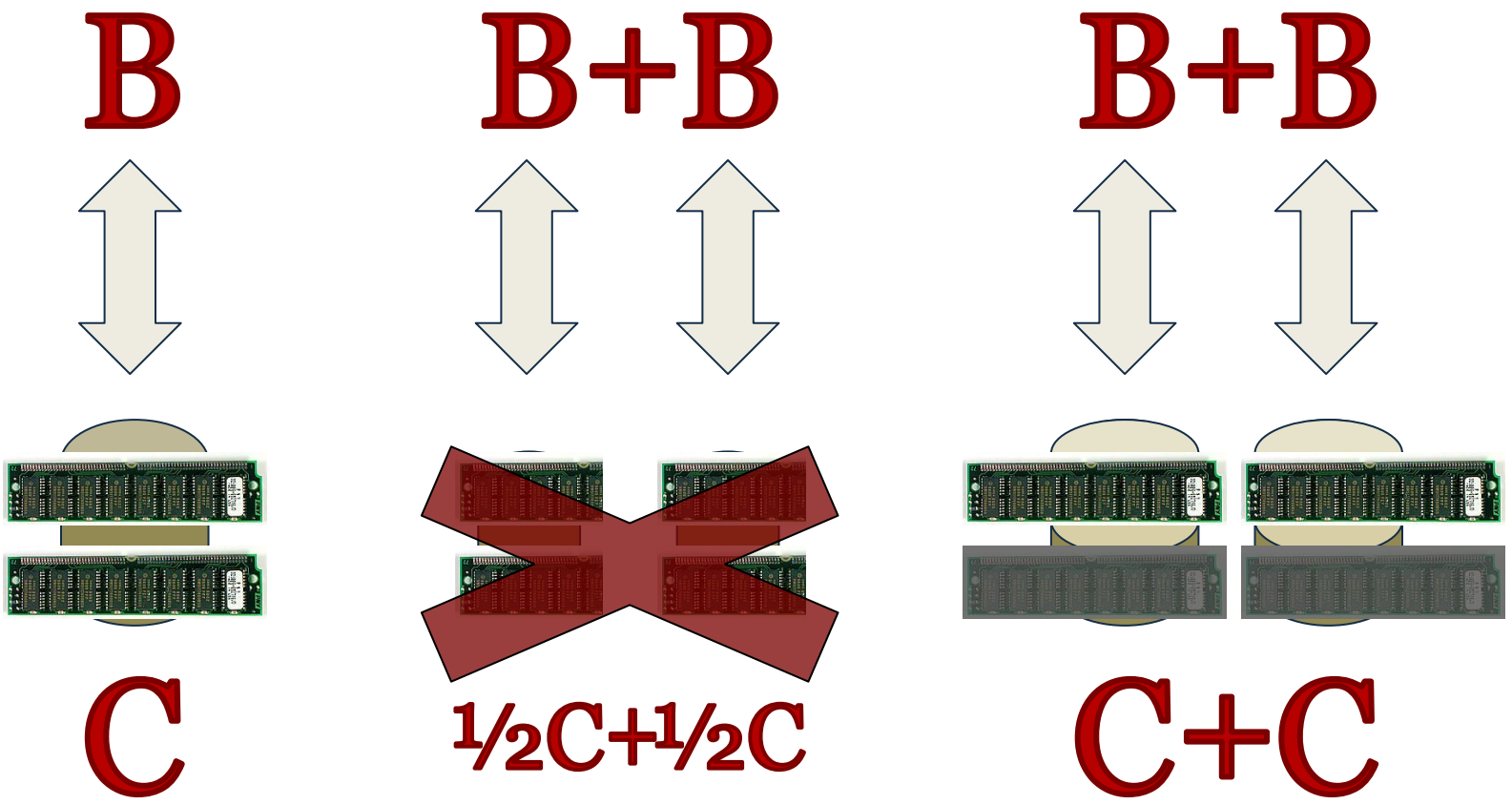
Performance Provisioning

10 ms SLA

Current systems require memory over provisioning



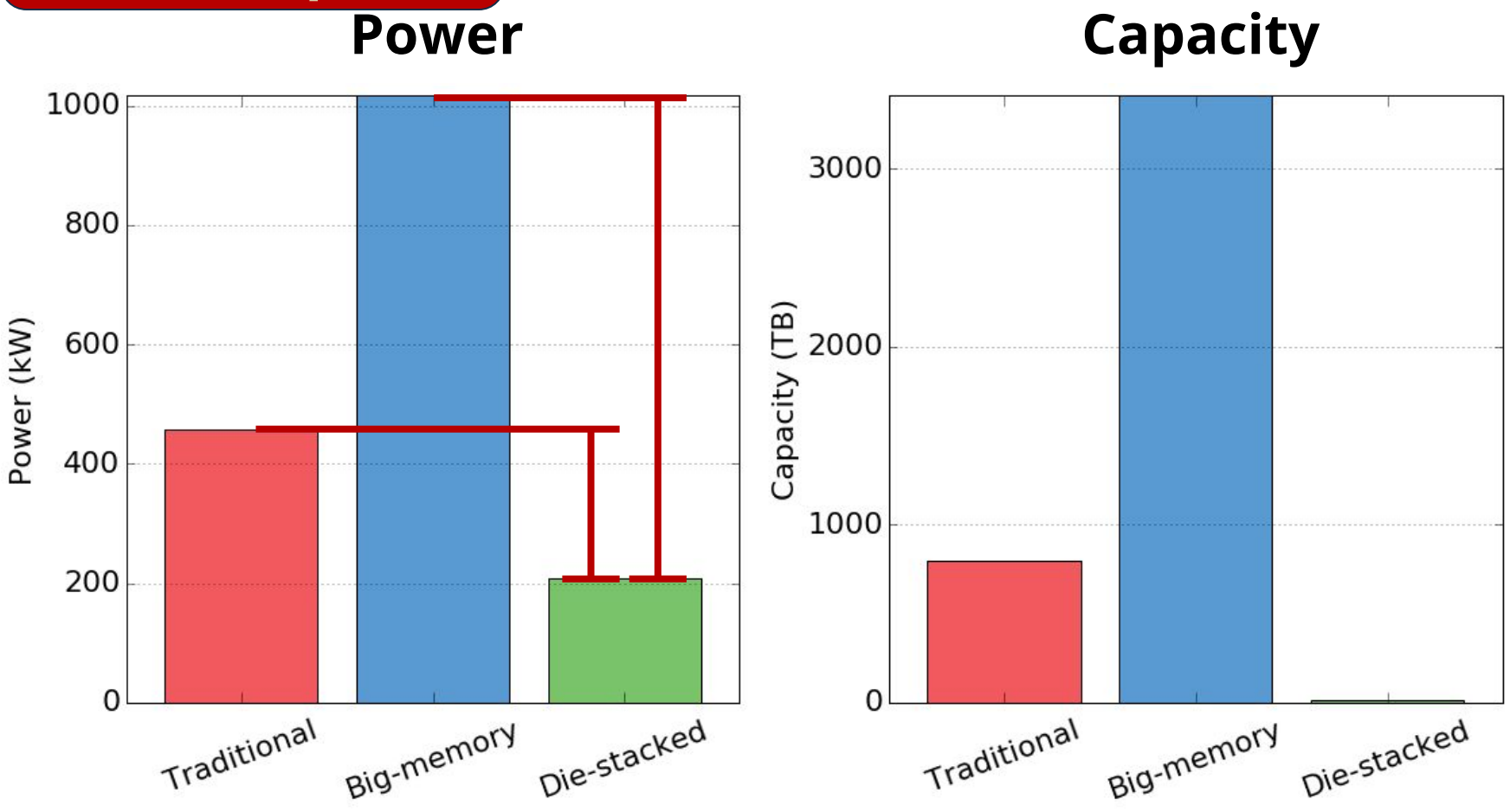
Memory Over Provisioning



Performance Provisioning

