Performance Tuning the OpenEdge Database in the Modern World

Mike Furgal
PROGRESS Bravepoint – Database Services

Introduction

Mike Furgal

- Progress Employee since 1989
- Developer of the OpenEdge database
- Joined Bravepoint in 2012
- Heads up Database Services
  - Including Managed Database Services

Bravepoint

- Largest Progress/OpenEdge consulting firm
- Founded in 1987
- Purchased by Progress in April 2014
- Specializes in all things OpenEdge
  - Database Services
  - Programming
  - QAD
- Pro2SQL
  - Real-time Replication to SQL Target

Abstract

Modern computing demands large memory, many CPUs and elaborate storage.

How do you meet these demands for your OpenEdge environment? In this talk we give you advice, tips, useless information, and pointers on the technologies you can use to meet your requirements. Among other things, we will discuss NUMA (Non-Uniform Memory Access), RAID, SSD, and some of the more advanced OpenEdge RDBMS tuning techniques.

What's in it for you? We'll address that question in a discussion of benefits.

Performance tuning is not

just about software configuration

and turning knobs.

Situation:

Your server is 5 years old.

Vendor support fees rise.

Parts prices rise.

Parts are harder to find.

With what do you replace old server ???

Good news !

Hardware is cheap.

Your new server will have:

- Processors
- Memory
- Storage
- Software

Numbers you should know

(from Jeff Dean @ google)

thing	time
Read or write L1 cache memory	0.5 ns
Branch mispredict	5 ns
Mutex lock/unlock	100 ns
Read 1 byte from main memory	100 ns
Send 2K bytes over 1 Gbps network	20,000 ns
Read 1 MB sequentially from memory	250,000 ns
Round trip packet within same datacenter	500,000 ns
Disk seek	10,000,000 ns
Read 1 MB sequentially from network	10,000,000 ns
Read 1 MB sequentially from disk	30,000,000 ns
Send packet CA -> Netherlands -> CA	150,000,000 ns
1 second	1,000,000,000 ns

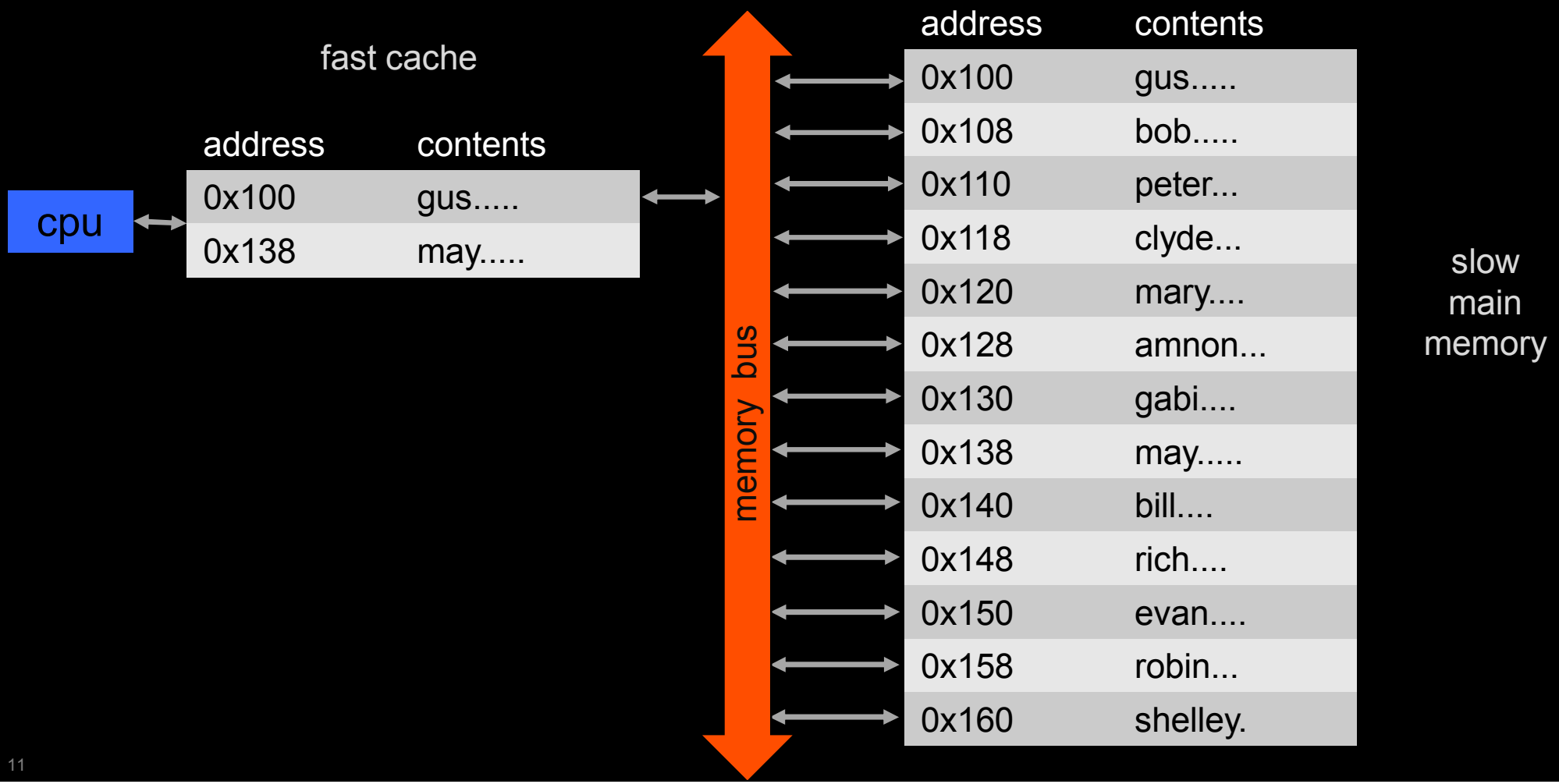
Processors

Modern processors are very fast.

