Putting

Modern Hardware Performance

to Work

- or: ”This is not your fathers computer”

Poul-Henning Kamp - phk@FreeBSD.org

```
$ cd /FreeBSD/src/sys
$ cvs annotate |& fgrep -c '(phk'
135926
```

#N kpd
#S Zilog 8000/12, Zeus 3.21
#O Kuwait Petroleum Denmark
#C Poul-Henning Kamp
#E phk@kpd ...!mcvax!diku!dkuug!kpd!phk
#T +45 1 32 20 85
#P Kv{sthusgade 3, DK-1003 Klbenhavn K.
#L 12 35 E / 55 35 N
#W k...@diku.dikunet Feb 18 10:05:16 1987

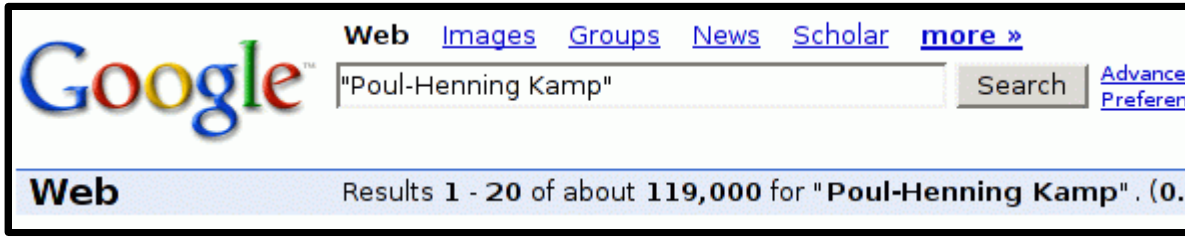
D-Link Firmware Abuses Open NTP Servers

Posted by ScuttleMonkey on Fri Apr 07, '06 10:36 AM
from the **frustration-in-a-box** dept.

DES writes

> "FreeBSD developer and NTP buff Poul-Henning Kamp runs a stratum-1 NTP server specifically for the benefit of networks directly connected to the Danish Internet Exchange (DIX). Some time last fall, however, D-Link started including his server in a hardcoded list in their router firmware. Poul-Henning

http://www.bikeshed.org




```
root:$1$IgMjWs2L$Nu12OCqeKEPPioOI7TmUN1:0:0::0:0:Charlie &:/root:/bin/csh
```

```
pub   1024R/0358FCBD 1995-08-01
      Key fingerprint = A3 F3 88 28 2F 9B 99 A2  49 F4 E2 FA 5A 78 8B 3E
uid                  Poul-Henning Kamp <phk@FreeBSD.org>
```

```
/*
 * ----------------------------------------------------------------------
 * "THE BEER-WARE LICENSE" (Revision 42):
 * <phk@FreeBSD.ORG> wrote this file. As long as you retain this notice you
 * can do whatever you want with this stuff. If we meet some day, and you think
 * this stuff is worth it, you can buy me a beer in return Poul-Henning Kamp
 * ----------------------------------------------------------------------
 */
```

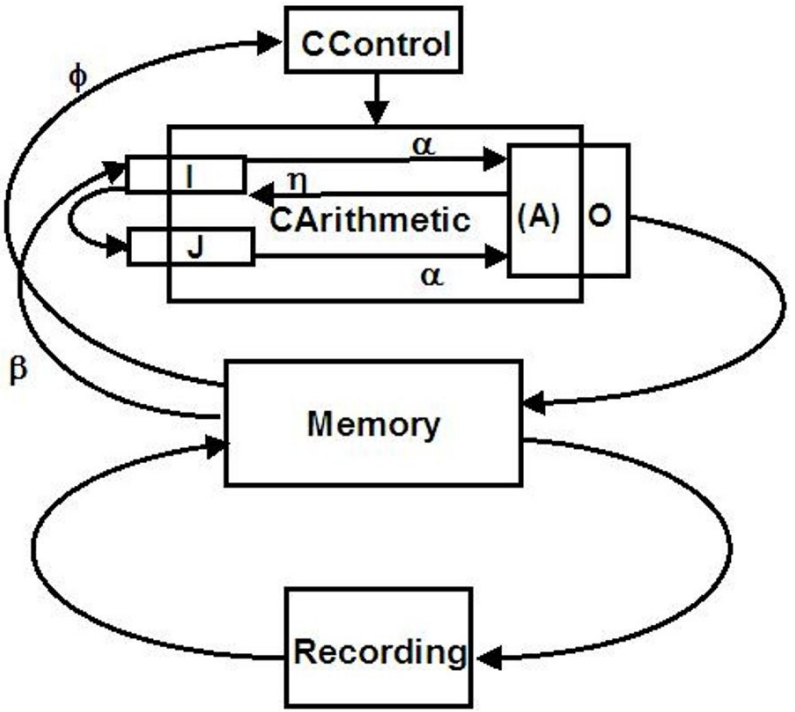
What happened ?

What can we do about it ?

Does it matter ?

(The Varnish story)

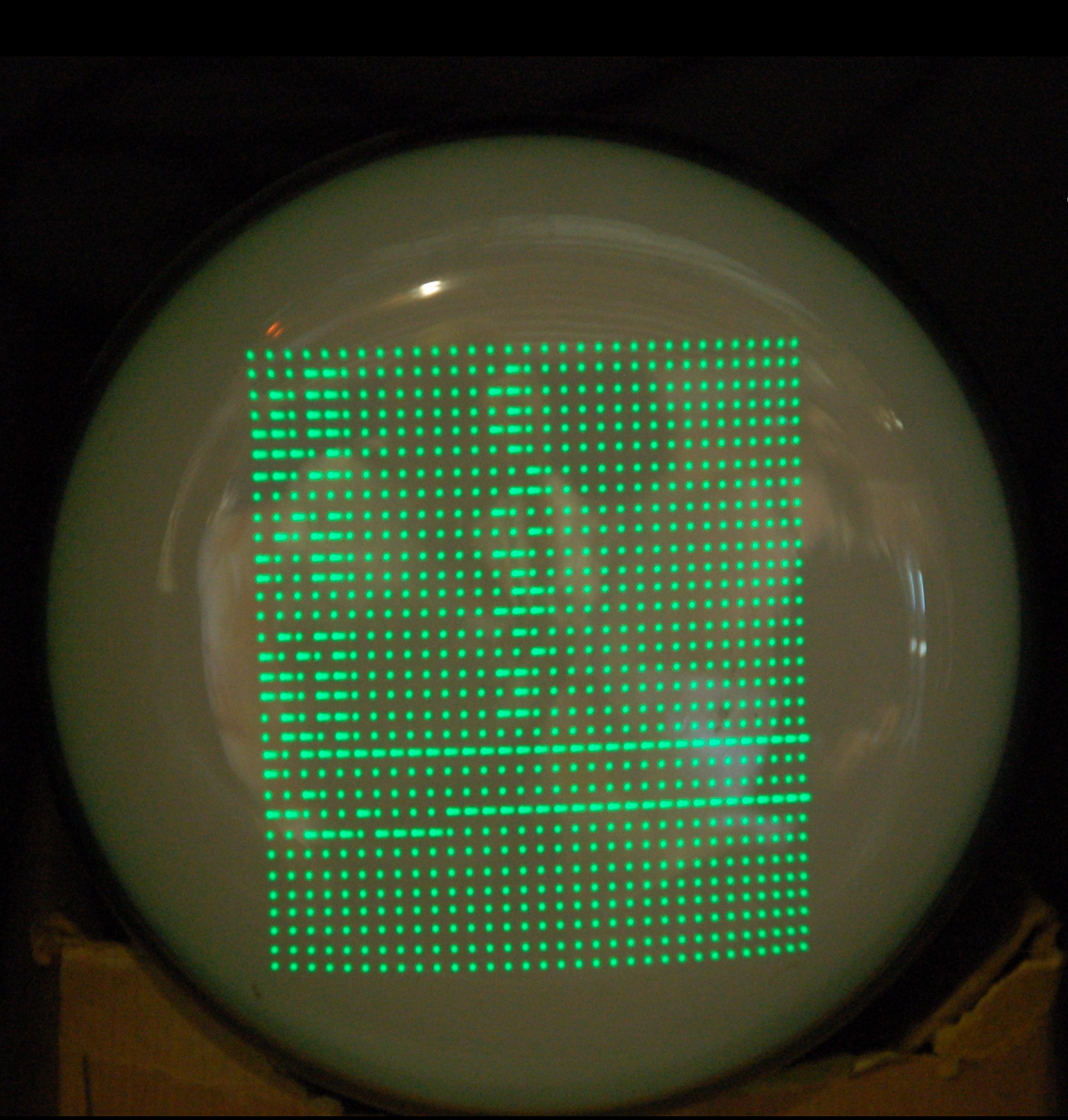
John von Neumann *et al.*, EDVAC Architecture