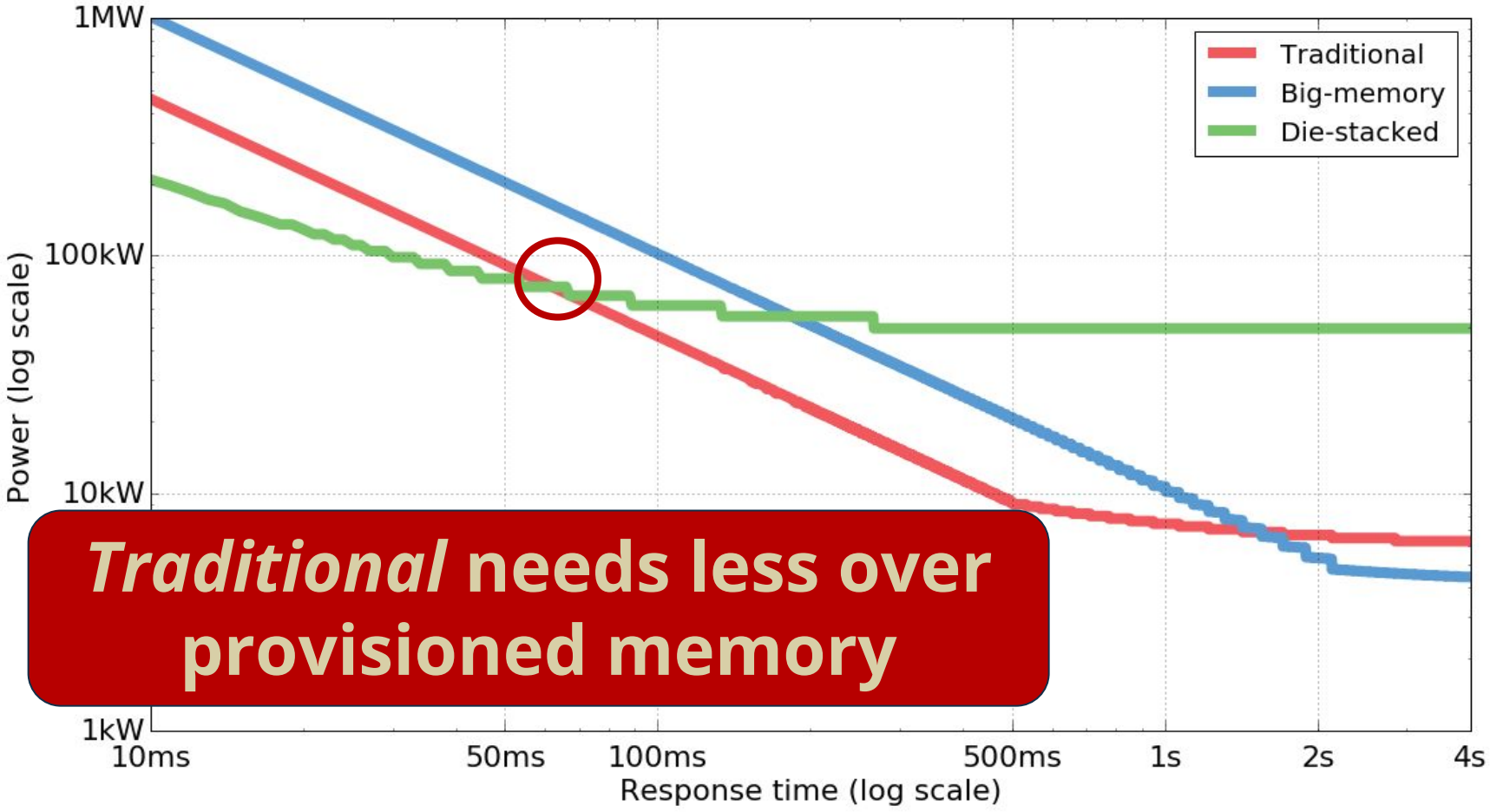
***Die-stacking*: 2–5× less power**

10 ms SLA



Performance Provisioning

Power for relaxed SLAs



Traditional **needs less over provisioned memory**

Power Provisioning

