Bassi/Power5 Architecture

John Shalf

NERSC Users Group Meeting

Princeton Plasma Physics Laboratory

June 2005

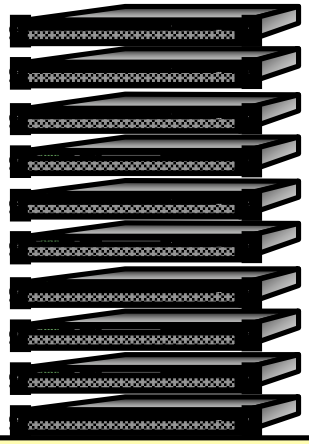
U.S. Department of Energy
Office of Science

POWER5 IH Overview

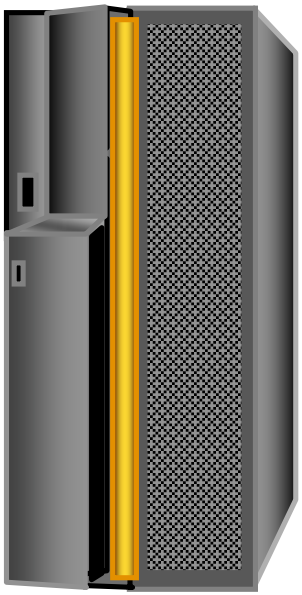
Berkeley Lab

POWER5 IH System

- 2U rack chassis
  - Rack: 24" X 43 " Deep, Full Drawer



16 Systems / Rack
128 Processors / Rack



	POWER5 IH Node
Architecture	4W or 8W POWER5 Processors
L3 Cache	144MB / 288MB (total)
Memory	2GB - 128/256GB
Packaging	2U (24" rack) 16 Nodes / Rack
DASD / Bays	2 DASD (Hot Plug)
I/O Expansion	6 slots (Blindswap)
Integrated SCSI	Ultra 320
Integrated Ethernet	4 Ports 10/100/1000
RIO Drawers	Yes (1/2 or 1)
LPAR	Yes
Switch	HPS
OS	AIX 5.2 & Linux

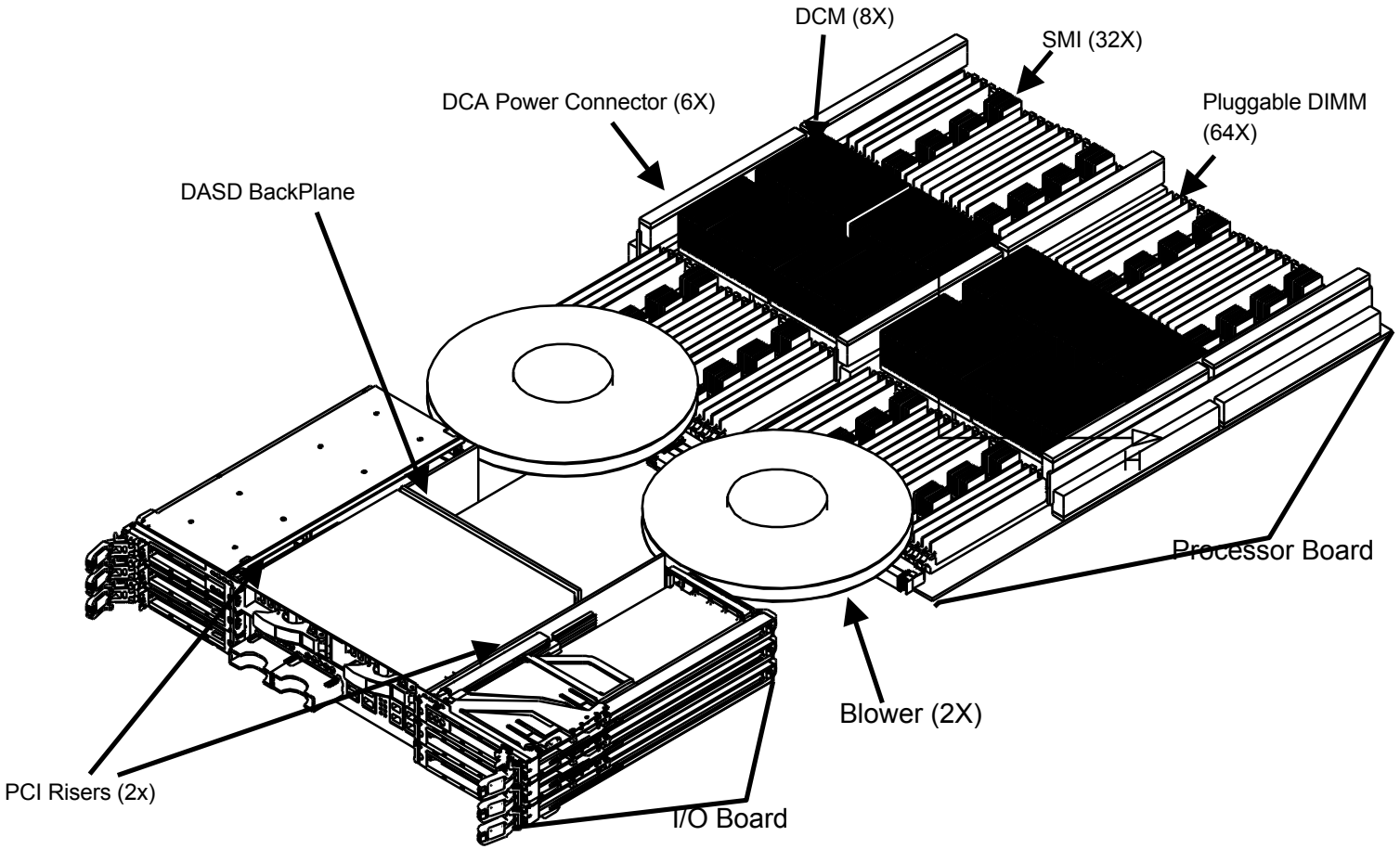
NERSC

Image From IBM

U.S. Department of Energy
Office of Science

POWER5 IH Physical Structure

BERKELEY LAB



NERSC

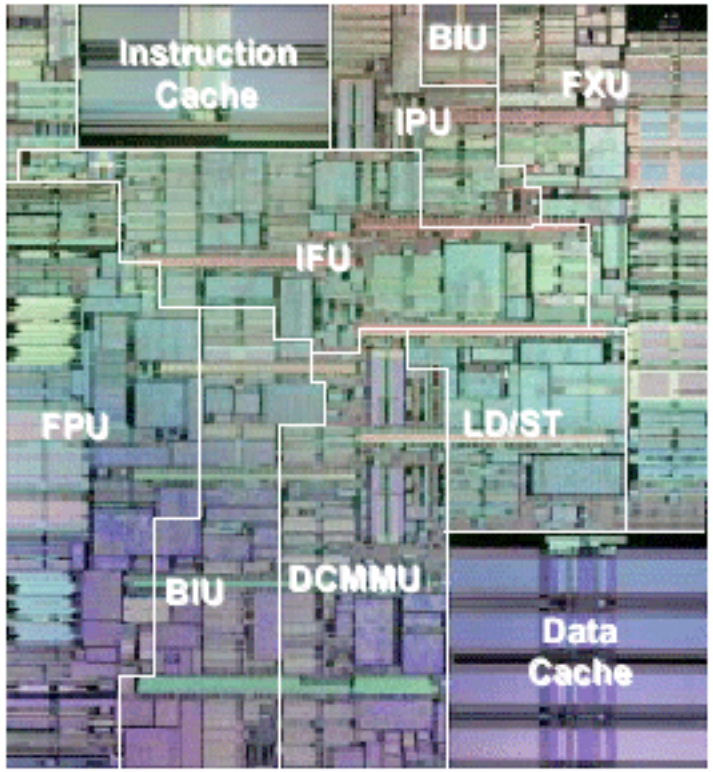
Image From IBM

IBM Power Series Processors

	Power3+	Power4	Power5
MHz	375	1300	1900
FLOPS/clock	4	4	4
Peak FLOPS	1.5	5.2	7.6
L1 (D-Cache)	64k	32k	32k
L2 (unified)	8MB	1.5M (0.75M)	1.9M
L3 (unified)	--	32M (16M)	36M
STREAM GB/s	0.4 (0.7)	1.4 (2.4)	5.4 (7.6)
Bytes/FLOP	0.3	0.44	0.7