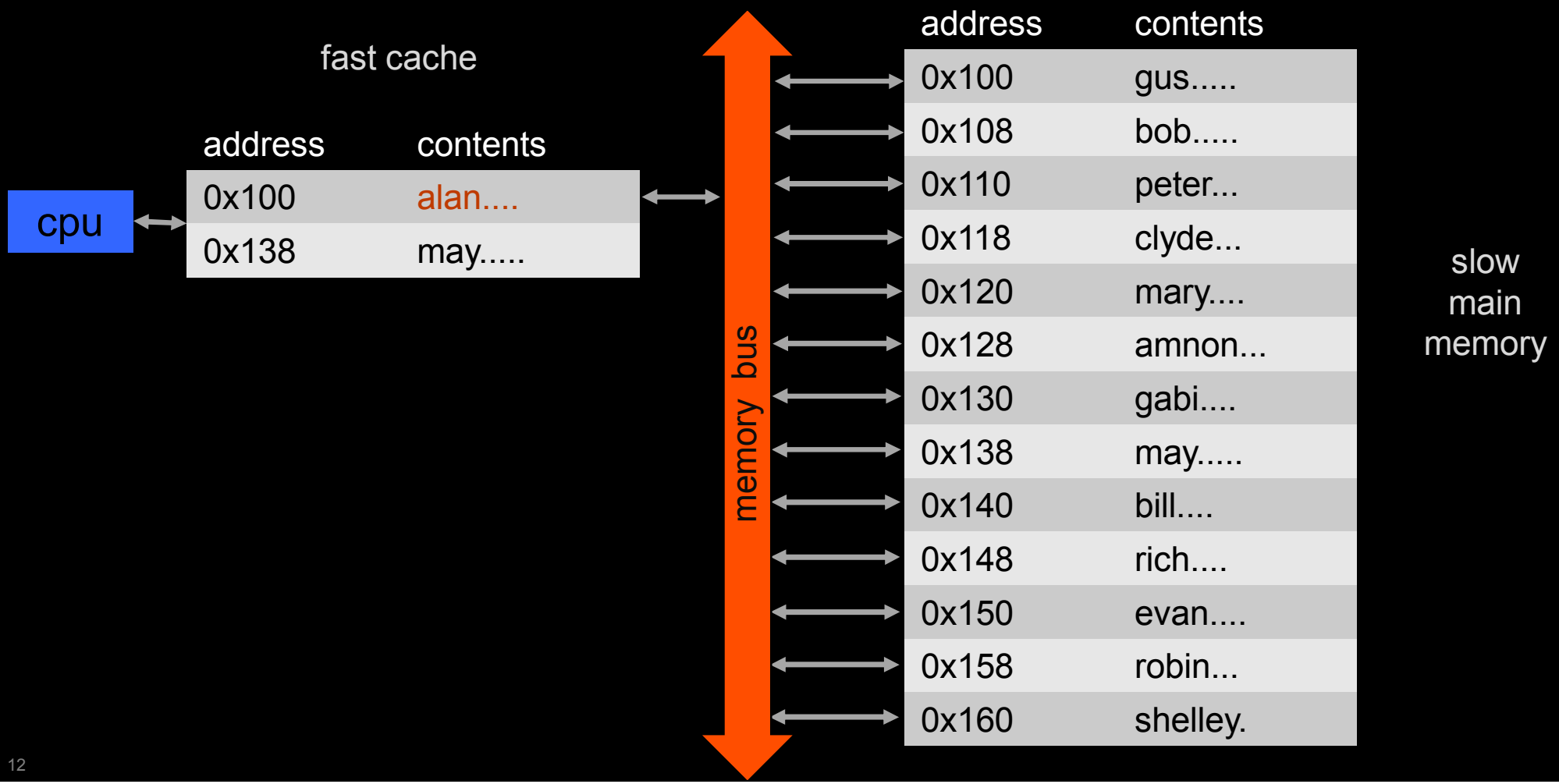
Single cpu machines hardly exist anymore.

You can have way more cpu power than you can ever use

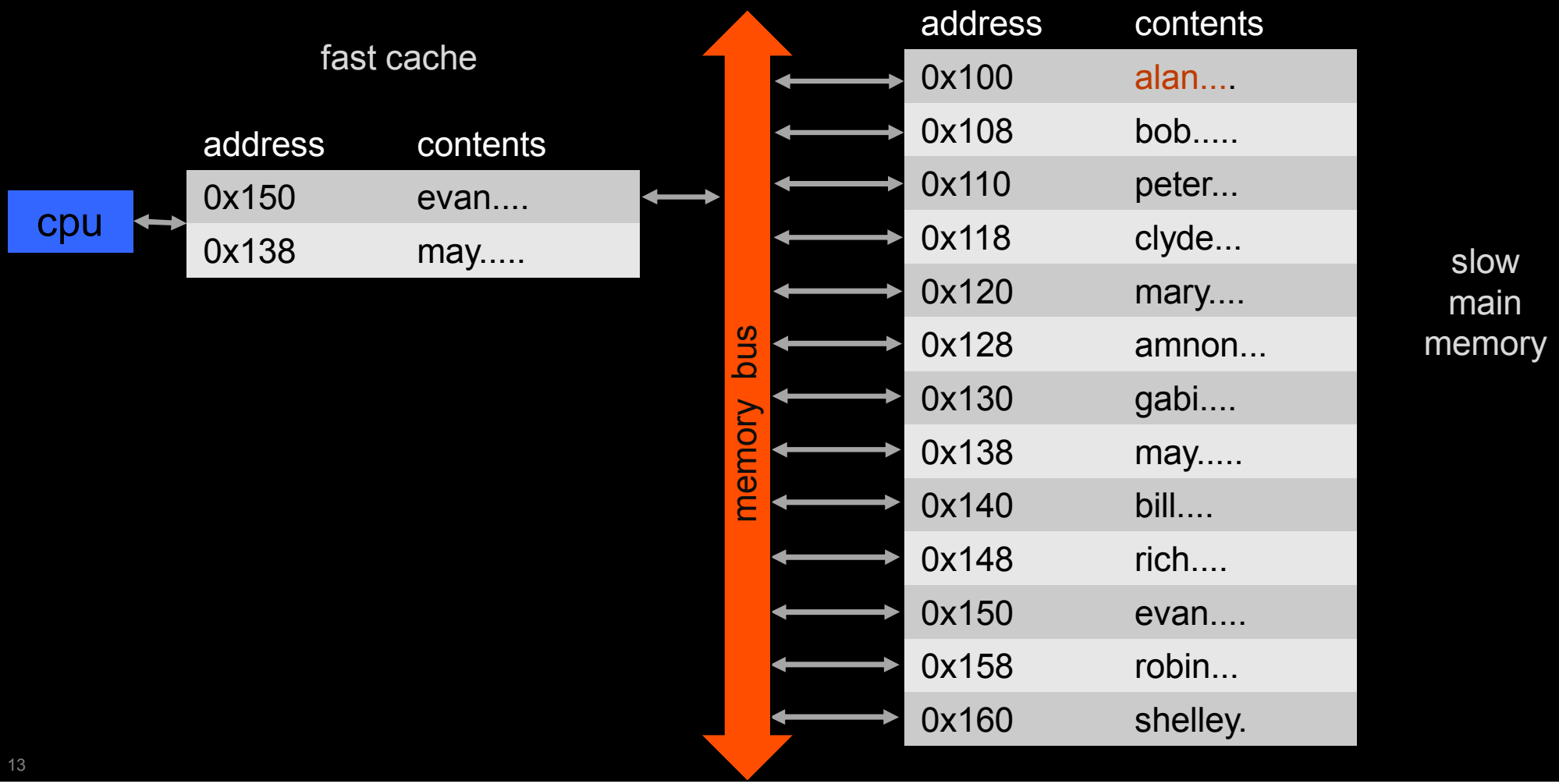
Simple single processor architecture
one level high speed cache memory