Goal: Design cluster to not exceed some *power constraint*

10–20 kW

100kW–1MW

10–100 MW

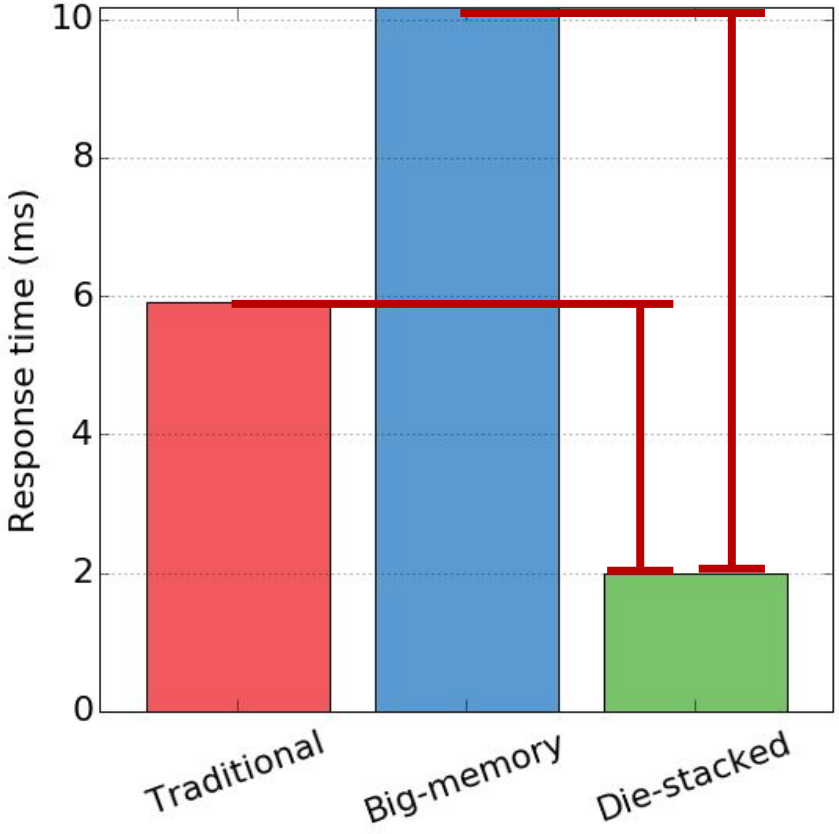
Power Provisioning

1 MW Power

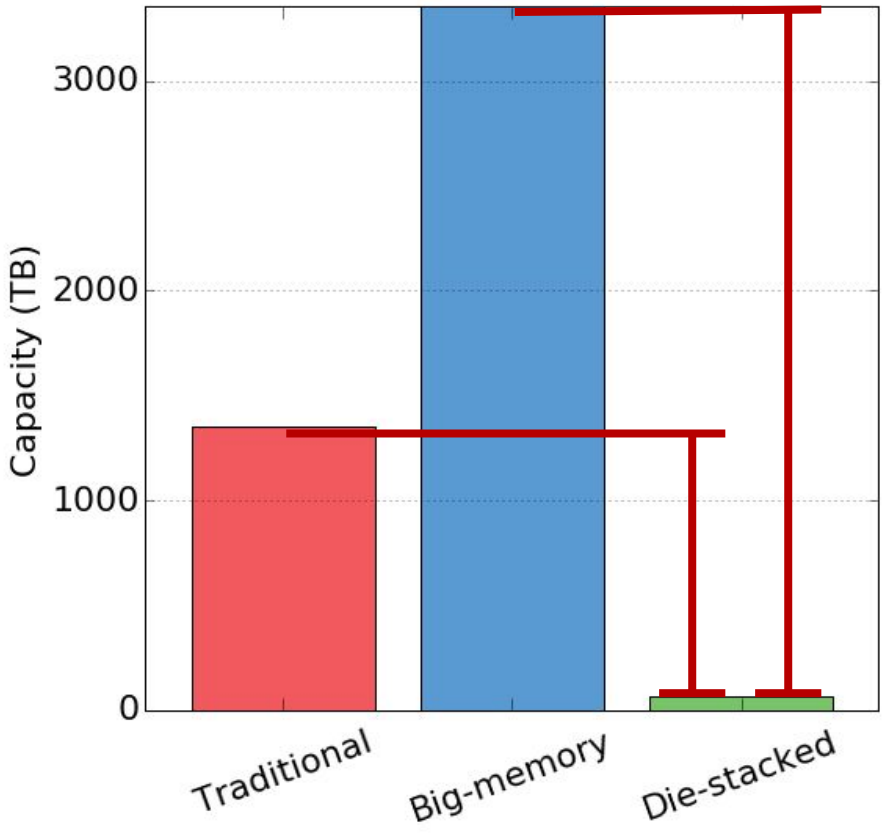
***Die-stacking*: Less capacity for power budget**