Power3 vs. Power5 die

Power3



Power5

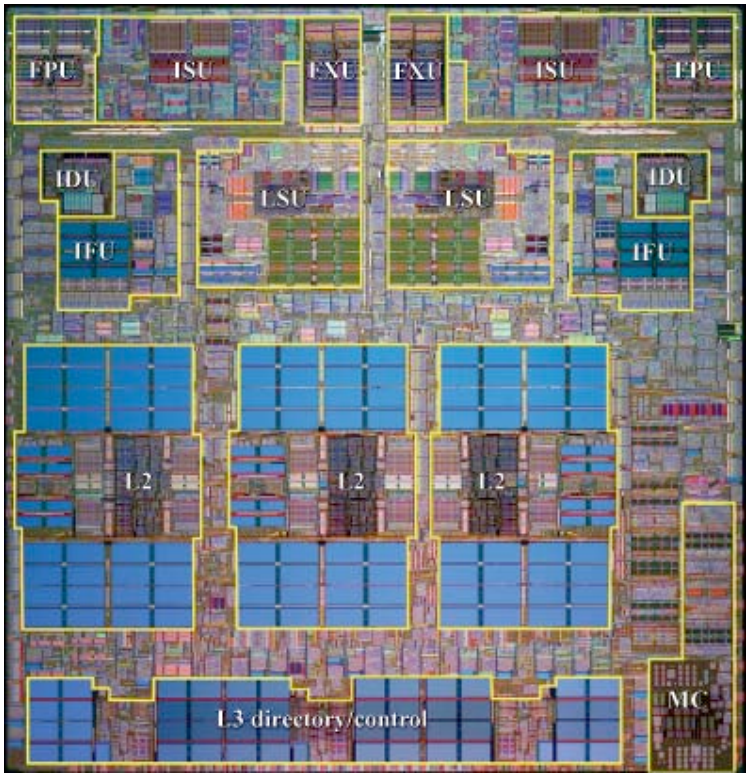


Image From IBM Power3 Redbook

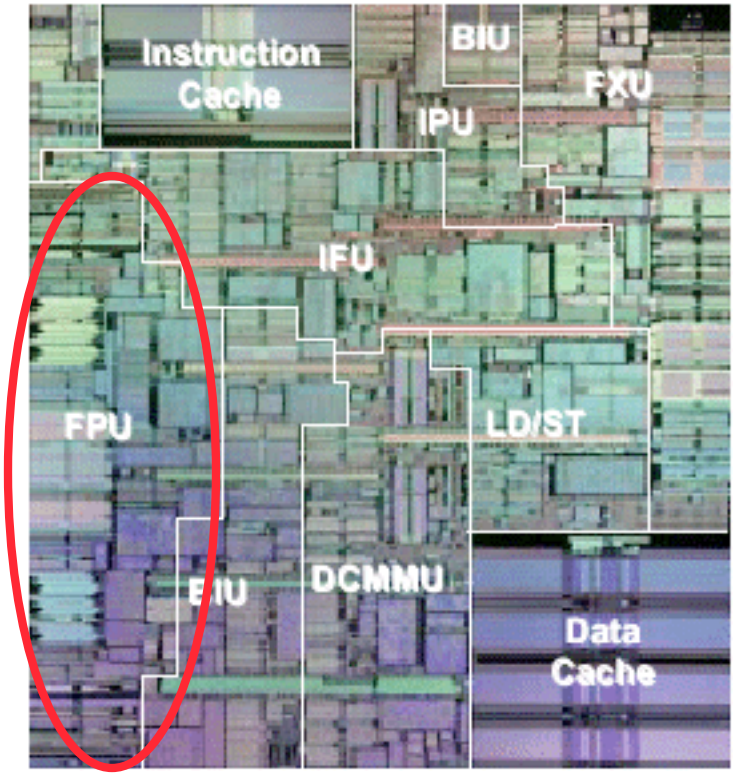
Image From IBM Power5 Redbook

U.S. Department of Energy
Office of Science

Power3 vs. Power5 die

BERKELEY LAB

Power3



Power5

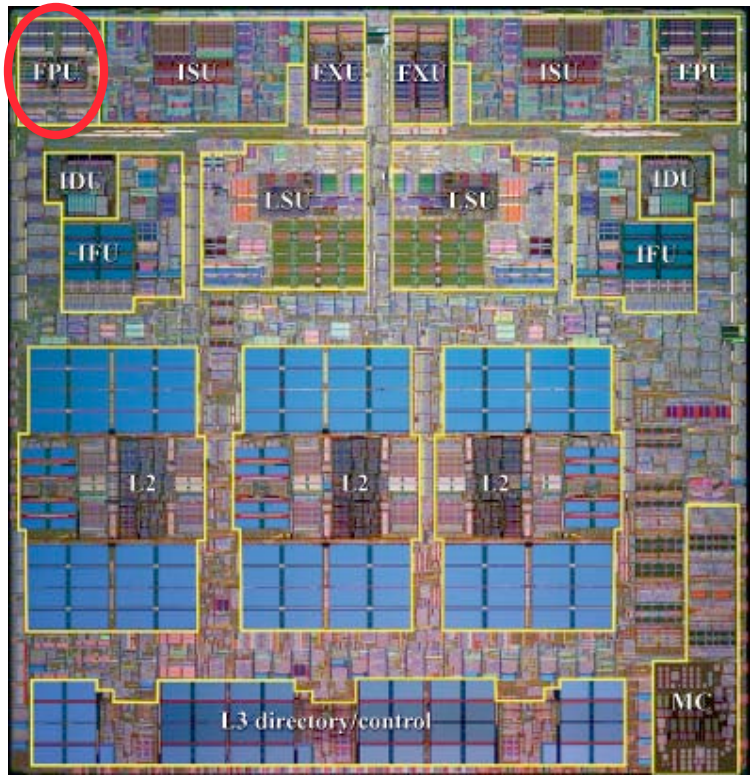


Image From IBM Power3 Redbook

NERSC

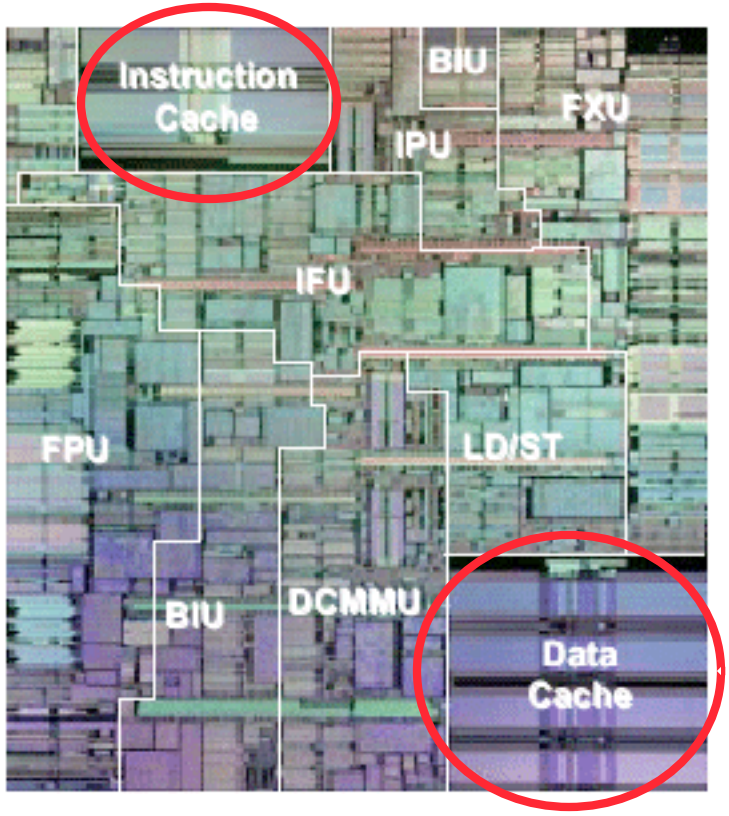
Image From IBM Power5 Redbook

U.S. Department of Energy
Office of Science

Power3 vs. Power5 die

BERKELEY LAB

Power3



Power5

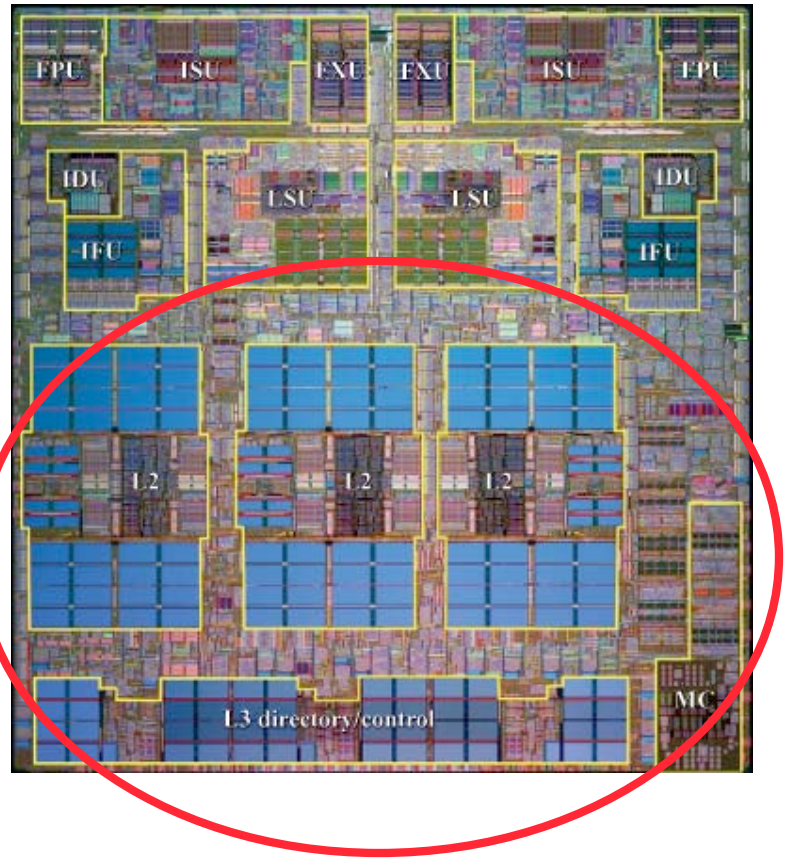