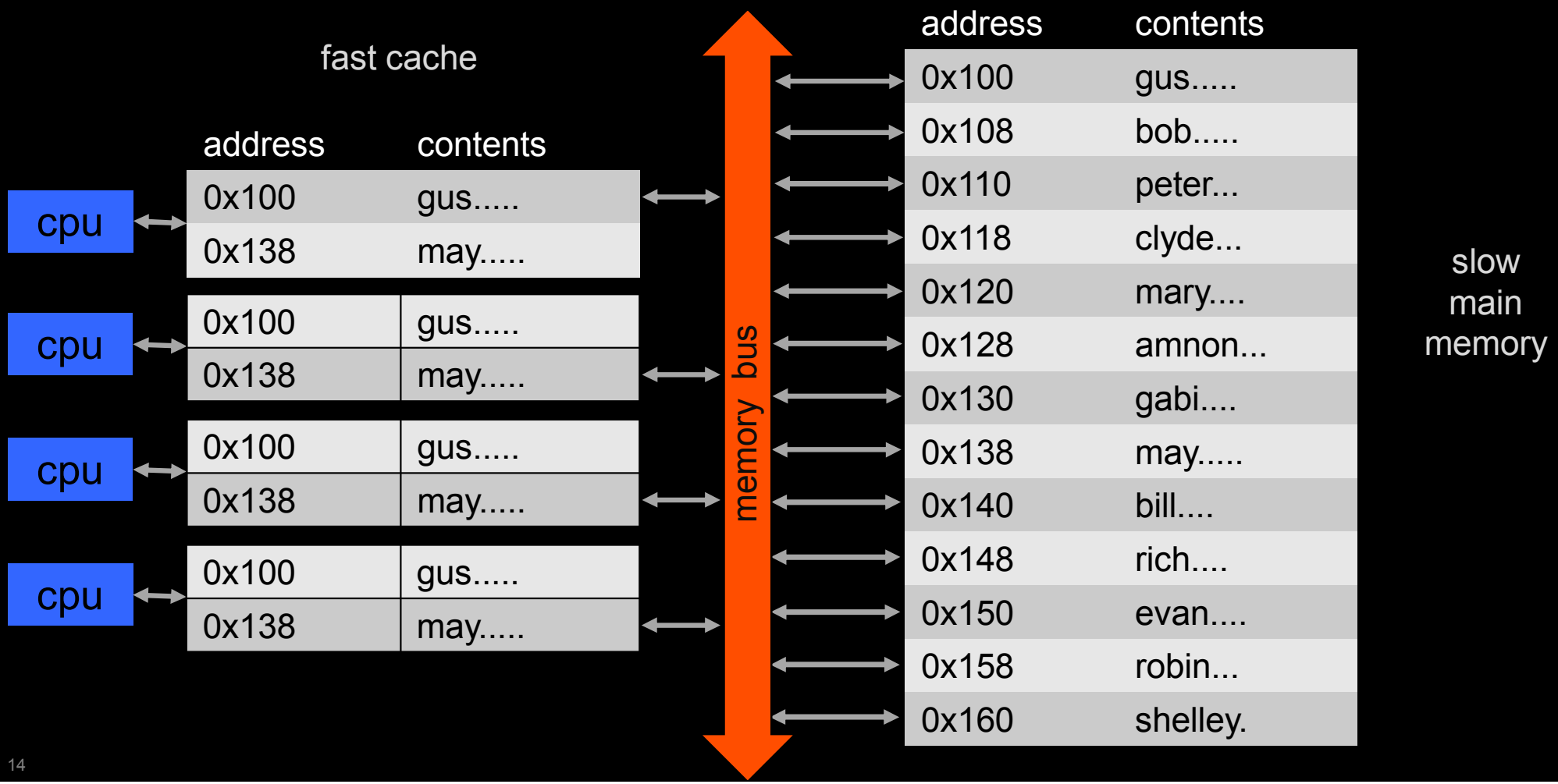
Simple single processor architecture
one level high speed cache memory



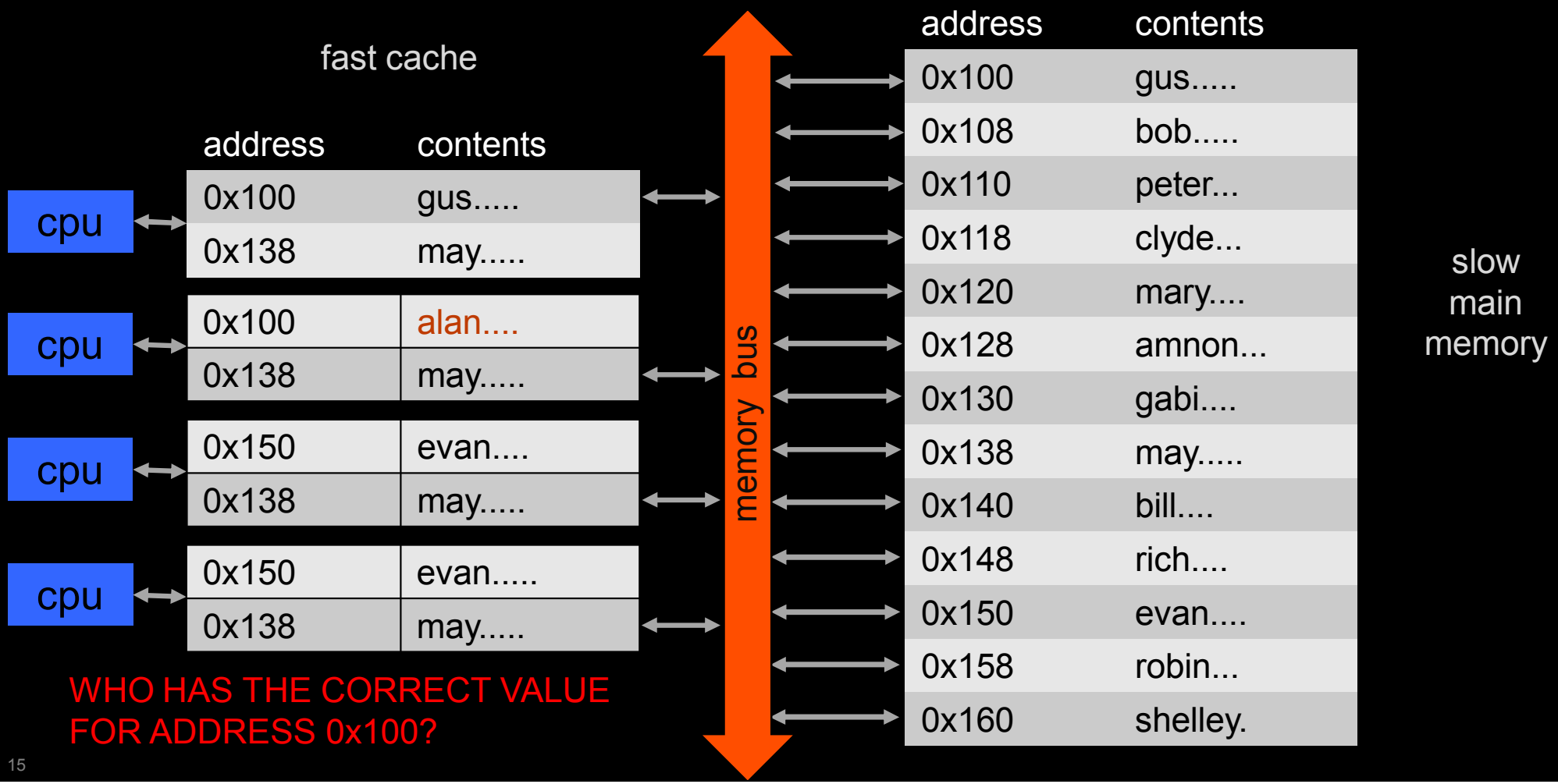
Simple single processor architecture
one level high speed cache memory