+	A ⟵ I + J
-	A ⟵ I - J
*	A ⟵ A + I*J
/	A ⟵ I/J
i	A ⟵ I
j	A ⟵ J
s	A ⟵ (A >= 0 ? I : J)

msm 98

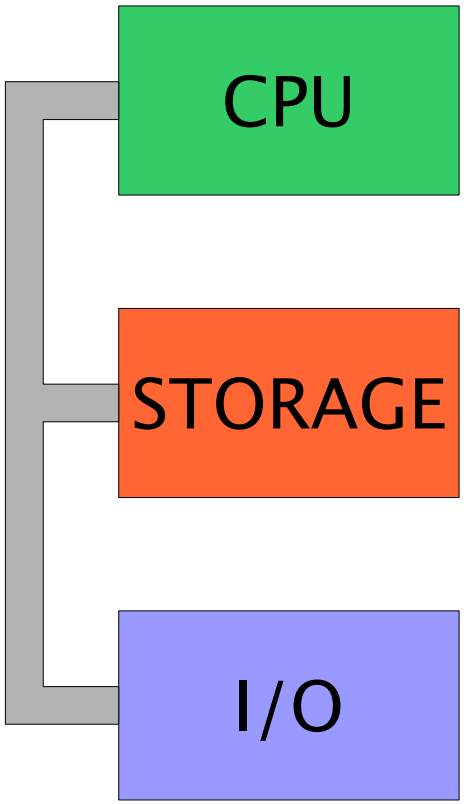

The Manchester ”Baby” (Working replica)



21 june 1948

The worlds first computer program runs on the worlds first Von Neuman architecture computer.