Image From IBM Power3 Redbook

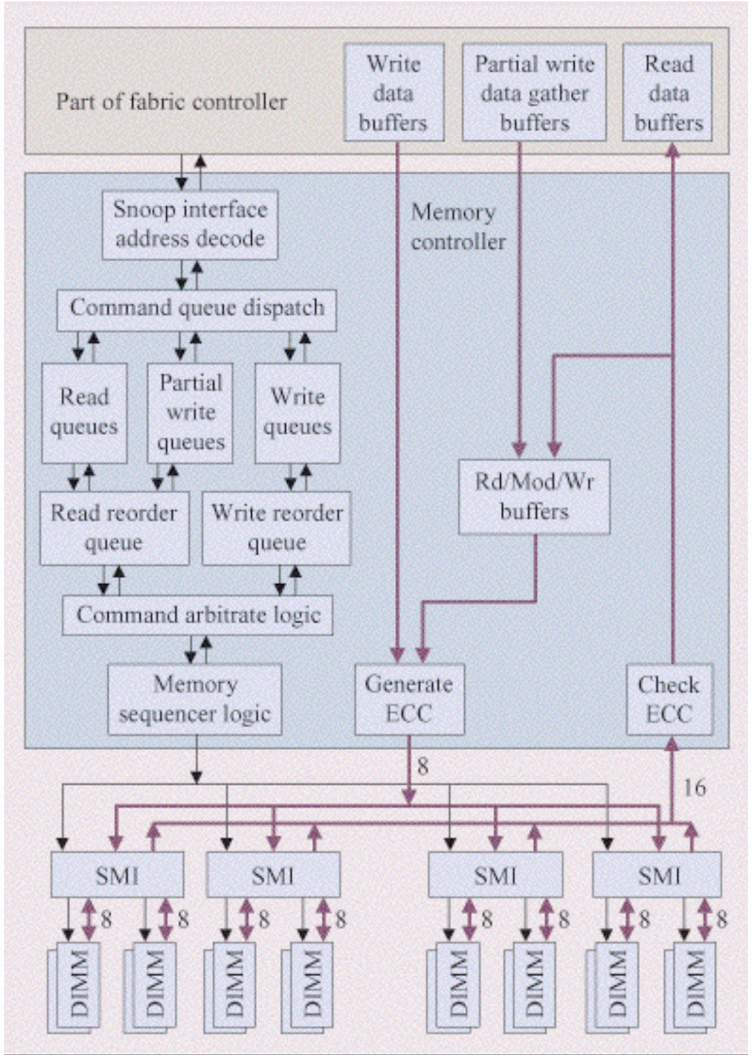
NERSC

Image From IBM Power5 Redbook

U.S. Department of Energy
Office of Science

Memory Subsystem

BERKELEY LAB



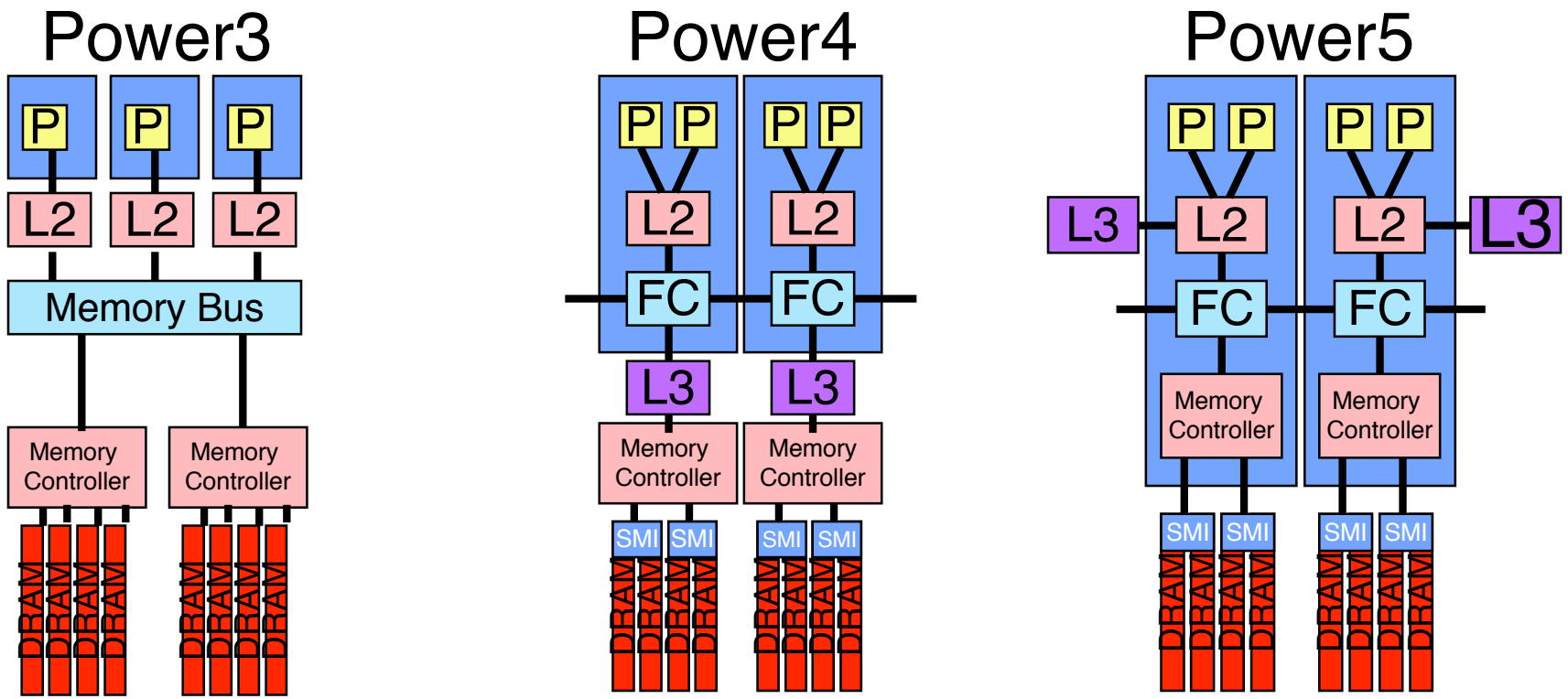
NERSC

Image From IBM Power5 Redbook

U.S. Department of Energy
Office of Science

SMP Fabric

Berkeley Lab



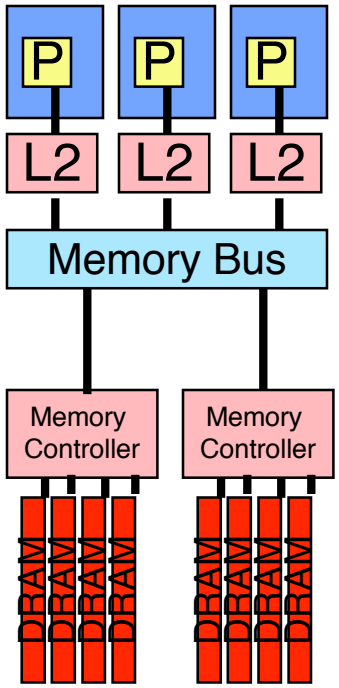
NERSC

U.S. Department of Energy
Office of Science

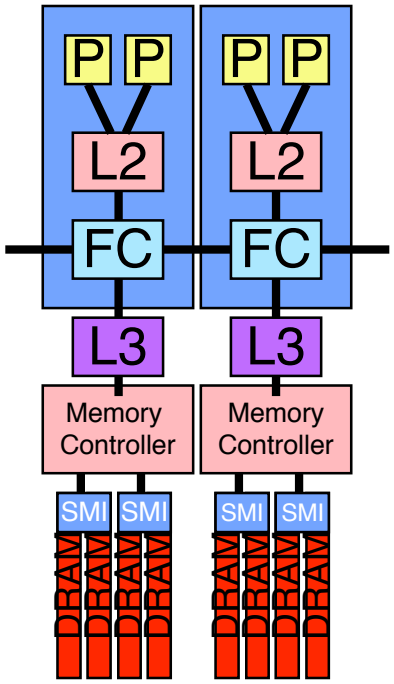
SMP Fabric

Berkeley Lab

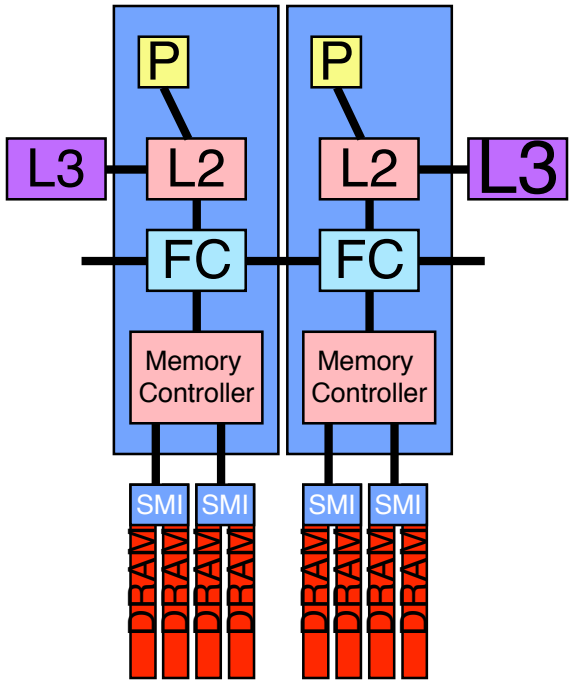
Power3



Power4



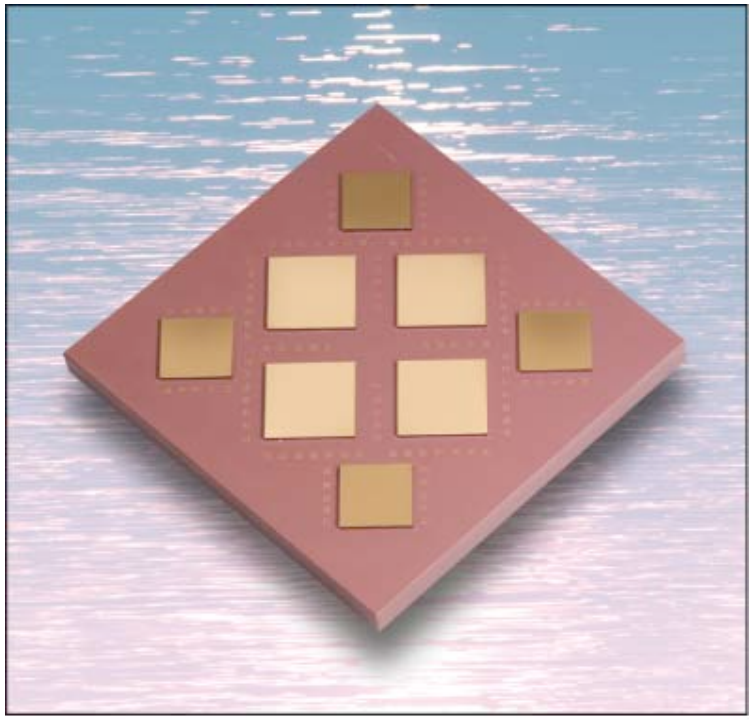