Multiprocessor caches



Multiprocessor caches



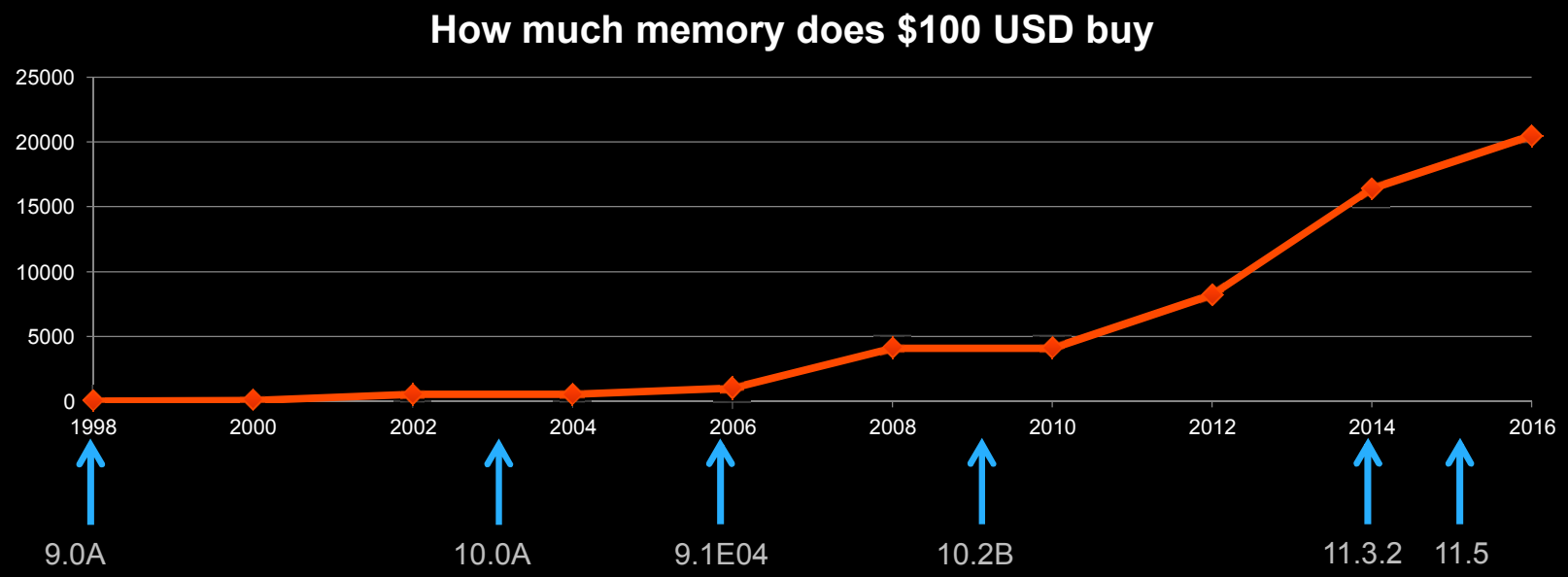
WHO HAS THE CORRECT VALUE FOR ADDRESS 0x100?

A techniques to avoid Cache Coherency issues

Lessen the number of processes connected directly to shared memory.

Main Memory

Memory prices have dropped significantly over the past years. For example in the year 2000, 64 MB of memory cost $100 USD. In 2010 for $100 USD you could get 4 GB of memory. Today (2015) that same $100 USD gets you about 20 GB of memory.



Main Memory

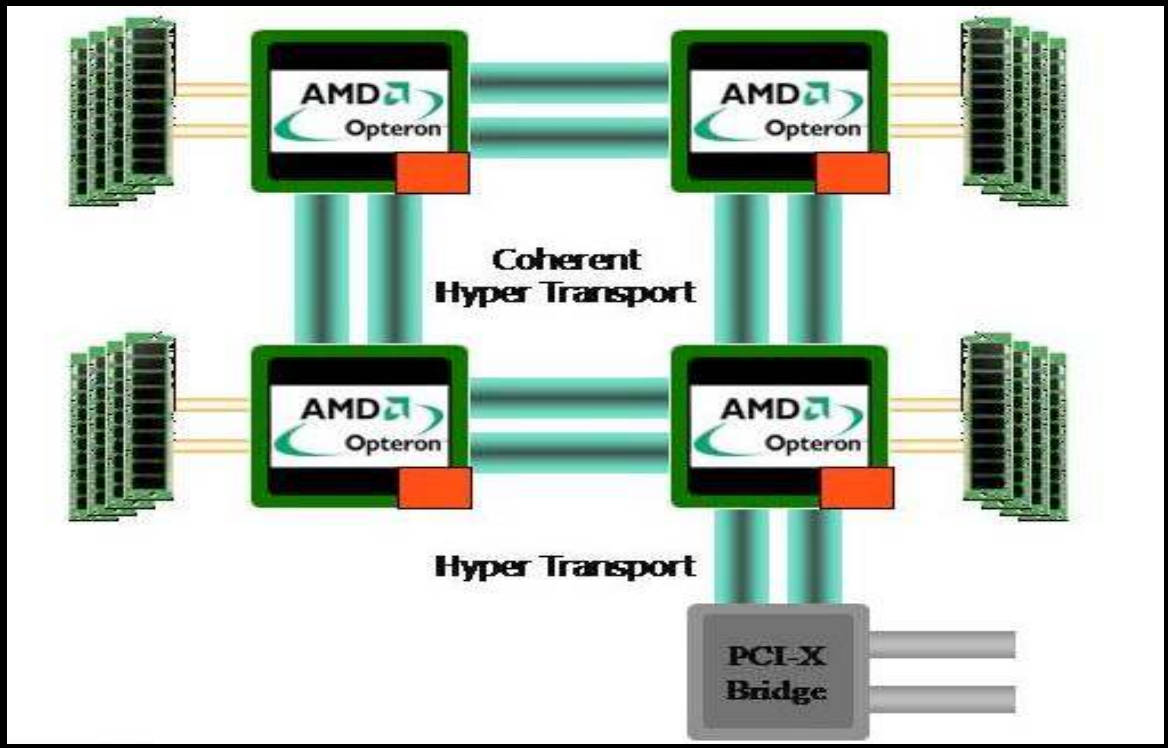
The *least* expensive way
to enhance performance.

Buy as much as you can.

NUMA

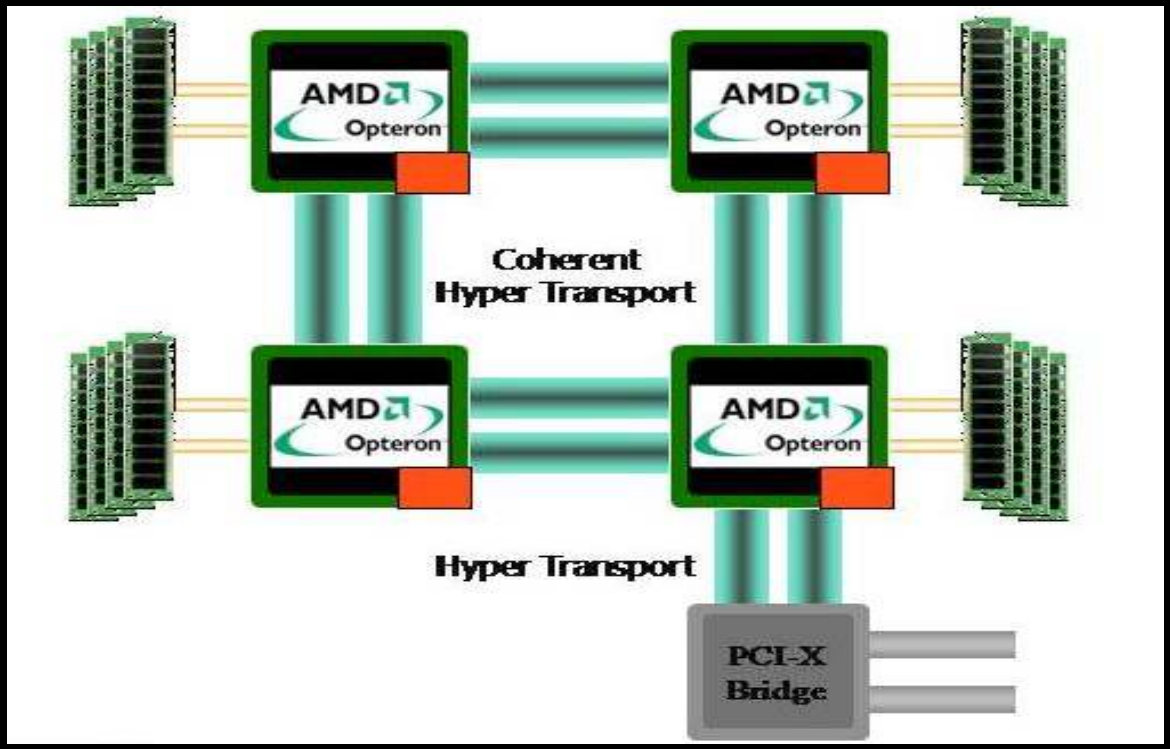
NUMA stands for Non-Uniform Memory Access

In layman's terms, a NUMA machine is the coupling of several machines in a single physical unit, running a single Operating System. Like a "cluster" (if you squint).