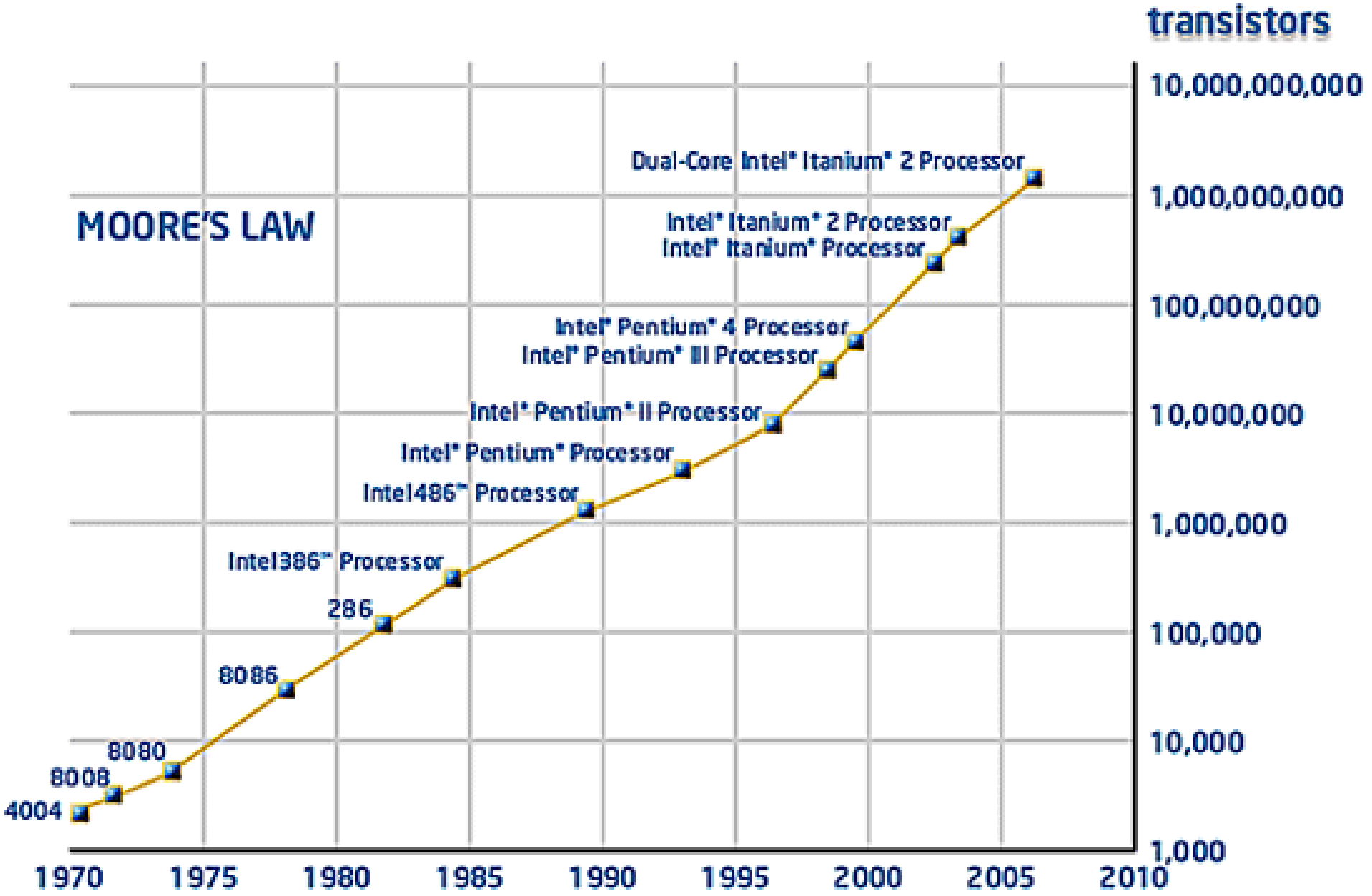
32 bits
32 words
7 instructions

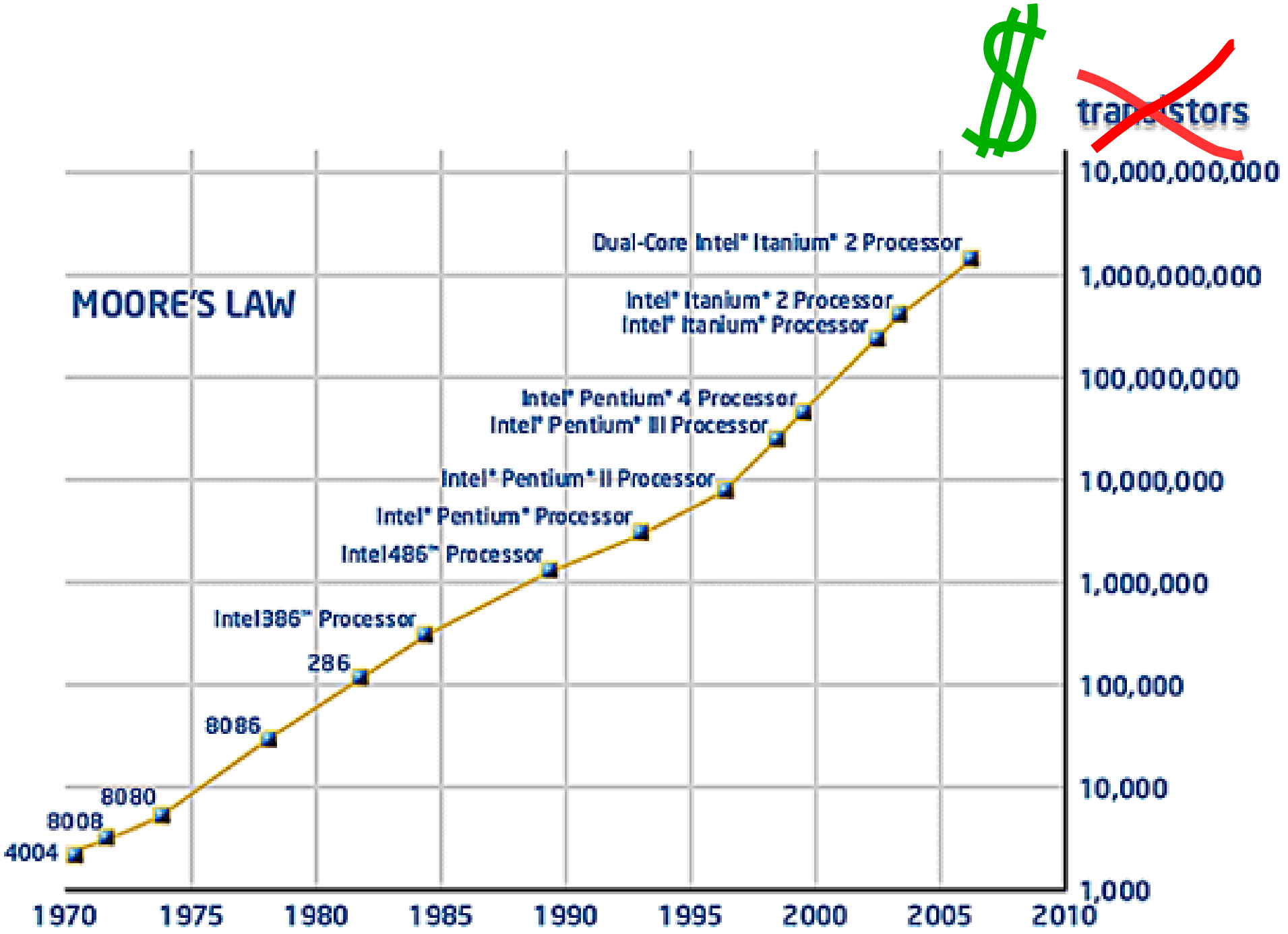


4 → 8 → 16 → 32 → 64 bits

CRT → Hg → Core → SRAM → DRAM

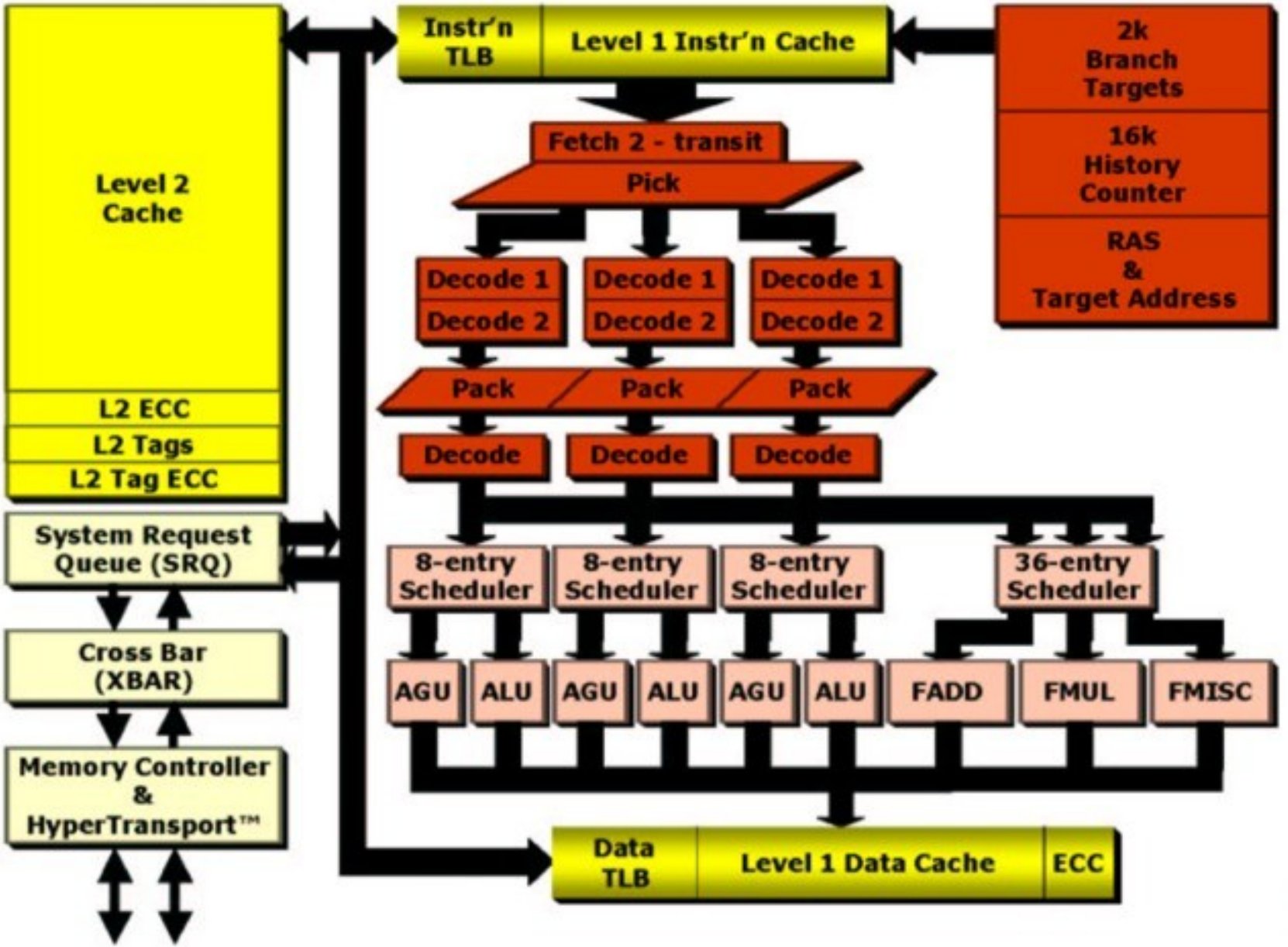
Paper → Magtape → Disk (24”...1.8”) → Flash