Power5 SC



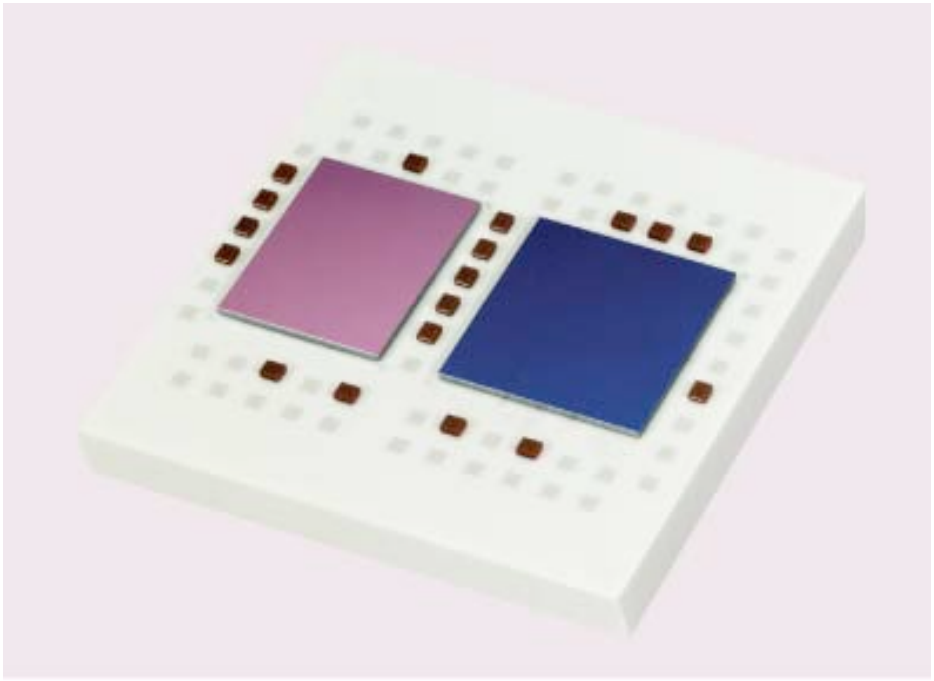
Remaining core gets
- All of L2 + L3 cache
- All memory BW
- Increased Clock Rate

NERSC

System Packaging (MCM vs. DCM)



- Multi-Chip Module (MCM)
- 4 x Power5 + L3 Caches
- Up to 8 Power5 cores
- L2 shared among cores



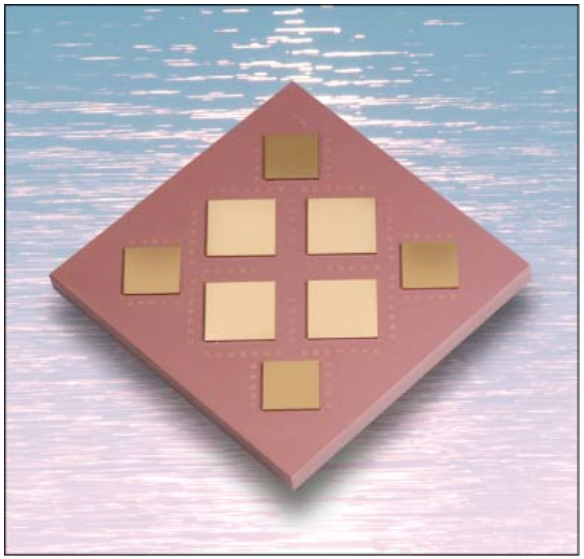
- Dual-Chip Module (DCM)
- 1 x Power5 + L3 Cache
- Up to 2 Power5 cores
- Private L2 and L3

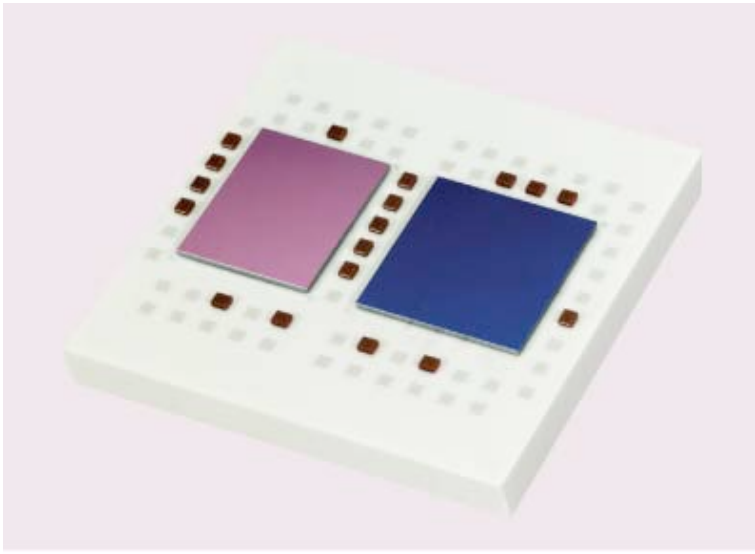
Image From IBM Power5 Redbook

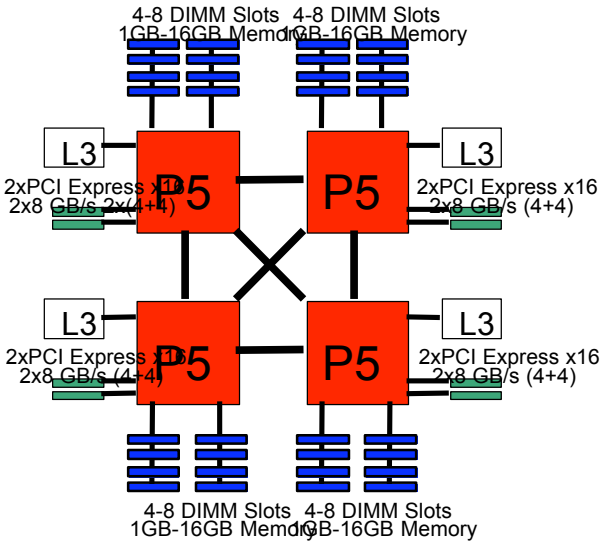
U.S. Department of Energy
Office of Science