***Die-stacking*: 3–5× faster**

Response time



Capacity



Data Capacity Provisioning

Goal: Design cluster *capacity* for workload

Search: Inverted Index

32 TB

Graph: Friends lists

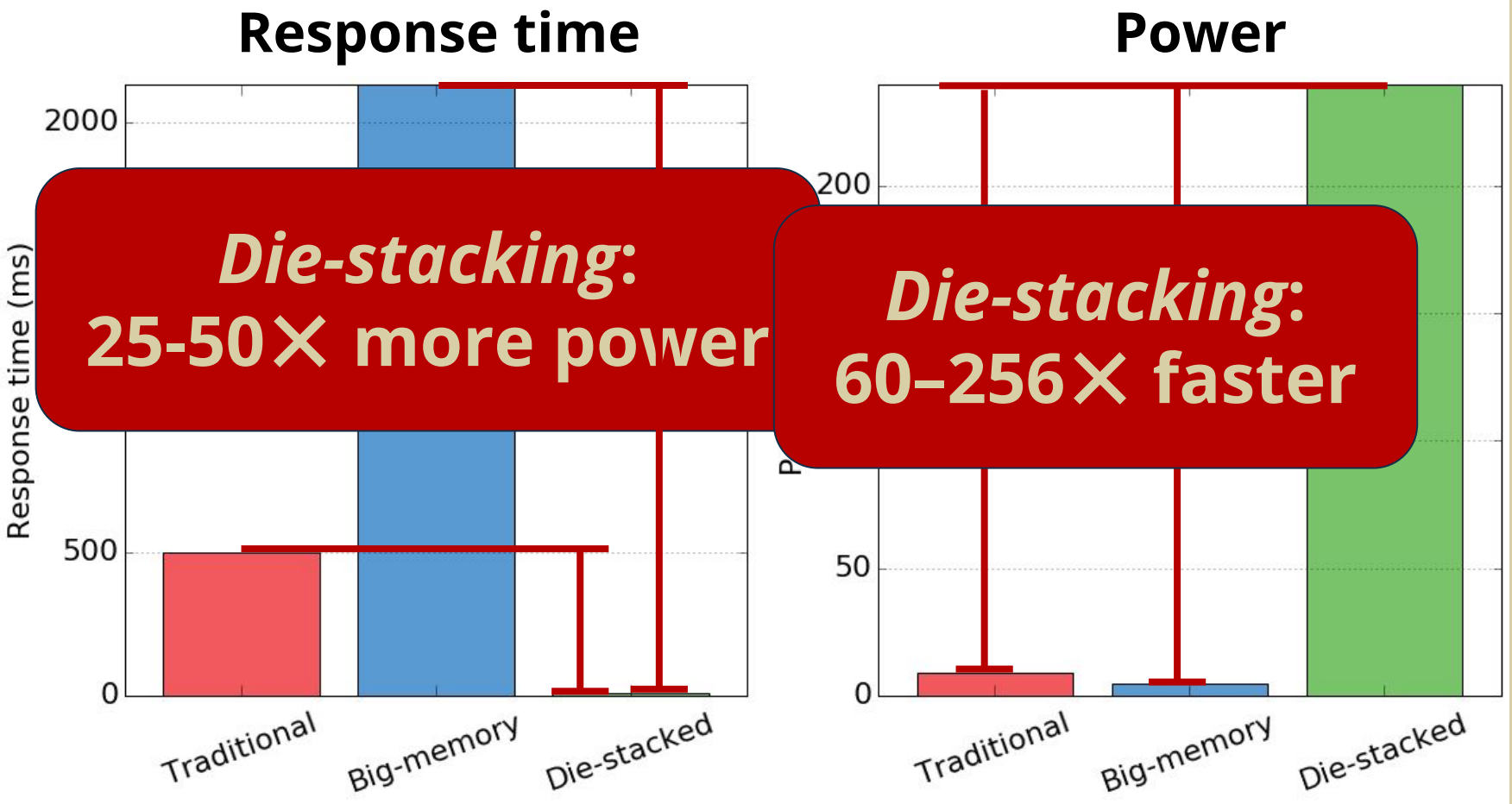
100 TB

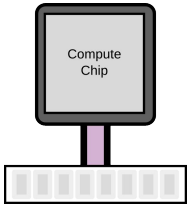
Database: Purchases

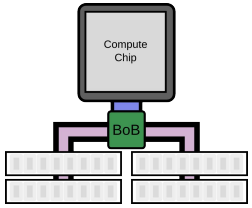
1 PB

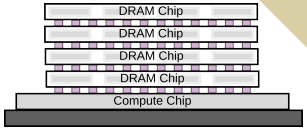
Data Capacity Provisioning

16 TB Database









	Traditional	Big Memory	Die-Stacked
Performance	Best for SLA 60+ms	Over provisioned memory	2–5x less power for 10ms SLA
Power	2x faster with 50 KW	3x memory capacity	3–4x faster with 1 MW
Data capacity	Somewhere between	2–50x less power	60–250x faster

Outline

Model and Workload

$$\text{mem modules} = \frac{\text{db size}}{\text{module capacity}} \quad (1)$$

$$\text{compute chips} = \left\lceil \frac{\text{mem modules}}{\text{mem channels}} \times \frac{1}{\text{channel modules}} \right\rceil \quad (2)$$

$$\text{chip bandwidth} = \text{mem channels} \times \text{channel bandwidth} \quad (3)$$

$$\text{chip perf} = min\{\text{core perf} \times \text{chip cores}, \text{chip bandwidth}\} \quad (4)$$

$$\text{chip cores} = \left\lceil \frac{\text{chip perf}}{\text{core perf}} \right\rceil \quad (5)$$

$$\text{mem power} = \text{mem modules} \times \text{module power} \quad (6)$$

$$\text{compute power} = \text{chip cores} \times \text{core power} \times \text{compute chips} \quad (7)$$

$$\text{blades} = \left\lceil \frac{\text{compute chips}}{\text{blade chips}} \right\rceil \quad (8)$$

$$\text{response time} = \frac{\text{percent accessed} \times \text{db size}}{\text{chip perf} \times \text{compute chips}} \quad (9)$$

$$\text{power} = \text{mem power} + \text{compute power} + \text{blades} \times \text{blade power} \quad (10)$$

Model results

Discussion

Model deficiencies

You chose the wrong number!