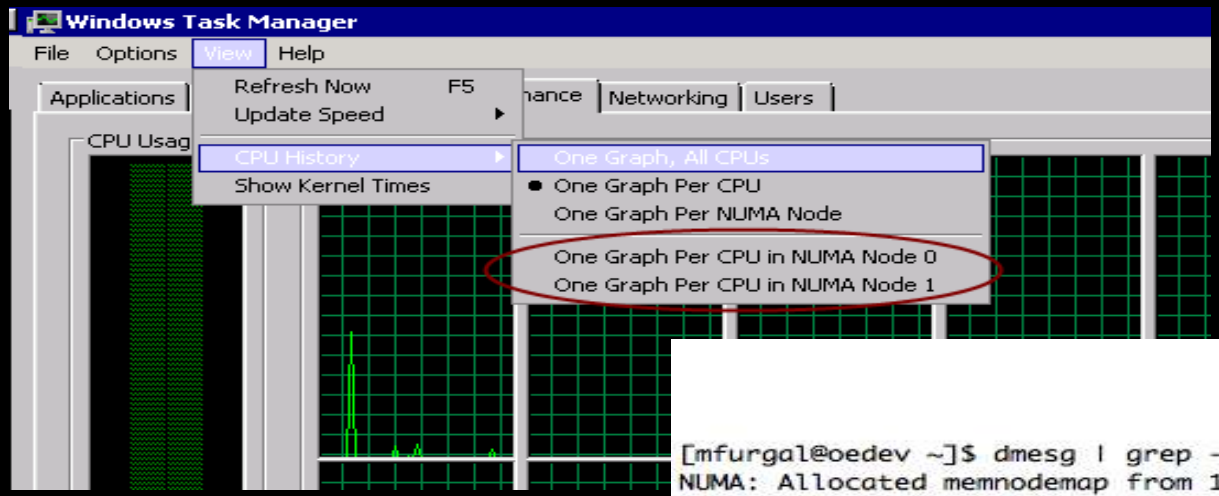


The NUMA Quotient

This is the time it takes for a CPU to read memory on a remote node as compared to reading memory locally



How do you know if you have a NUMA machine?




```
[mfurgal@oedev ~]$ dmesg | grep -i numa
NUMA: Allocated memnodemap from 11000 - 12040
NUMA: Using 20 for the hash shift.
[mfurgal@oedev ~]$ ssh pmtest
mfurgal@pmtest's password:
Last login: Fri Aug 29 13:08:32 2014 from 172.16.61.127
[mfurgal@pmtest ~]$ dmesg | grep -i numa
No NUMA configuration found
[mfurgal@pmtest ~]$
```

So now you know you have a NUMA machine.

Is all hope lost?

On some machines you can pin memory and processes to a particular node.

On some you can disable nodes but may lose memory too

Bottom line – don’t buy a NUMA machine

If you have a NUMA machine, redeploy it for VMWare, etc

Storage

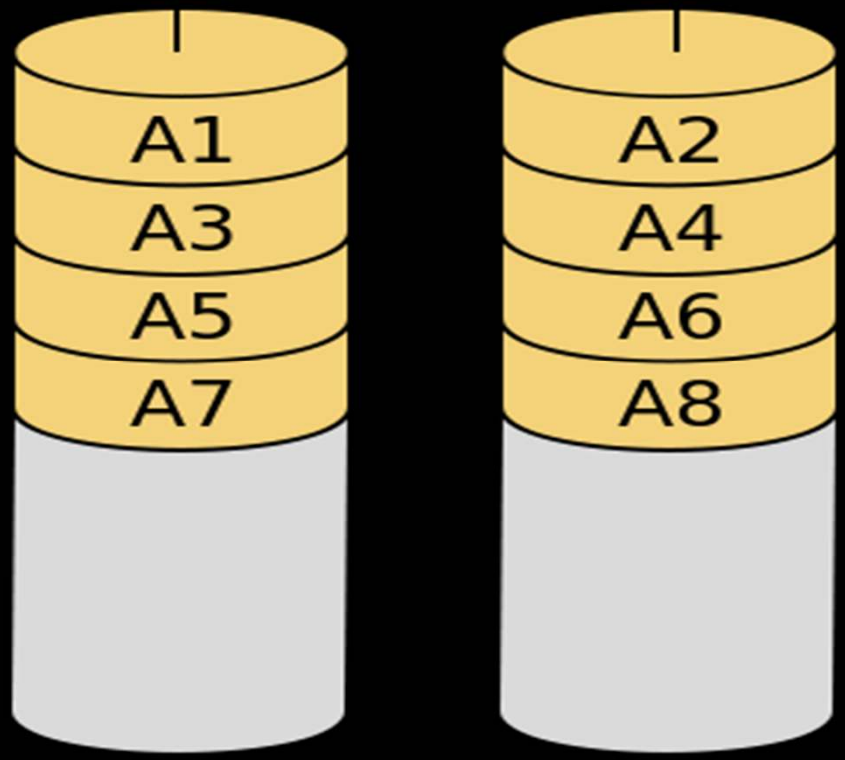
RAID

RAID
Why?