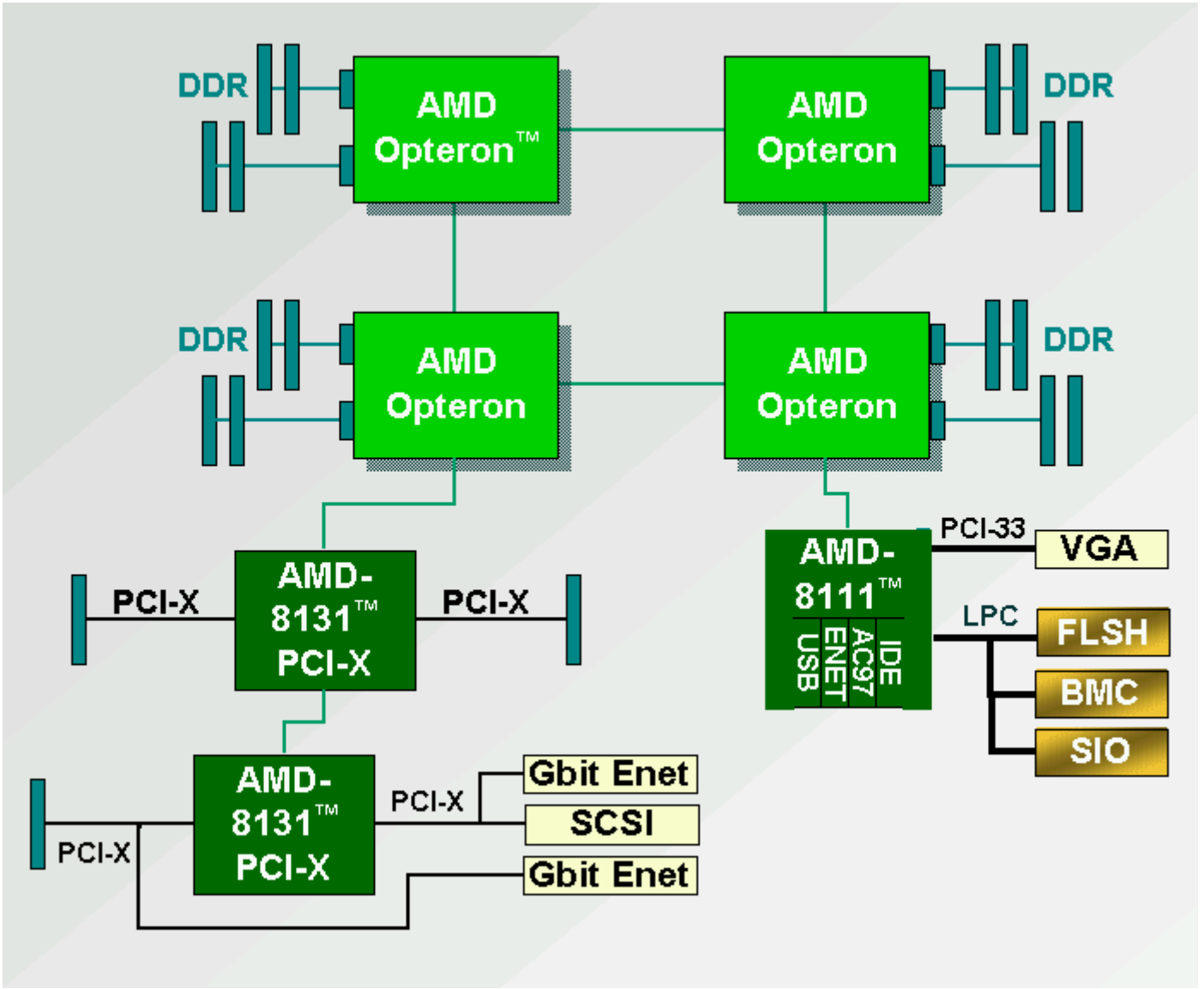

transistors
MOORE'S LAW
10,000,000,000
1,000,000,000
100,000,000
10,000,000
1,000,000
100,000
10,000
1,000
Dual-Core Intel® Itanium® 2 Processor
Intel® Itanium® 2 Processor
Intel® Itanium® Processor
Intel® Pentium® 4 Processor
Intel® Pentium® III Processor
Intel® Pentium® II Processor
Intel® Pentium® Processor
Intel486™ Processor
Intel386™ Processor
286
8086
8080
8008
4004
1970
1975
1980
1985
1990
1995
2000
2005
2010


$
transistors
MOORE'S LAW
Dual-Core Intel® Itanium® 2 Processor
Intel® Itanium® 2 Processor
Intel® Itanium® Processor
Intel® Pentium® 4 Processor
Intel® Pentium® III Processor
Intel® Pentium® II Processor
Intel® Pentium® Processor
Intel486™ Processor
Intel386™ Processor
286
8086
8080
8008
4004
10,000,000,000
1,000,000,000
100,000,000
10,000,000
1,000,000
100,000
10,000
1,000
1970
1975
1980
1985
1990
1995
2000
2005
2010

Processor Core Overview

AMD




DDR
AMD Opteron™
AMD Opteron
DDR
DDR
AMD Opteron
AMD Opteron
DDR
PCI-X
AMD-8131™ PCI-X
PCI-X
AMD-8111™
IDE
AC97
ENET
USB
PCI-33
VGA
LPC
FLSH
BMC
SIO
AMD-8131™ PCI-X
PCI-X
Gbit Enet
SCSI
Gbit Enet
PCI-X