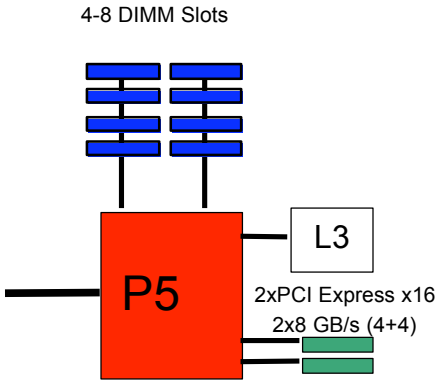
System Packaging (MCM vs. DCM)

BERKELEY LAB







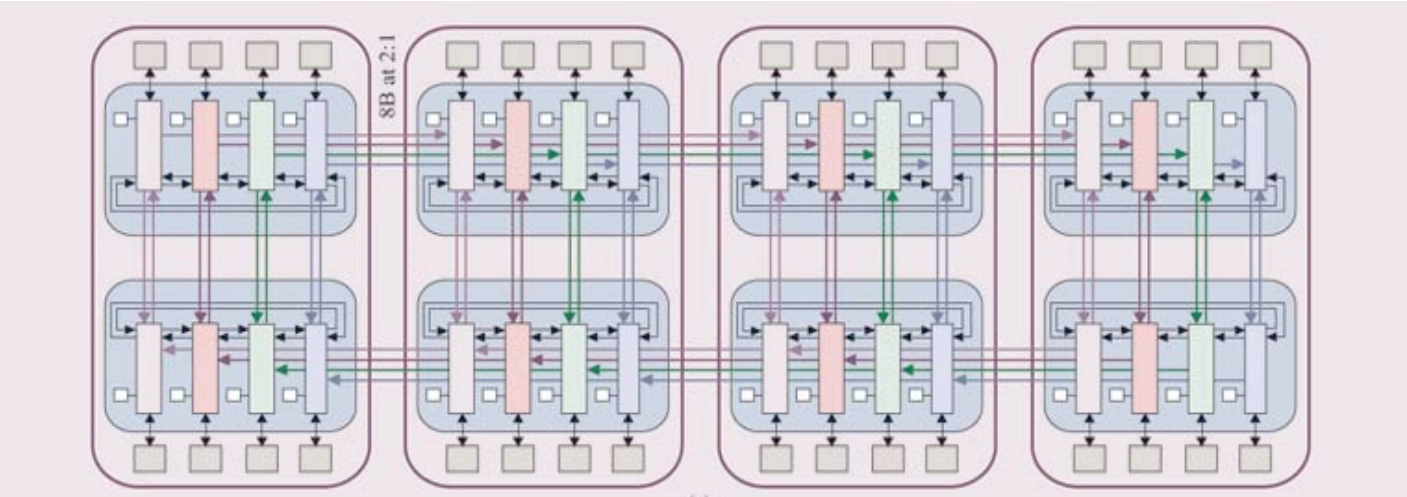


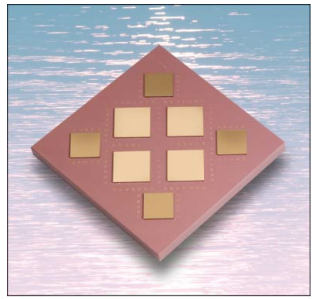
NERSC

U.S. Department of Energy
Office of Science

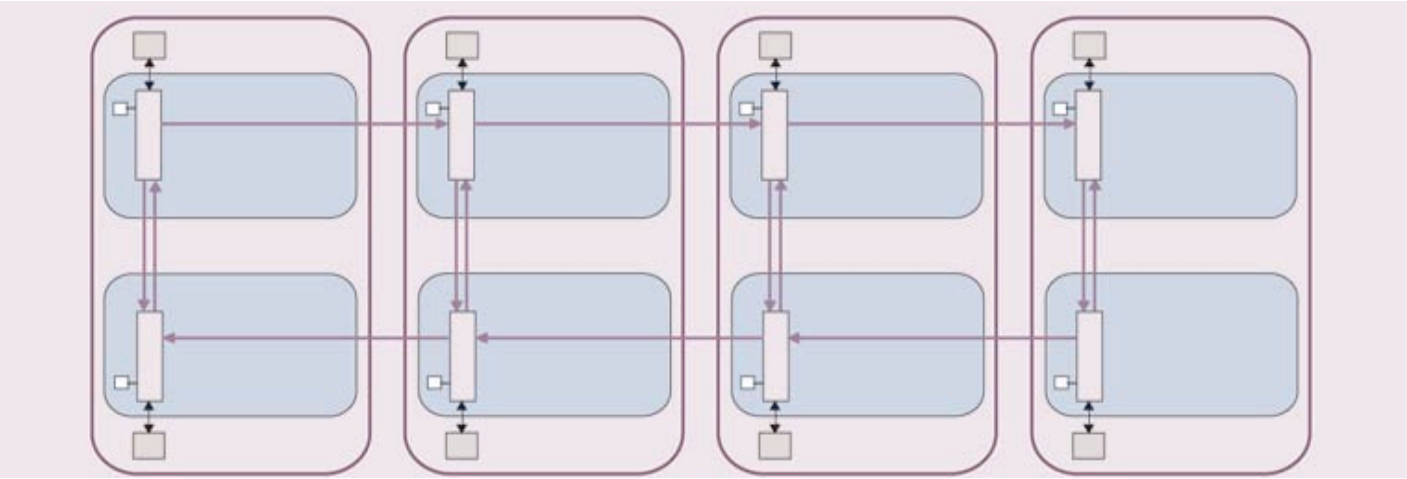
Power5 Cache Hierarchy

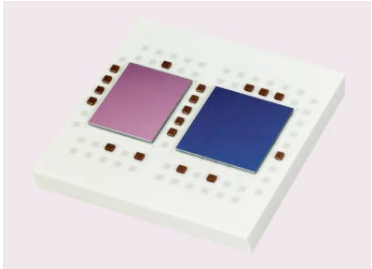
Berkeley Lab





P5 Squadron
- 4 MCM
- 64 proc SMP