raid diagrams are from wikipedia

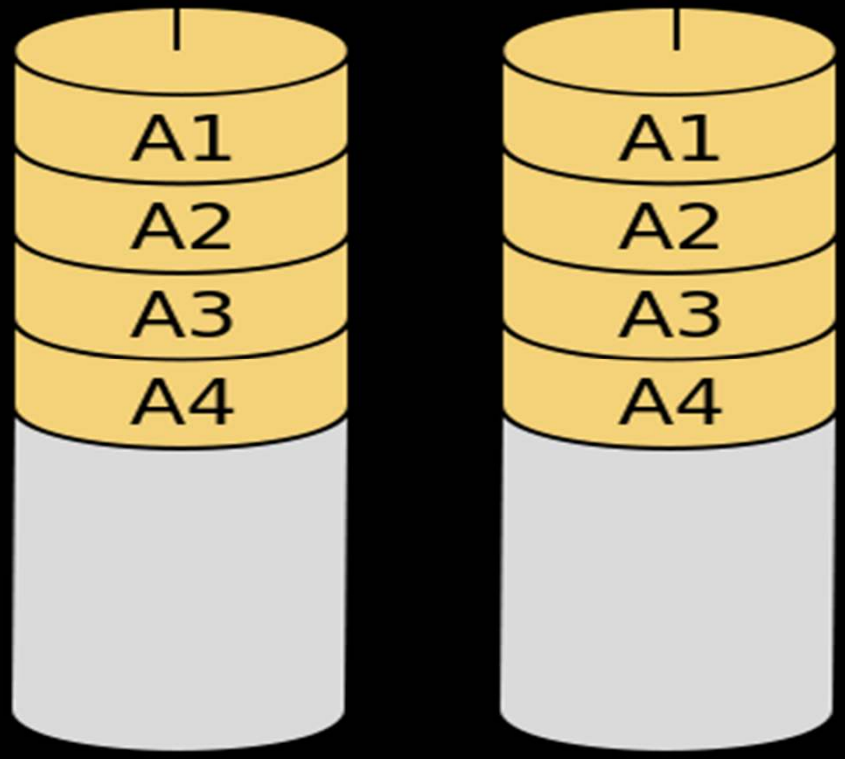
RAID 0: block striping

performance but NO reliability



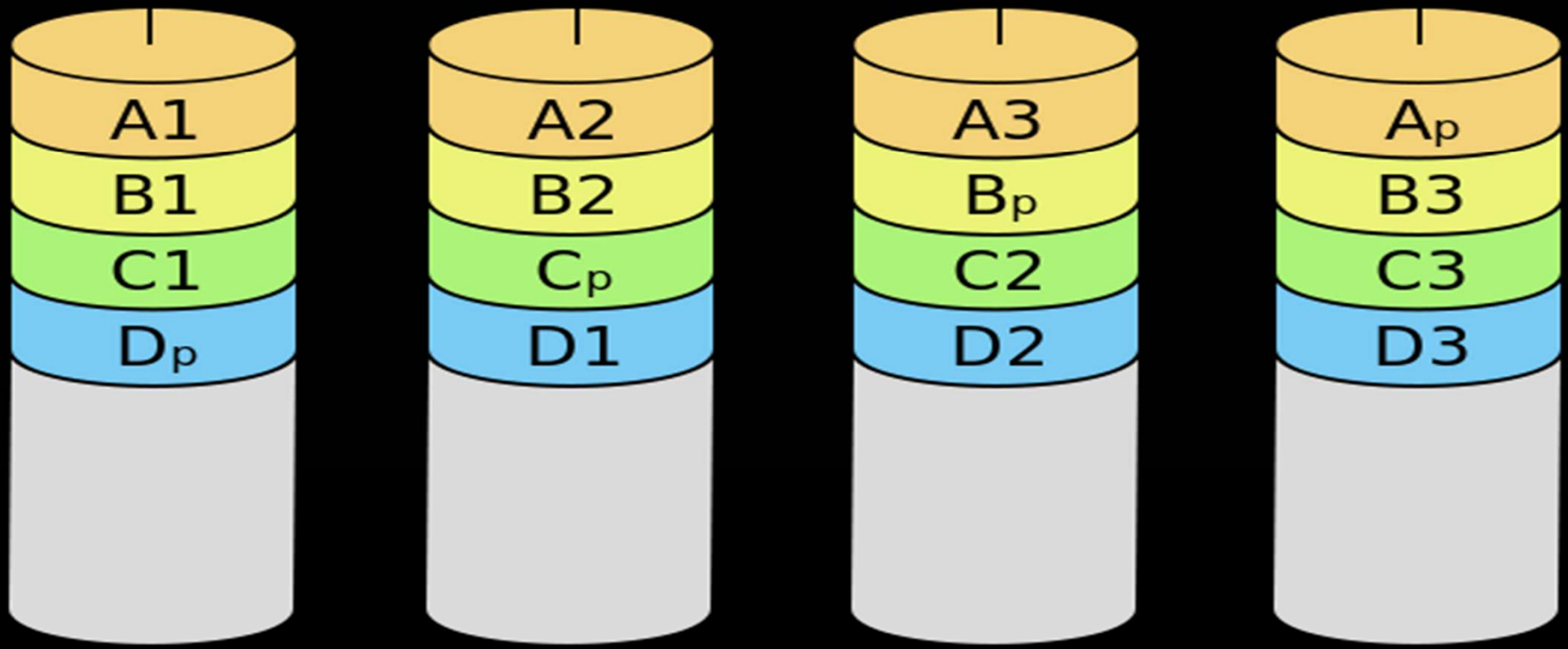
RAID 1: disk mirroring

reliability – two copies



RAID 5: block striping with parity

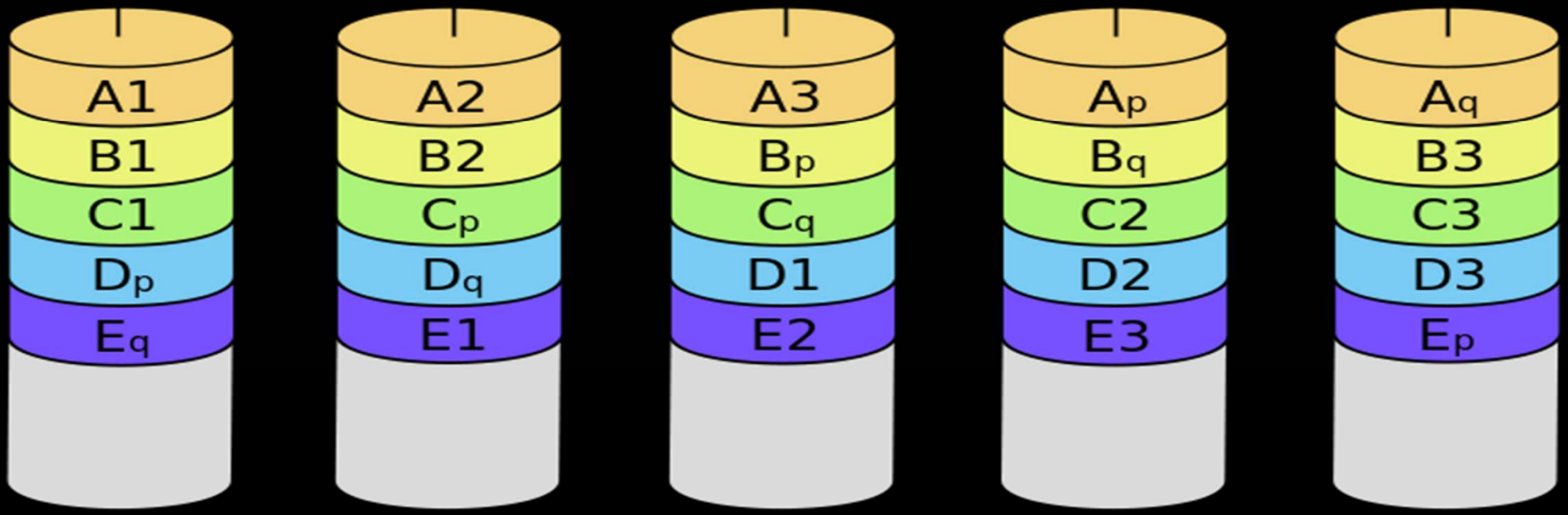
reliability and bad performance



all writes must update 2 drives

RAID 6: block striping with two parity disks

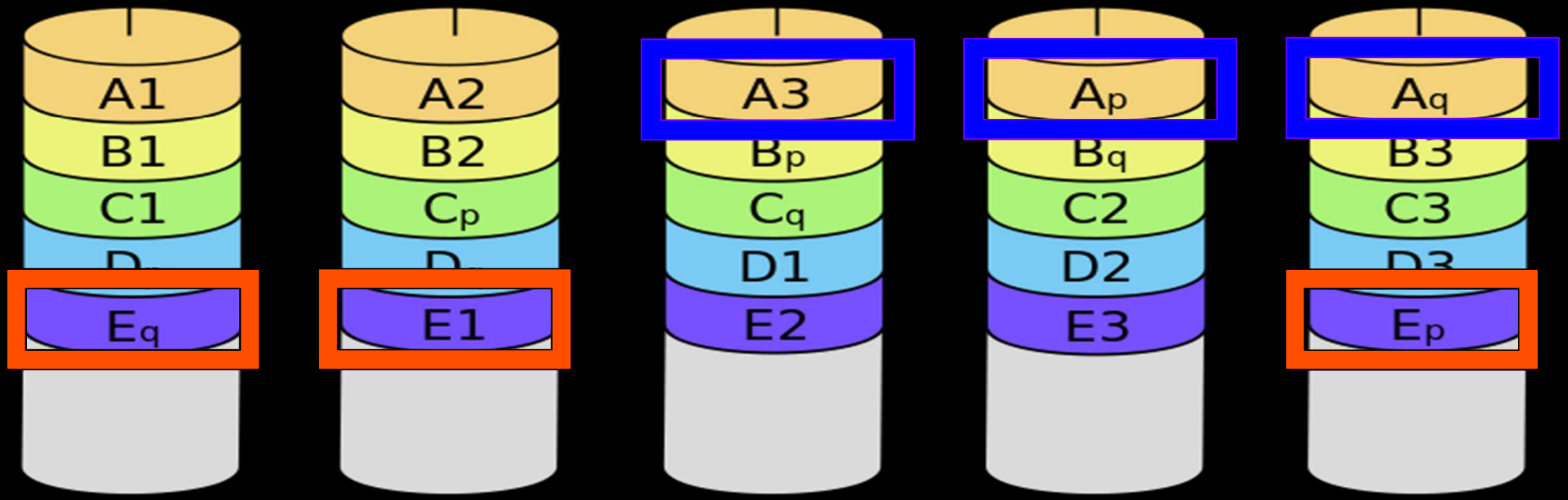
reliability and worse performance



all writes must update 3 drives

RAID 6: block striping with two parity disks

reliability and worse performance



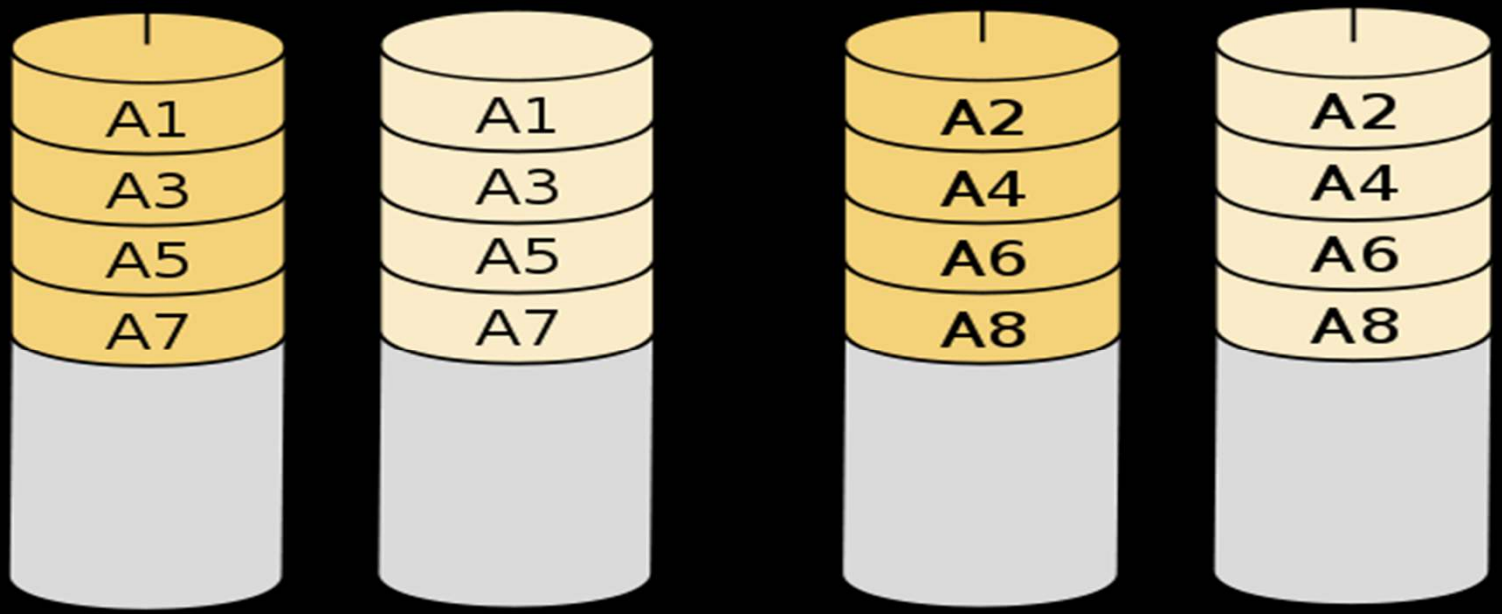
all writes must update 3 drives

write to block E1 has to wait for write to block A3

RAID 10: disk mirroring and block striping

reliability – two copies

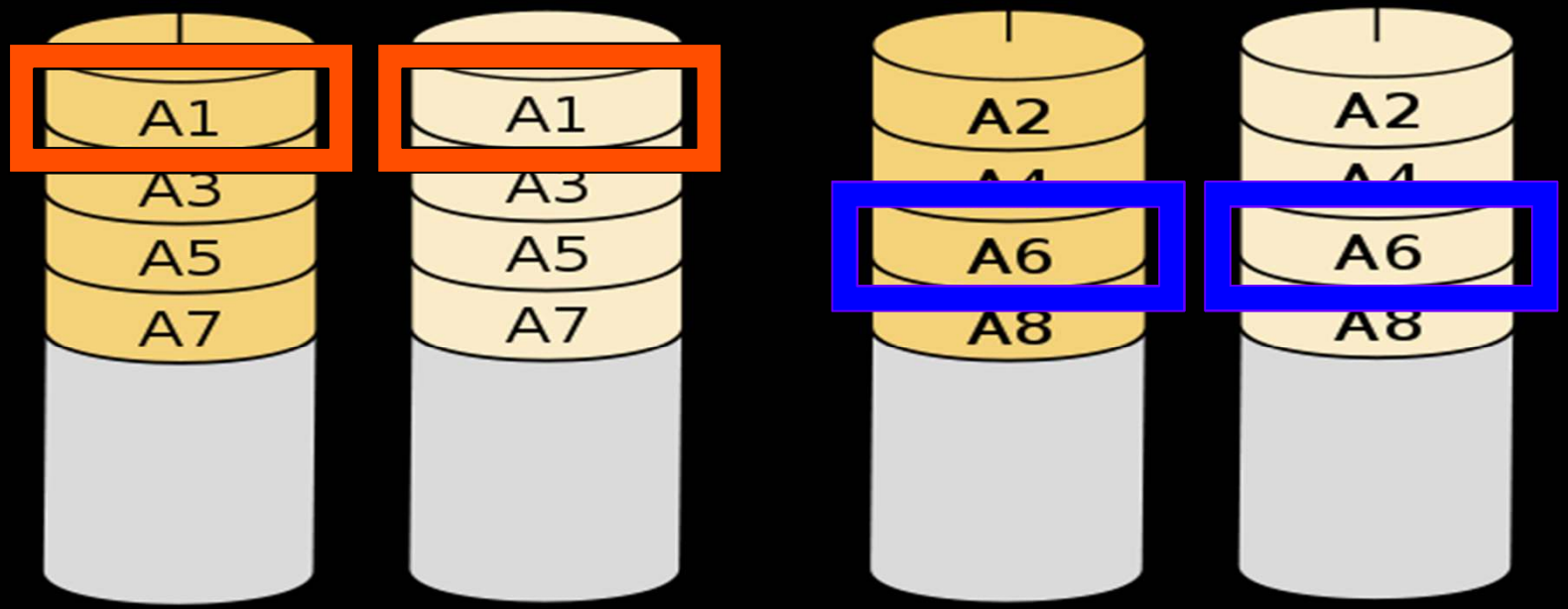
performance – data spread over multiple drives



RAID 10: disk mirroring and block striping

reliability – two copies

performance – data spread over multiple drives



RAID choices

Type	Description	Use ?