Time Sharing

↓

Protection

↓

Address Mapping

↓

Virtual Memory

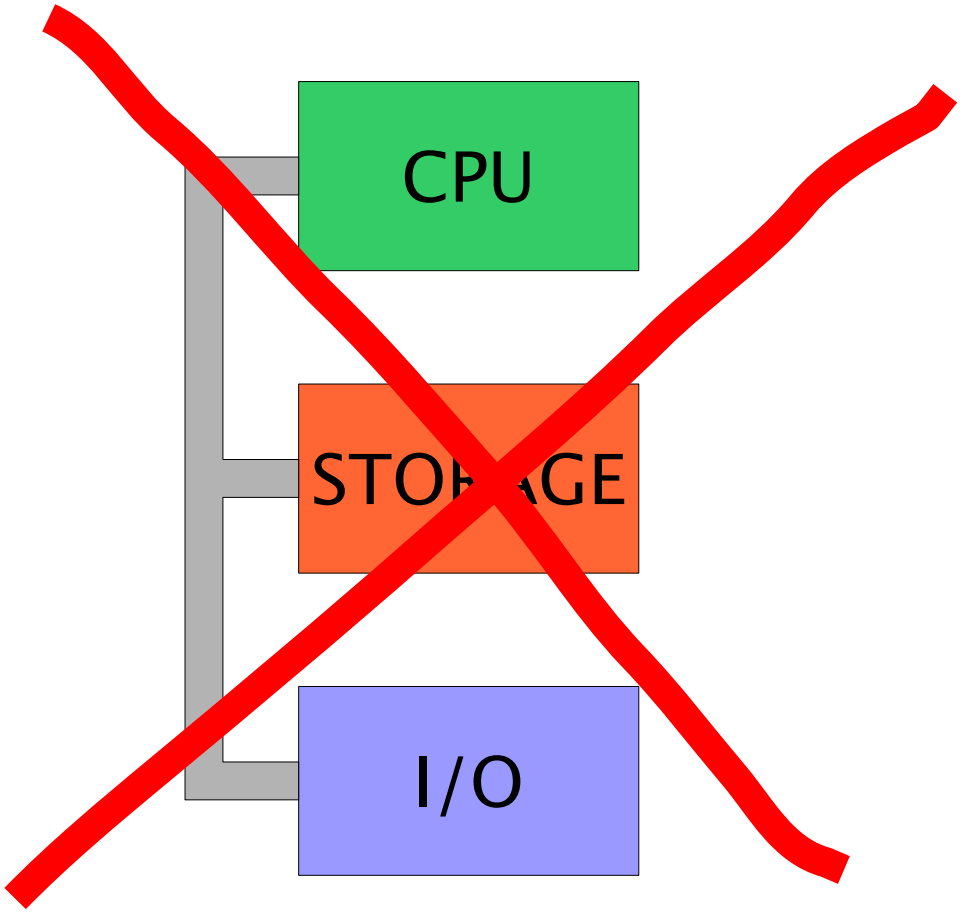
↓

Virtual Machines

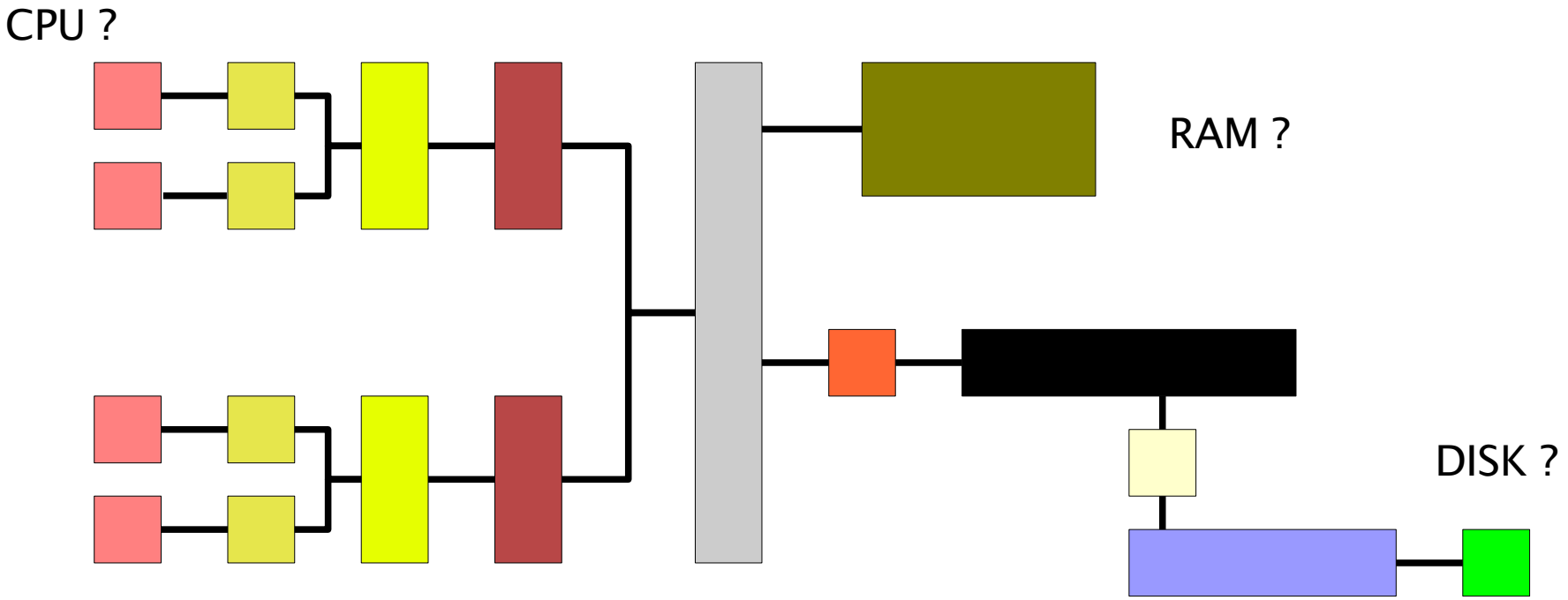
↓

Virtual Instruction Set

Your fathers computer:



Not your fathers computer:



There is no memory, it's all caches.

I/O is horribly expensive

Multiprogramming

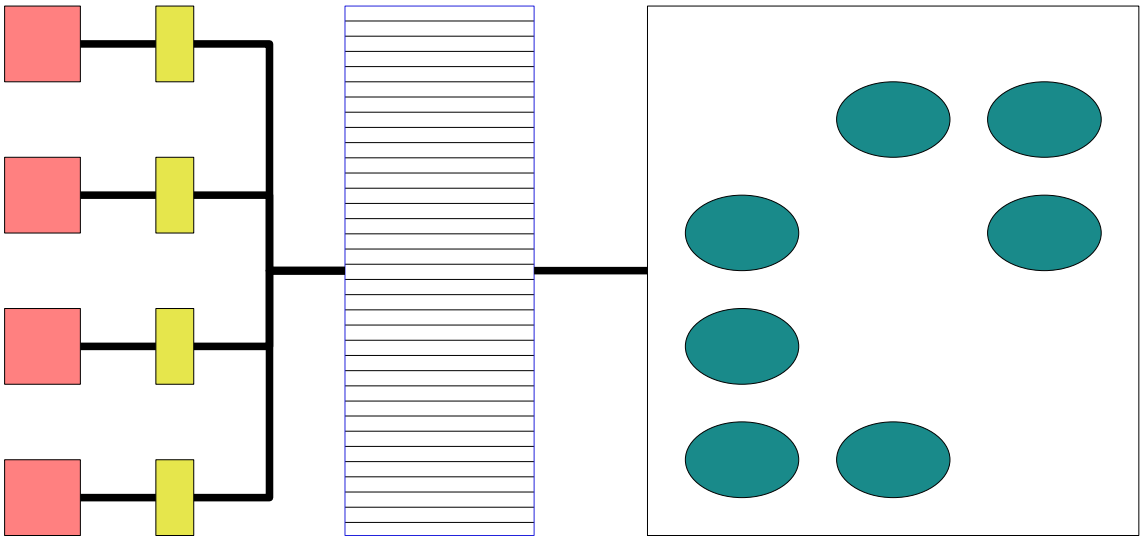

Unlimited*
Virtually FREE memory
offer good from 1980 onwards
16EB (16M GB)
* Max 4GB per process