P5 Squadron IH
- 4 DCM
- 8 proc@1.9GHz
- 16 proc@1.5GHz

Image From IBM Power5 Redbook

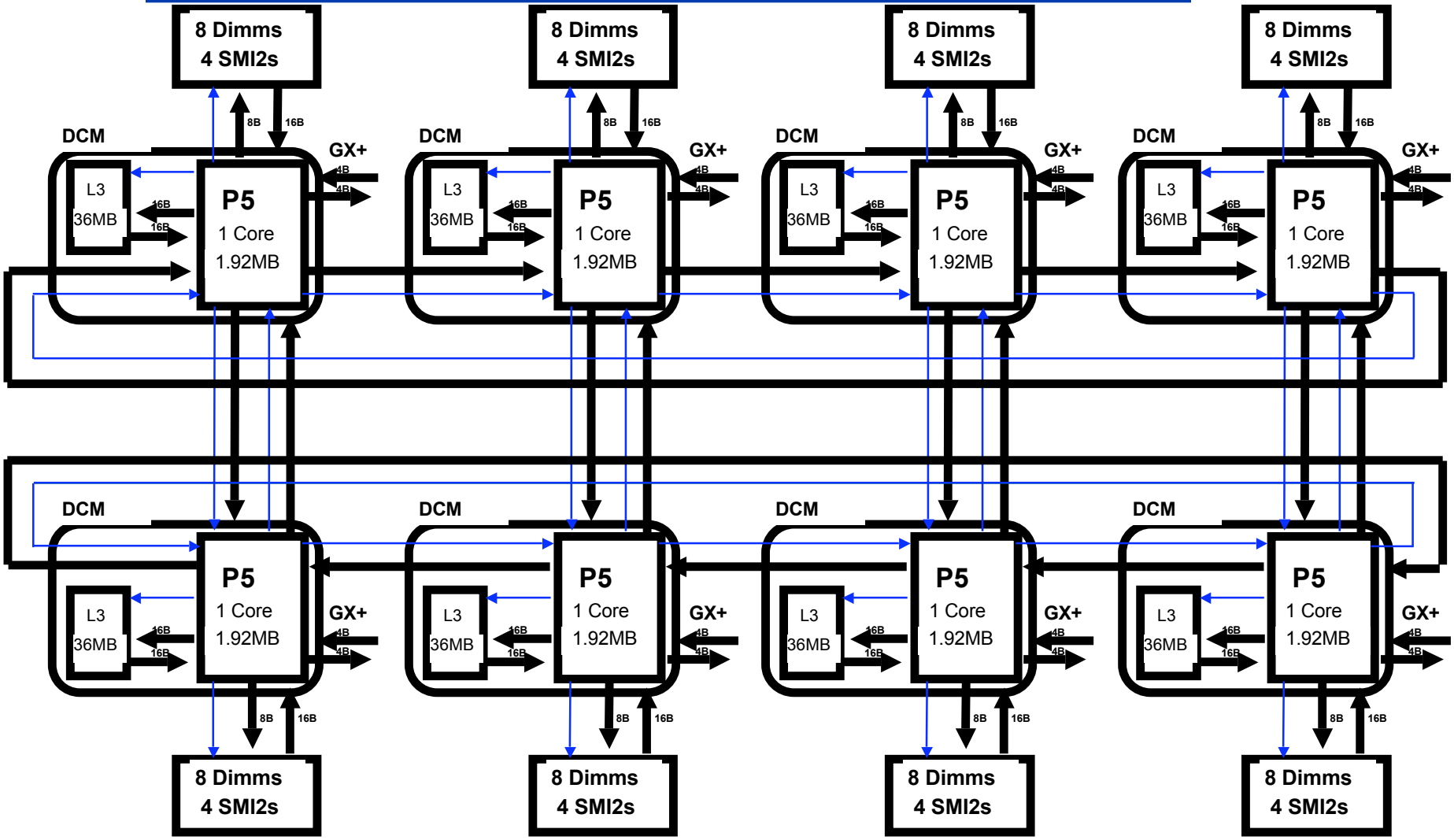
NERSC

U.S. Department of Energy

Office of Science

Power5-IH Node Architecture

BERKELEY LAB



Address/Control Buses in Green

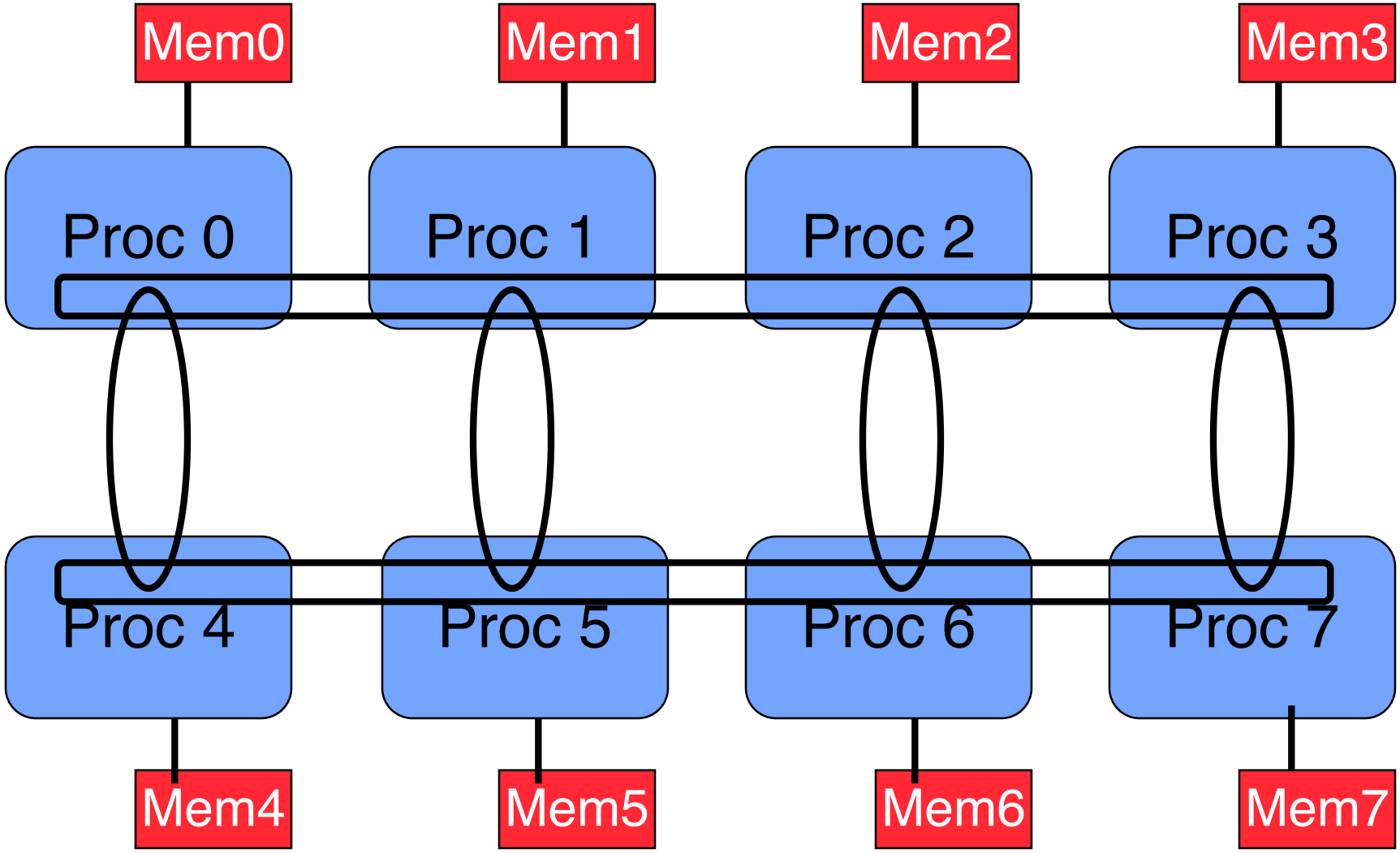
Notes: 1) SMP Buses run at 1.0 Ghz
2) L3 Buses run at 1.0 Ghz
3) Memory Buses run at 1.066 Ghz