RAID 0	Block striping (no redundancy at all)	Bad
RAID 1	Mirroring	OK
RAID 10	Block striping + mirroring	Excellent
RAID 2	Bit level striping, dedicated parity	Bad
RAID 3	Byte level striping, dedicated parity	Bad
RAID 4	Block striping, dedicated parity	Bad
RAID 5	Block striping with striped parity	Poor
RAID 6	Block striping with dual striped parity	Poor
RAID 60, 6+, DP, etc.	Marketing	Poor

RAID choices – only 1 good one

Type	Description	Use ?
~~RAID 0~~	~~Block striping (no redundancy at all)~~	~~Bad~~
~~RAID 1~~	~~Mirroring~~	~~OK~~
RAID 10	Block striping + mirroring	Excellent
~~RAID 2~~	~~Bit level striping, dedicated parity~~	~~Bad~~
~~RAID 3~~	~~Byte level striping, dedicated parity~~	~~Bad~~
~~RAID 4~~	~~Block striping, dedicated parity~~	~~Bad~~
~~RAID 5~~	~~Block striping with striped parity~~	~~Poor~~
~~RAID 6~~	~~Block striping with dual striped parity~~	~~Poor~~
~~RAID 60, 6+, DP, etc~~	~~Marketing~~	~~Poor~~

Advancements in technology can never make a silk purse from the RAID 5 / 6 sow's ear. Vendors can't fool mother nature !!!