CPUs/Cores

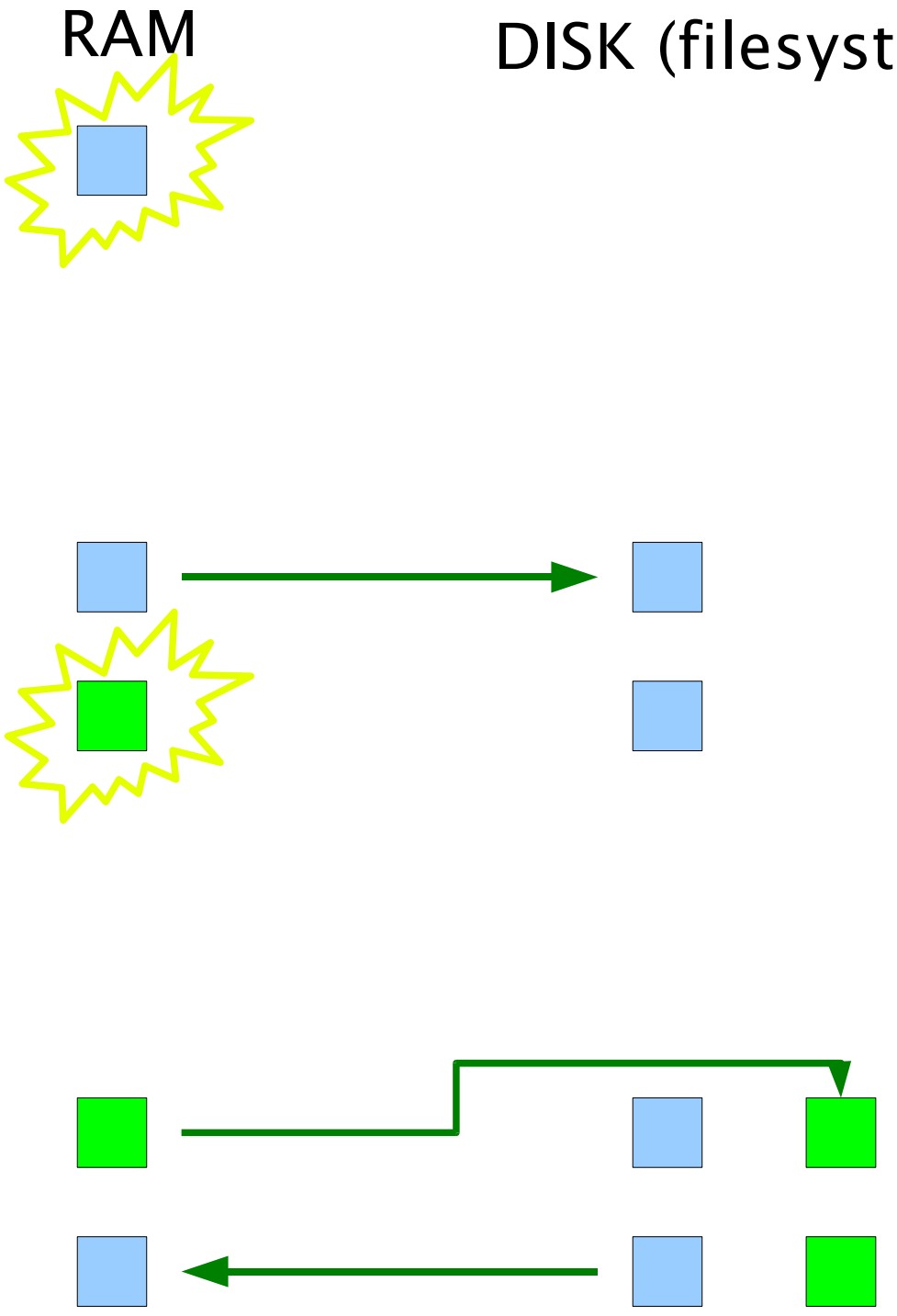
Caches

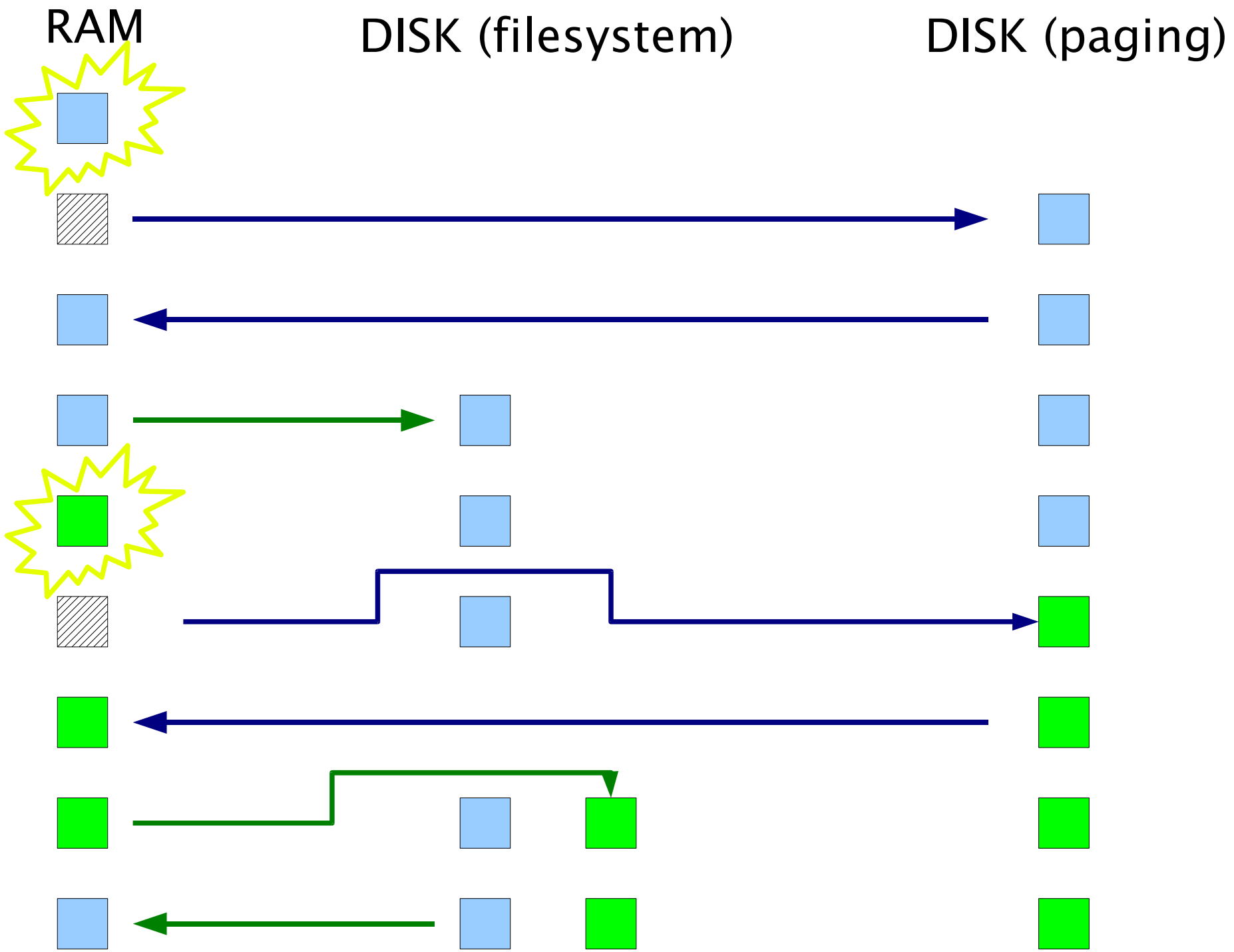
The Virtual Page Cache

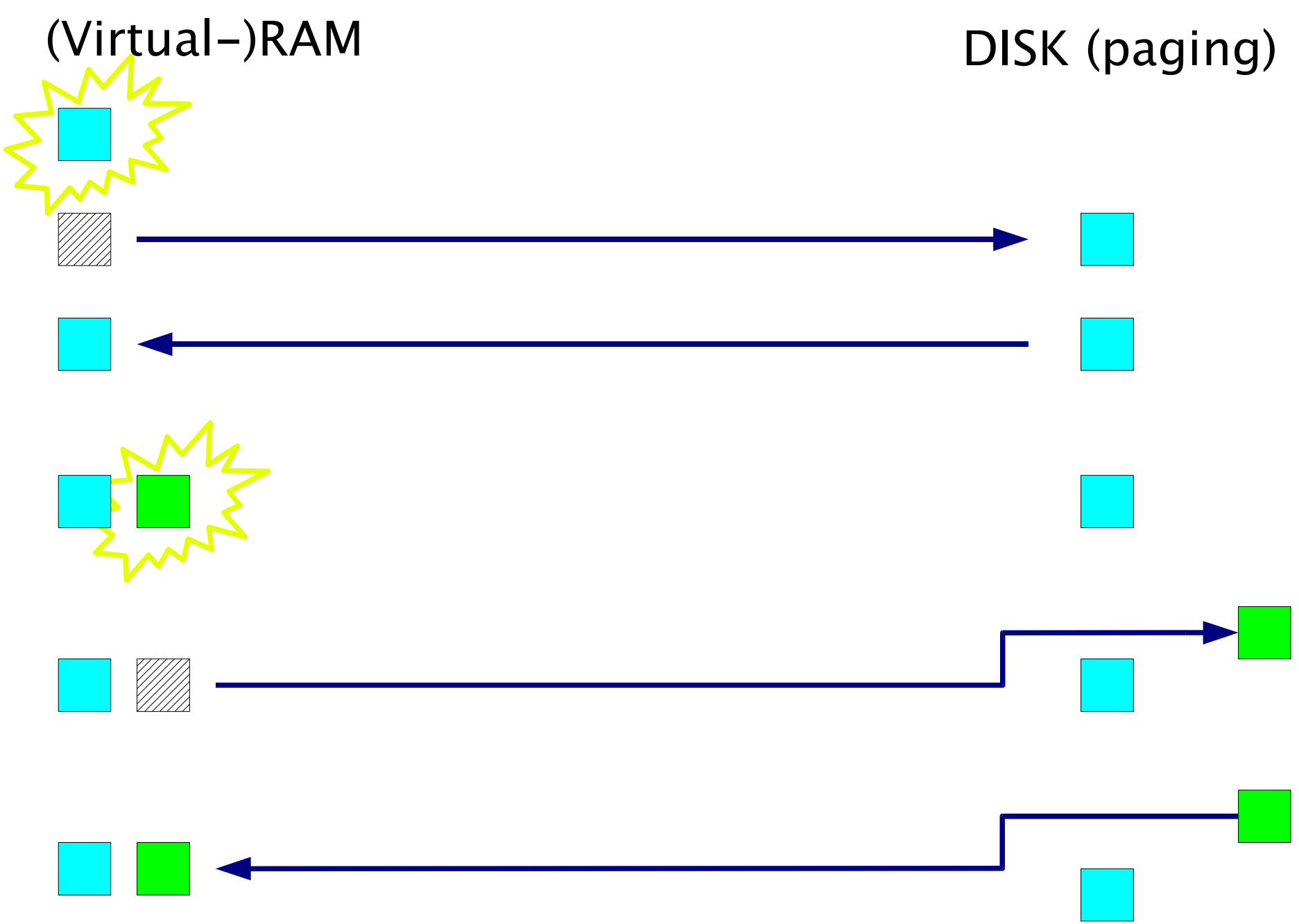
formerly known as "RAM"



Objects,
possibly
on storage

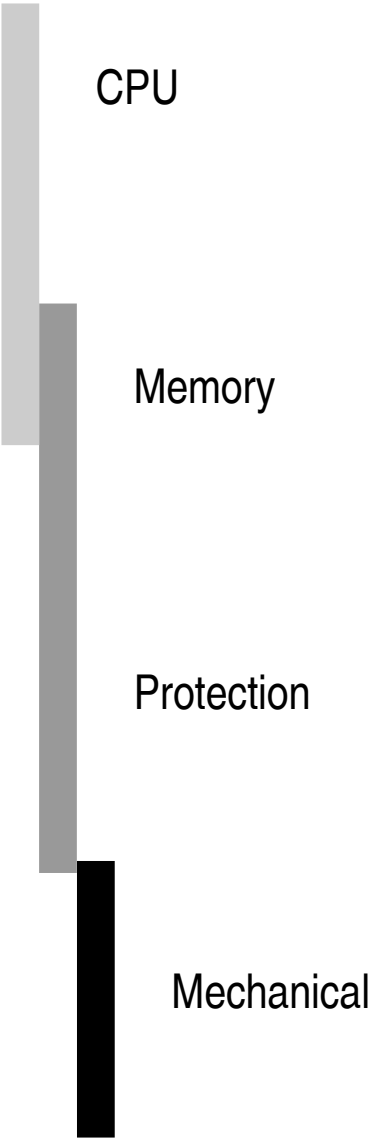

RAM
DISK (filesystem)


RAM
DISK (filesystem)
DISK (paging)


(Virtual-)RAM
DISK (paging)

Performance Pricelist

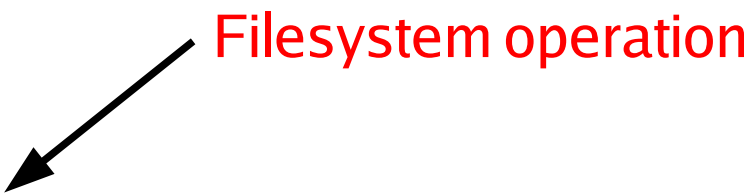
- char *p += 5;
- strlen(p);
- memcpy(p, q, l);
- Locking
- System Call
- Context Switch
- Disk Access
- File System Operation



0.00000001s

0.01s

Logging to a file is expensive:




```
FILE *flog;

flog = fopen("/var/log/mylog", "a");

[...]
```


```
fprintf(flog, "%s Something went wrong with %s\n",
    timestamp(), myrepresentation(object));
fflush(flog);
```

1 * 10msec + 1,000,000 * 1msec = 16 minutes

Logging to shared memory is cheap:

```
fd = open(...);
logp = mmap(..., size);
loge = logp + size;

[...]

logp[1] = LOG_ERROR;
logp[2] = sprintf(logp + 3,
    "Something went bad with %s",
     myrepresentation(obj));
logp[3 + logp[2]] = LOG_END;
logp[0] = LOG_ENTRY;
logp += 3 + logp[2];
```

Filesystem operations

Memory operations

2 * 10msec + 1,000,000 * 1µsec = 1.02 second

Multi-programming is hard.