Image From IBM

U.S. Department of Energy
Office of Science

Power5 IH Memory Affinity

Berkeley Lab



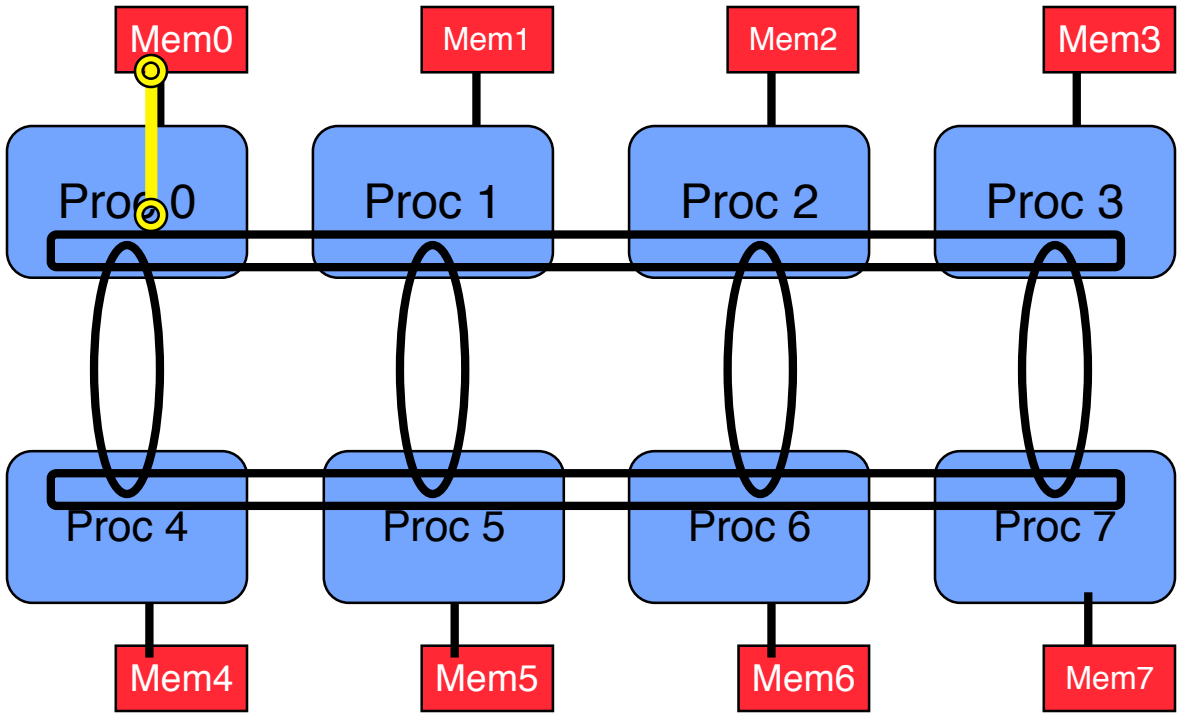
NERSC

U.S. Department of Energy
Office of Science

Power5 IH Memory Affinity

Berkeley Lab

Proc0 to Mem0 == 90ns



NERSC

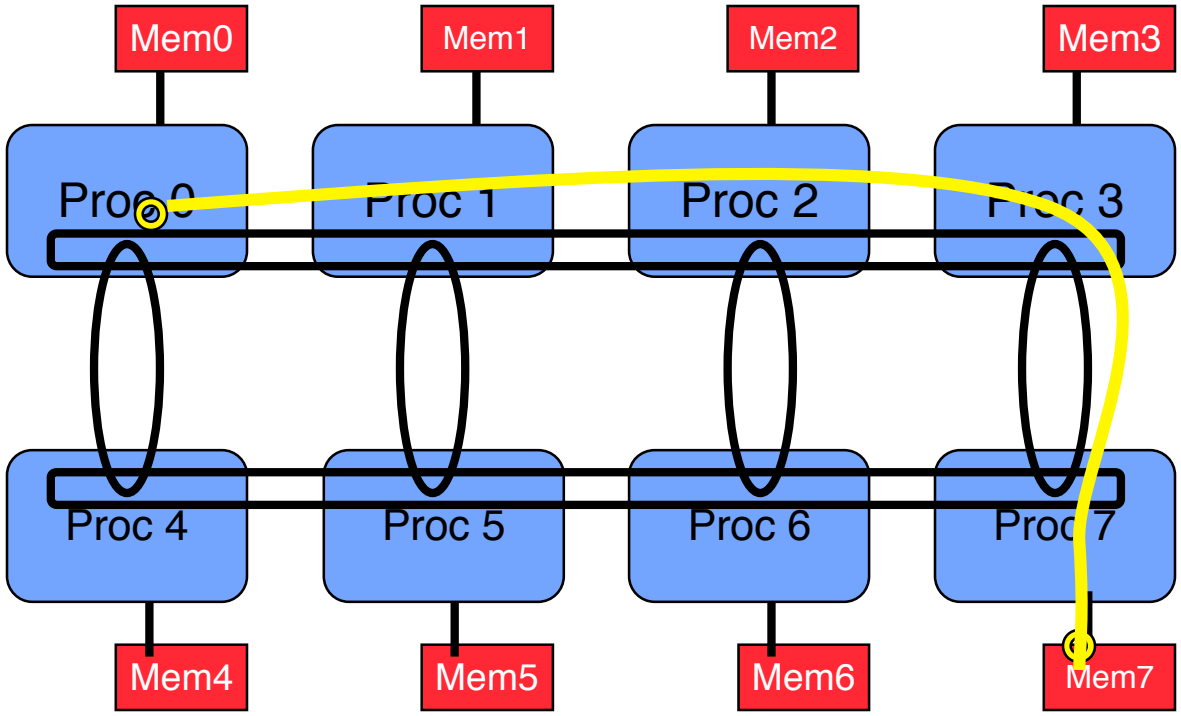
U.S. Department of Energy
Office of Science

Power5 IH Memory Affinity

Berkeley Lab

Proc0 to Mem0 == 90ns

Proc0 to Mem7 == 200+ns



NERSC

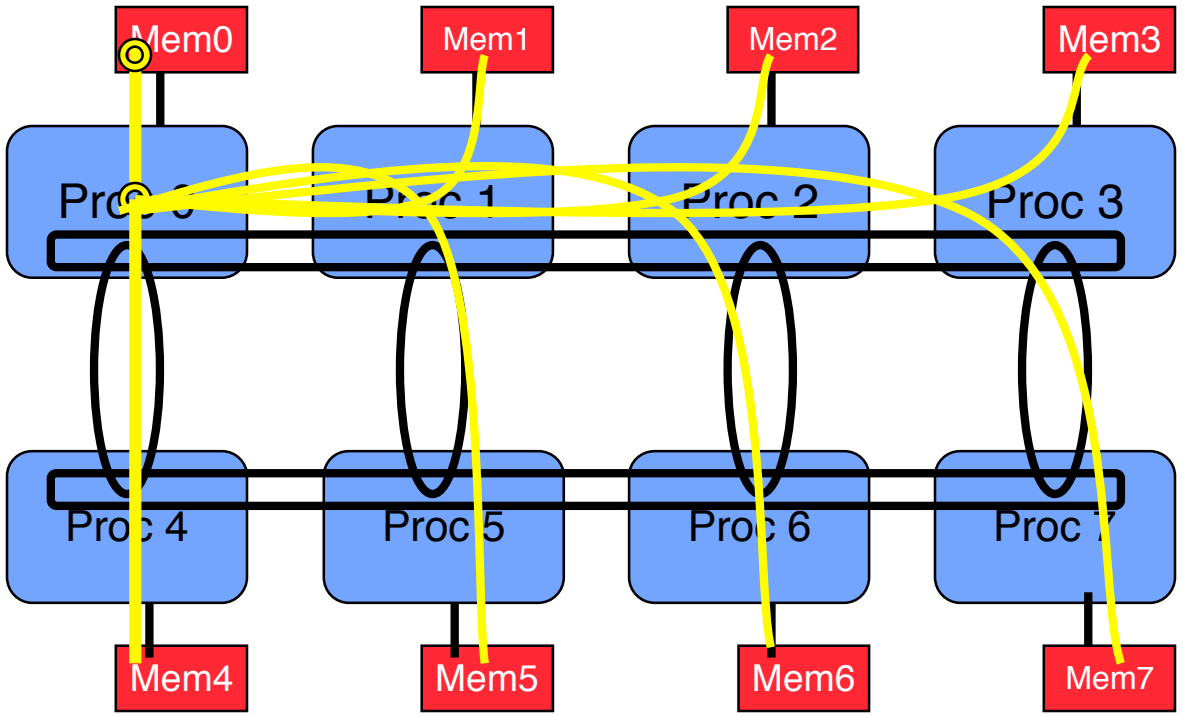
U.S. Department of Energy

Office of Science

Page Mapping

Berkeley Lab

Proc0 Page#	Mem# RR
0	0
1	1
2	2
3	3
5	4
6	5
7	6
8	7
9	0
10	1
11	2
12	3
13	4
14	5
15	6
16	7



Linear Walk through Memory Addresses

- Default Affinity is round_robin (RR)
- Pages assigned round-robin to mem ctrlrs.
- Average latency ~170-190ns

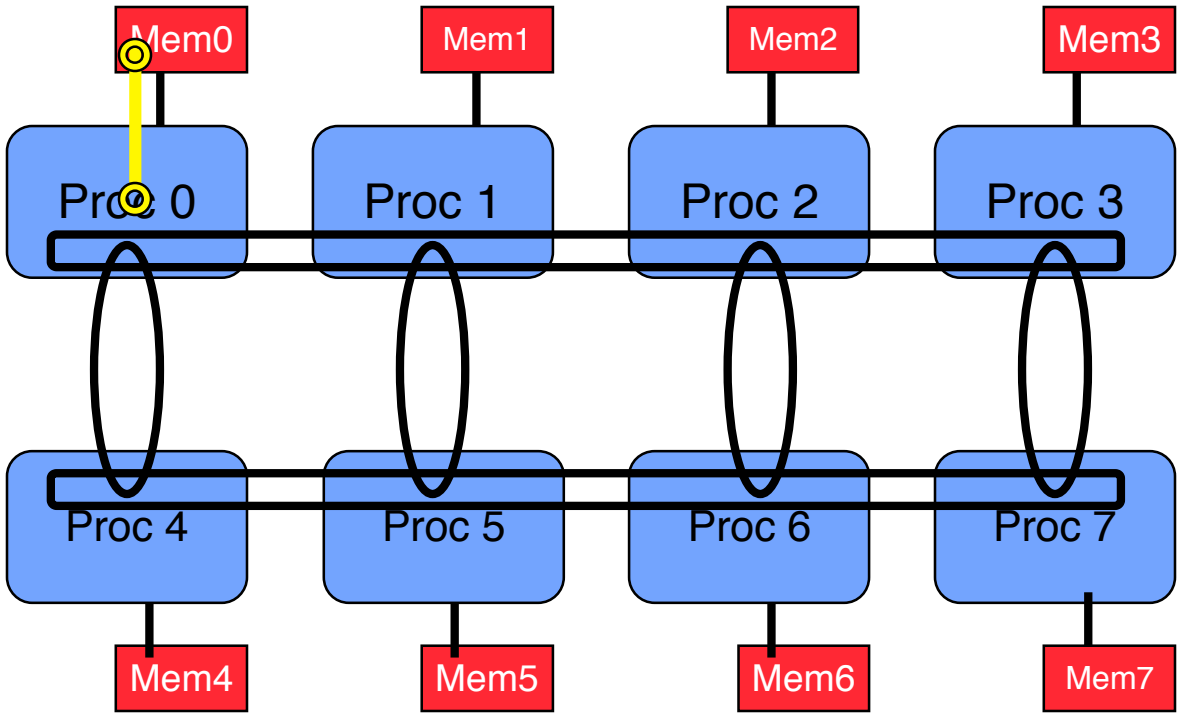
NERSC

U.S. Department of Energy
Office of Science

Page Mapping

Berkeley Lab

Proc0 Page#	Mem# *RR*	Mem# *MCM*
0	0	0
1	1	0
2	2	0
3	3	0
5	4	0
6	5	0
7	6	0
8	7	0
9	0	0
10	1	0
11	2	0
12	3	0
13	4	0
14	5	0
15	6	0
16	7	0