Local disks will beat SAN storage

APJ PROGRESS SPARK

SSD

SSD

- Fetching a record that is already in the database buffer pool is 75 times faster than SSD !!!!
- Prices have dropped – a LOT. Low end is $0.50 per gigabyte
- Reliability is now very good – better than spinning rust
- SSD devices are fast, and getting faster
- Use Mirrored pairs (RAID 1) – NO RAID 5
  or any striping
- When you need to replace one, you may not be able to get matching units anymore.

Time to grow a 96 MB file		
Disk Type	Duration	Speed
Spinning Disk	7 – 10	9 - 13 MB/Sec
SSD	1 - 2	43 – 96 MB/Sec

… in Big B You Should Trust!

Layer	Time	# of Recs	# of Ops	Cost per Op	Relative
Progress to –B	0.96	100,000	203,473	0.000005	1
-B to FS Cache	10.24	100,000	26,711	0.000383	75
FS Cache to SAN	5.93	100,000	26,711	0.000222	45
-B to SAN Cache*	11.17	100,000	26,711	0.000605	120
SAN Cache to Disk	200.35	100,000	26,711	0.007500	1500
-B to Disk	211.52	100,000	26,711	0.007919	1585

** Used concurrent IO to eliminate FS cache*

courtesy of Tom Bascom

Mid-range

server replacement example

Name	Qty	Value
CPU (32 CPUs)	4	Intel Xeon E5 4603, 8 cores
RAM (32 GB)	8	1866MT/s 4 GB RDIMM
Ether	1	Intel GB Ethernet Card
Disk Controller	1	PERC H10
Storage, hot plug	8	146 GB 15,000 rpm SAS
Stuff	?	dual psu, case, power cord, etc.
Operating system	1	Linux, not included

Select Components

1. COMPONENTS | 2. SERVICES & ACCESSORIES

PowerEdge R820

Starting Price	$12,962.00
Instant Savings	$3,637.09
Subtotal	**$9,324.91**

Price is in USD

Software:

Modern OpenEdge RDBMS

Use it on your new server

Advanced Tuning Techniques

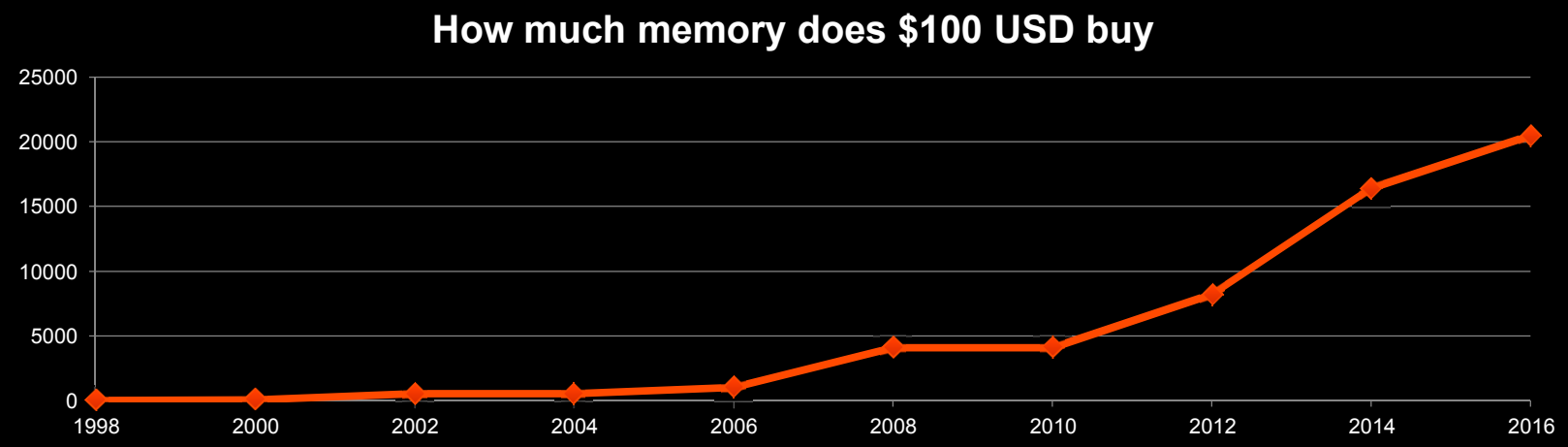
Get Current.

Better be on 10.2B08 or later

Get Current.

9.1E is over 10 years old!

10.1C is over 6 years old!



-lruskips

-B2

-napmax

index rebuild

Index Rebuild Performance (OE 10.2B06, OE 11.2)

-TB	*sort block size (8K – 64K, note new limit)*	64
-datascanthreads	*# threads for data scan phase*	1.5 X #CPUs
-TMB	*merge block size (default -TB)*	64
-TF	*merge pool fraction of system memory (in %*	80%
-mergethreads	*# threads per concurrent sort group merging*	X -threadnum = 1.5 X #CPUs
-threadnum	*# concurrent sort group merging*	2 or 4
-TM	*# merge buffers to merge each merge pass*	32
-rusage	*report system usage statistics*	-rusage
-silent	*a bit quieter than before*	-silent

Index Rebuild Performance (OE 10.2B06, OE 11.2)

-TB	*sort block size (8K – 64K, note new limit*	
-datascanthreads	*# threads for data scan phase*	
-TMB	*merge block size (de*	64
-TF	*merge pool fr*	80%
-mergethreads	*# t* *roup merging*	X -threadnum = 1.5 X #CPUs
-threadnum	*merging*	2 or 4
-TM	*to merge each merge pass*	32
-rusage	*t system usage statistics*	-rusage
-silent	*a bit quieter than before*	-silent

12 ½ hours ▯ 2 ½ hours

5X improvement!

PROGRESS SPARK

-omsize

How to manage Object Mapping Cache

- Do I have a problem?
  - Check latch statistics

```
define variable prev-latches as integer.
repeat:
   find _latch where _latch-name = "MTL_OM".
   display _Latch-Name
           _Latch-Lock   /* # times latch acquired */
           _Latch-Wait   /* # time conflict occurred */
           _Latch-Lock - prev-latches label "latch/sec".
           prev-latches = _Latch-Lock.
   pause 1.
end.
```

Client database-request statement caching

Procedure Call Stack

- Top is last procedure executed
- Bottom is first procedure executed
- Top down, newest to oldest
- One time full stack
- Continuous full stack
- Continuous current location

Top

Newest

#	Procedure Name	File Name
19 :	reallyLongNamedInternalProcedure3	proctestb.r
12 :	reallyLongNamedInternalProcedure2	proctestb.r
5 :	reallyLongNamedInternalProcedure1	proctesta.r
445 :	reallyLongNamedInternalProcedure0	proctesta.r
1 :	/usr1/stmtest/p72340_Untitled1.ped	

Oldest

Bottom

table partitioning

Summary

- Hardware has changed a lot
  - It's cheaper, it's faster
  - Watch out for buying too many CPUs as it may be NUMA
- While storage has changed, the RAID recommendations have not
  - RAID 10 is still required for good performance
- For best results, use the latest OpenEdge version


APJ
PROGRESS SPARK