- See research.cs.wisc.edu/multifacet/bpoe16_3d_bandwidth_model/

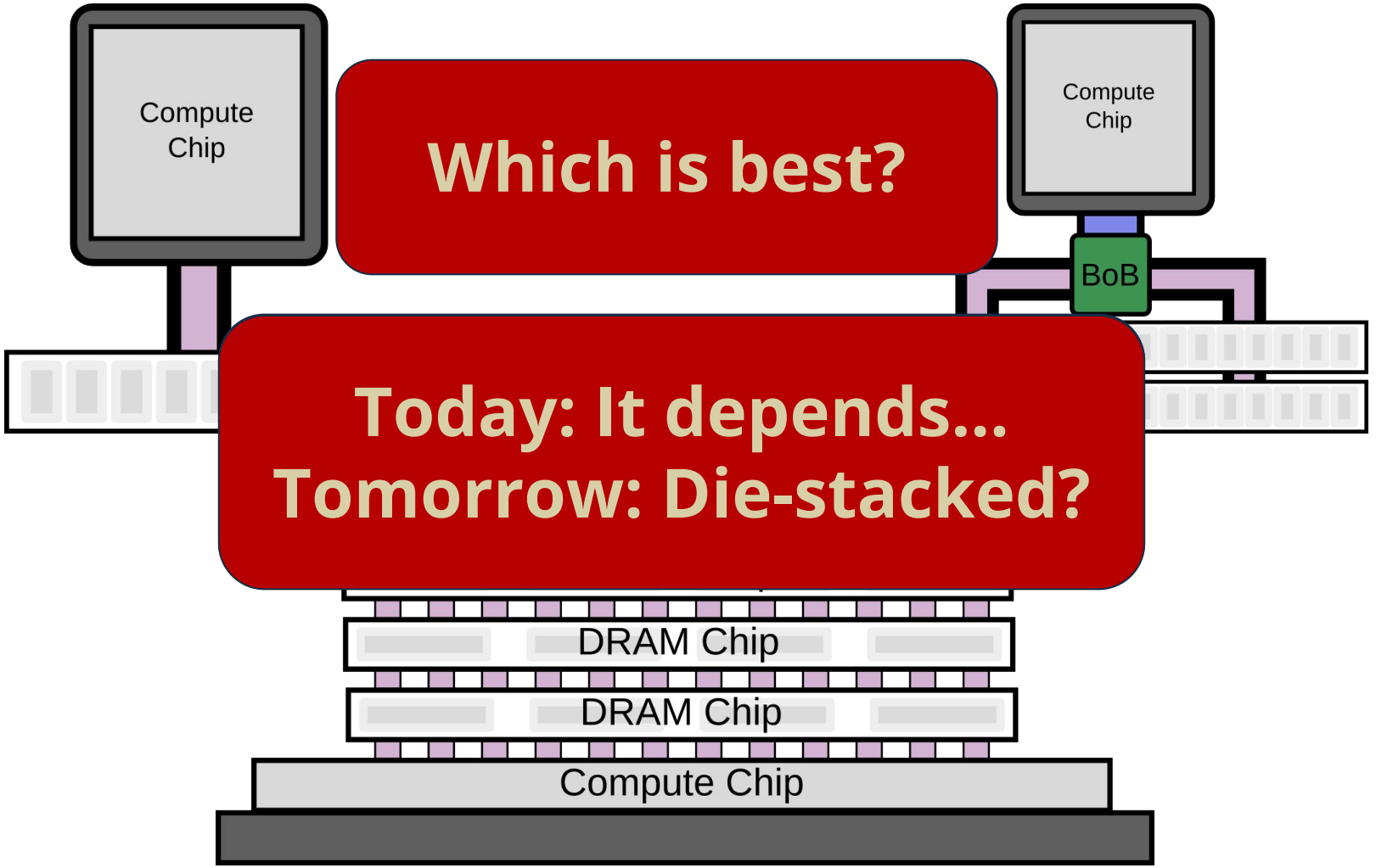
Communication between cores

- This makes 2048 die-stacked systems worse
- How to move data between stacks?

Compute energy or data energy?

Cost?

In Memory Big Data Workloads



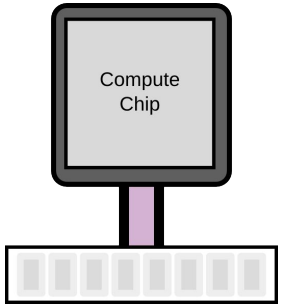
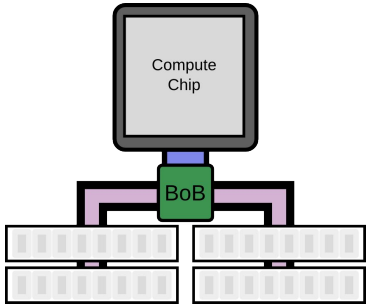
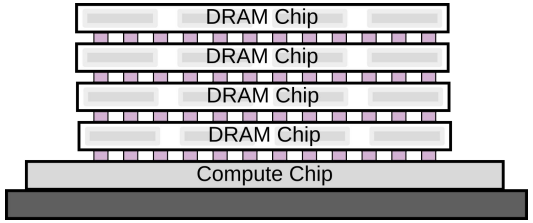
Questions?