Linear Walk through Memory Addresses

- MCM affinity (really DCM affinity in this case)
- Pages assigned to mem ctrlr where first touched.
- Average latency 90ns + no fabric contention

NERSC

Memory Affinity Directives

- For processor-local memory affinity, you should also set environment variables to
  - MP_TASK_AFFINITY=MCM
  - MEMORY_AFFINITY=MCM
- For OpenMP need to eliminate memory affinity
  - Unset MP_TASK_AFFINITY
  - MEMORY_AFFINITY=round_robin *(depending on OMP memory usage pattern)*

U.S. Department of Energy
Office of Science

Large Pages

BERKELEY LAB

- Enable Large Pages
  - -blpdata *(at link time)*
  - Or *ldedit -blpdata <exename>* on existing executable
- Effect on STREAM performance
  - TRIAD without -blpdata: 5.3GB/s per task
  - TRIAD with -blpdata: 7.2 GB/s per task (6.9 loaded)

U.S. Department of Energy
Office of Science

A Short Commentary about Latency

BERKELEY LAB

- Little's Law: bandwidth * latency = concurrency

- For Power-X (some arbitrary) single-core:
  - 150ns * 20 Gigabytes/sec (DDR2 memory)
  - 3000 bytes of data in flight
  - 23.4 cache lines (very close to 24 memory request queue depth)
  - 375 operands must be prefetched to fully engage the memory subsystem
  - THAT'S a LOT of PREFETCH!!! *(esp. with 32 architected registers!)*

NERSC

U.S. Department of Energy
Office of Science

Deep Memory Request Pipelining Using Stream Prefetch

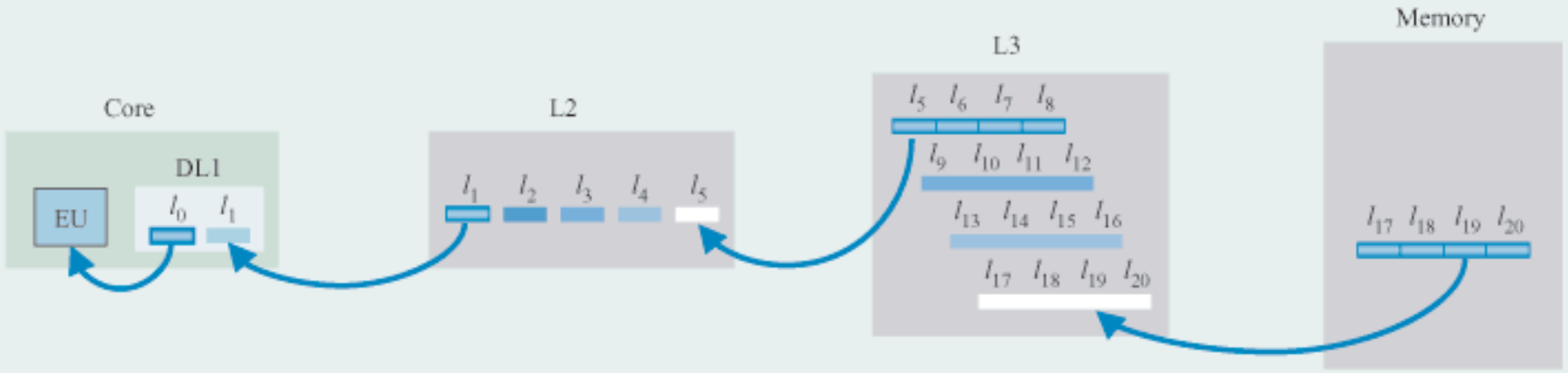
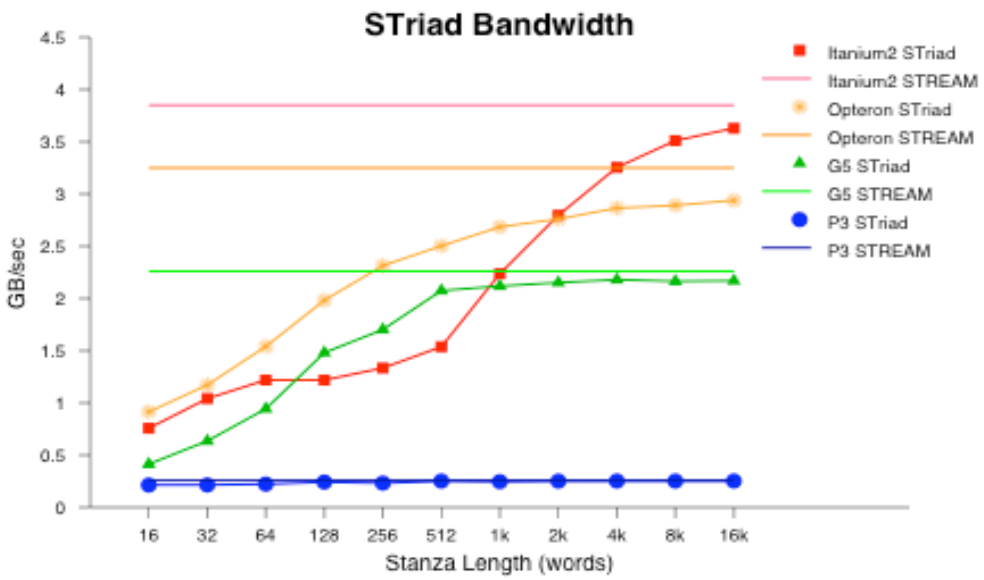


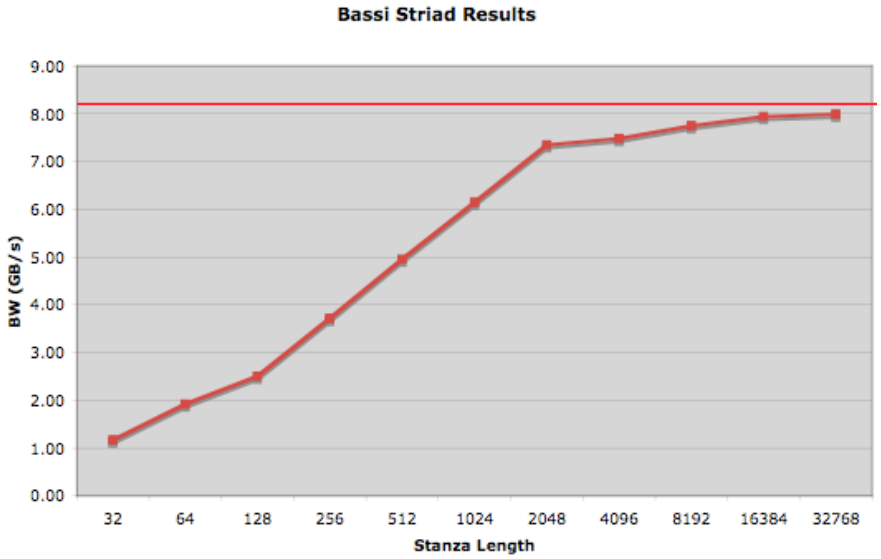
Image From IBM Power4 Redbook

U.S. Department of Energy
Office of Science

Stanza Triad Results

Berkeley Lab





- Perfect prefetching:
  - performance is independent of L, the stanza length
  - expect flat line at STREAM peak
  - our results show performance depends on L

NERSC

XLF Prefetch Directives

- DCBT (Data Cache Block Touch) explicit prefetch
  - Pre-request some data
  - !IBM PREFETCH_BY_LOAD(arrayA(I))
  - !IBM PREFETCH_FOR_LOAD(variable list)
  - !IBM PREFETCH_FOR_STORE(variable list)
- Stream Prefetch
  - Install stream on a hardware prefetch engine
  - Syntax: !IBM PREFETCH_BY_STREAM(arrayA(I))
  - PREFETCH_BY_STREAM_BACKWARD(variable list)
- DCBZ (Data Cache Block Zero)
  - Use for store streams
  - Syntax: !IBM CACHE_ZERO(StoreArrayA(I))
  - Automatically include using -qnopteovrlp option (*unreliable)*
  - Improve performance by another 10%

Power5: Protected Streams

- Protected Prefetch Streams
  - There are 12 filters and 8 prefetch engines
  - Need control of stream priority and prevent rolling of the filter list (takes a while to ramp up prefetch)
  - Helps give hardware hints about “stanza” streams
  - PROTECTED_UNLIMITED_STREAM_SET_GO_FORWARD(*variable, streamID*)
  - PROTECTED_UNLIMITED_STREAM_SET_GO_BACKWARD(*var,streamID*)
  - PROTECTED_STREAM_SET_GO_FORWARD/BACKWARD(*var,streamID*)
  - PROTECTED_STREAM_COUNT(*ncachelines*)
  - PROTECTED_STREAM_GO
  - PROTECTED_STREAM_STOP(*stream_ID*)
  - PROTECTED_STREAM_STOP_ALL
- STREAM_UNROLL
  - Use more aggressive loop unrolling and SW pipelining in conjunction with prefetch
- EIEIO (Enforce Inorder Execution of IO)