research.cs.wisc.edu/multifacet/bpoe16_3d_bandwidth_model/

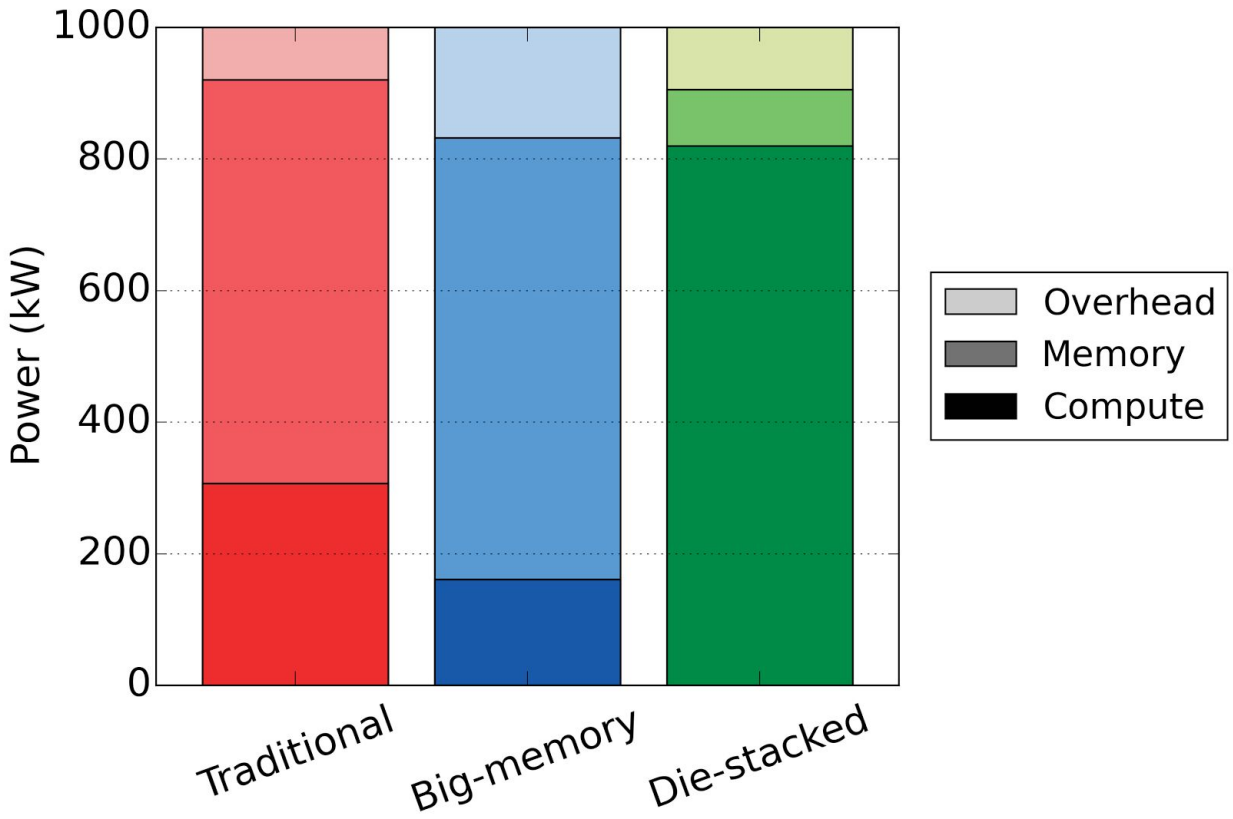
bit.ly/bpoe-interactive

powerjg@cs.wisc.edu

Systems

	Traditional	Big memory	Die-stacked
			
Bandwidth	102 GB/s	196 GB/s	256 GB/s
Capacity	256 GB	2 TB	8 GB
Blades (16TB)	16	8	228
Cluster bandwidth	6.4 TB/s	1.5 TB/s	512 TB/s

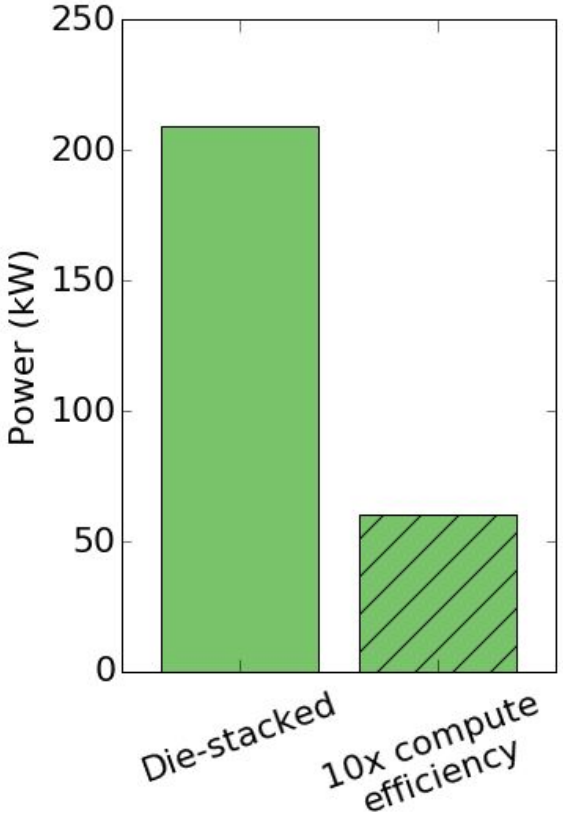
Power Breakdown



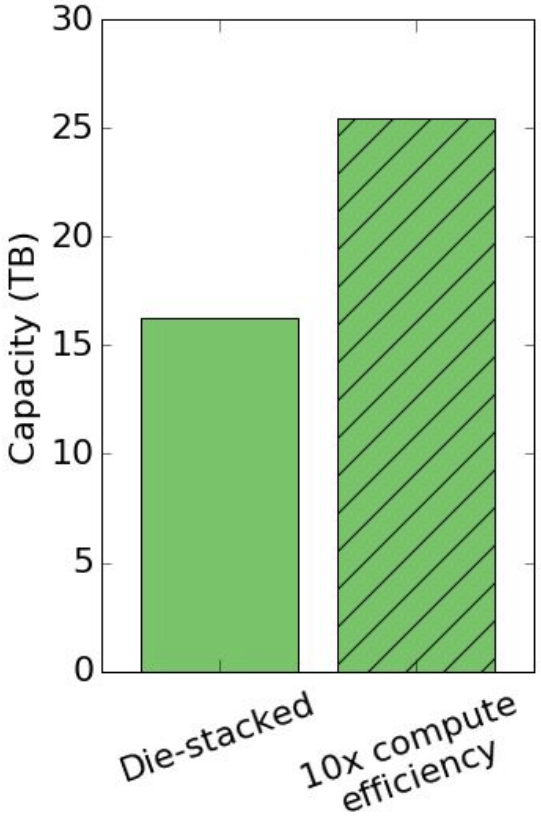
Compute power dominates *die-stacked*

Decreased Compute Power

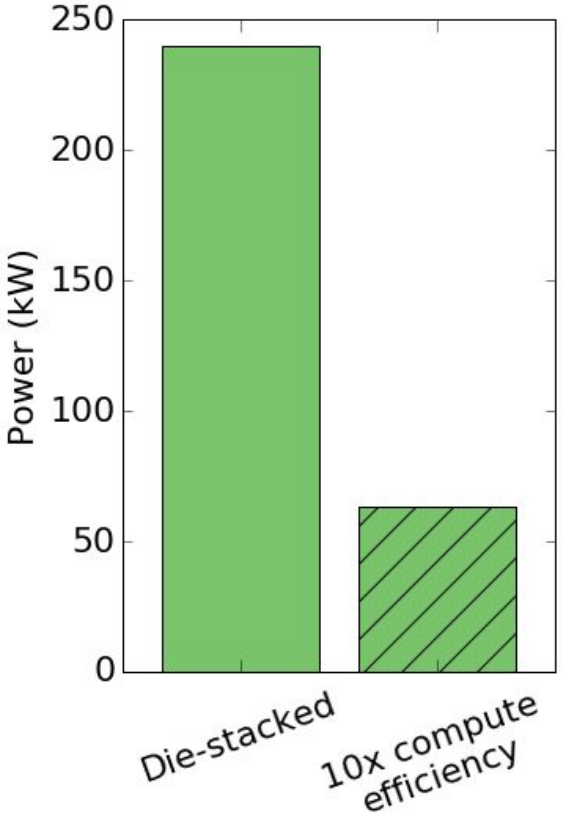
10 ms SLA



100 kW Power

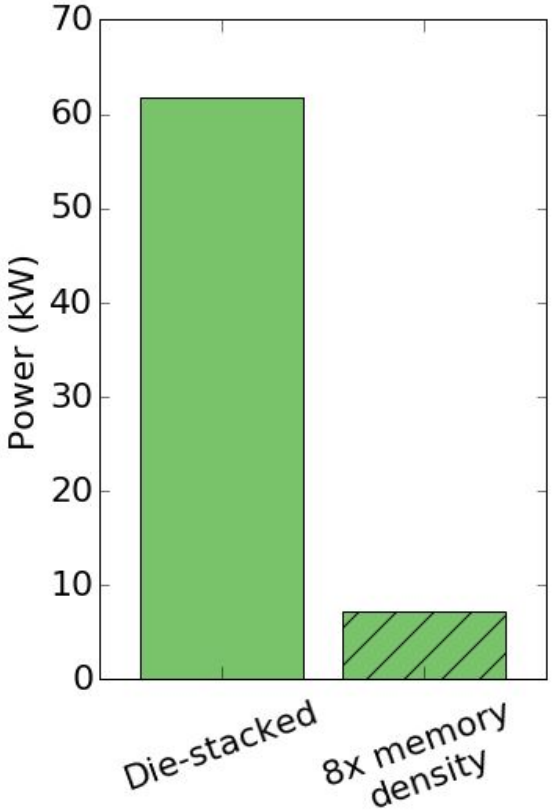


16 TB Capacity

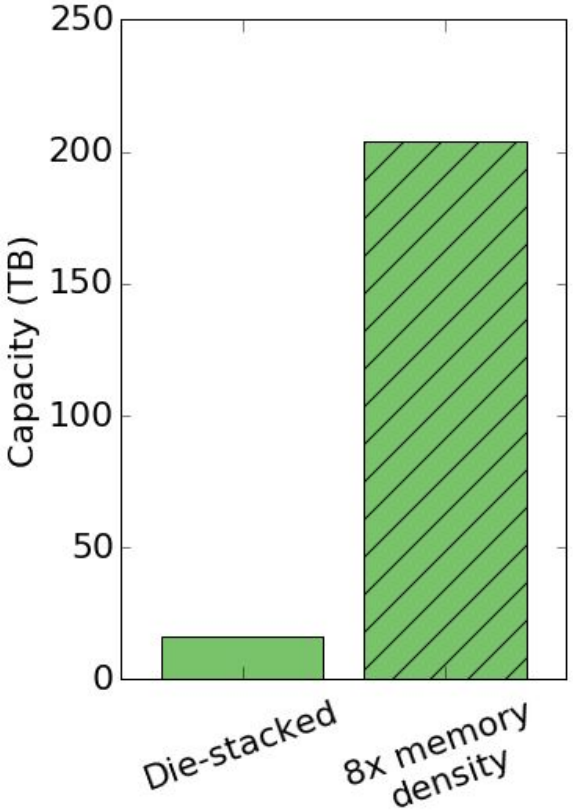


Increased Memory Density

100 ms SLA



100 kW Power



16 TB Capacity