Nobody asked for:

Multi-socket, Multi-core, SMP or NUMA

Hardware-guys ran out of steam

Software-guys pay the price

The trick to multiprogramming:

Keep everybody busy

Avoid conflict

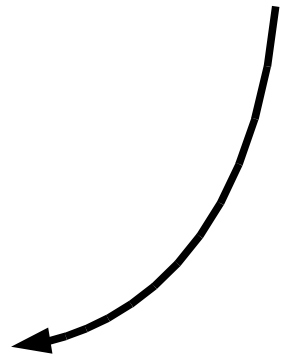
Avoid contention for resources

Malloc(3)

May need to synchronize between CPUs with locks

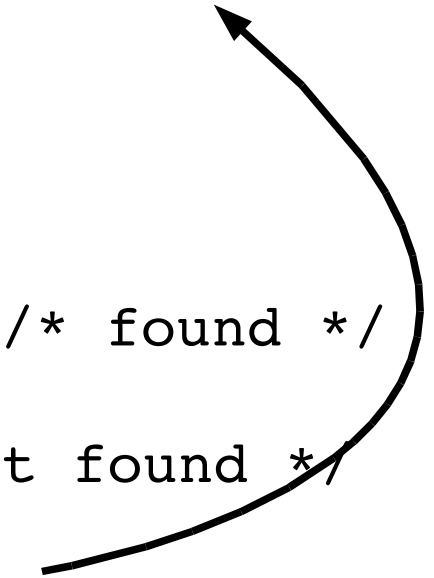
May need to get memory from kernel with syscall

```
LOCK(storage_lock);
TAILQ_FOREACH(...) {
        if (...)
                break;  /* found */
}
if (ptr == NULL) { /* not found */
        ptr = malloc(sizeof *wp->store);
        assert(wp->store != NULL);
        [...] // fill ptr->stuff
        TAILQ_INSERT_TAIL(...)
}
UNLOCK(storage_lock)
```



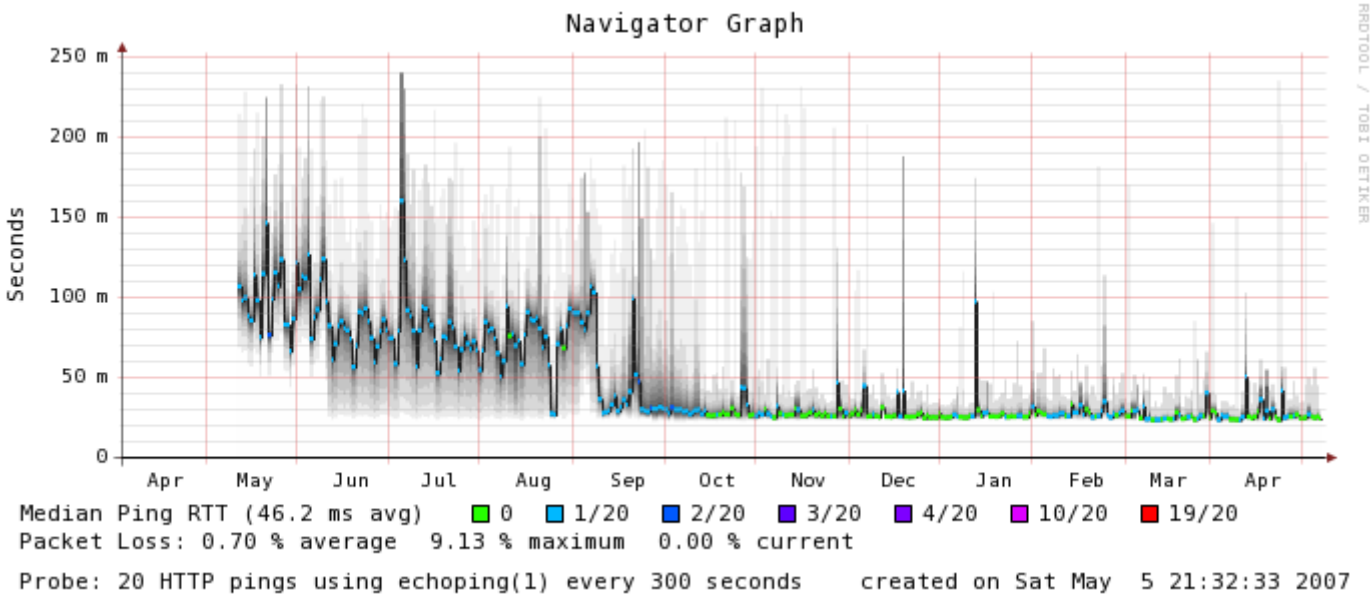
```
if (wp->store == NULL) {
        wp->store = malloc(sizeof *wp->store);
        assert(wp->store != NULL);
}
```

```
LOCK(storage_lock);
TAILQ_FOREACH(...) {
        if (...)
                break;  /* found */
}
if (ptr == NULL) { /* not found */
        ptr = wp->store;
        wp->store = NULL
        [...] // fill ptr->stuff
        TAILQ_INSERT_TAIL(...)
}
UNLOCK(storage_lock)
```



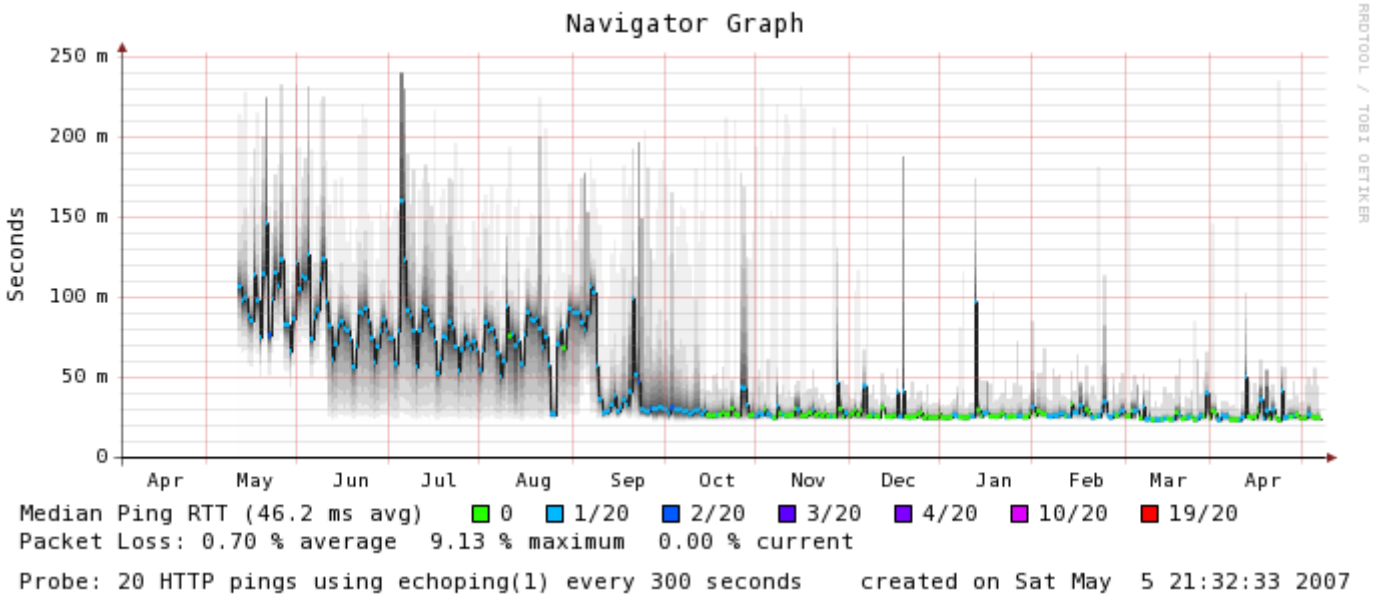
malloc call moved out of locked code section.

Lock is held shorter time!



Squid

Varnish



Squid
12 servers

Varnish
3 servers

9 servers @ 150W = 12 MWh/y

12 MWh * 500 kg = 6t CO_2 equiv.

Further savings:

Network equipment
Rack-space
Cooling
System Admin Hours

The website VG.no is one of Norways largest in terms of traffic.

Classical news site: rapidly changing contents in a slow CMS system.

12 Squid caches used as accelerators.

Unhappy with performance and stability.

VG NETT

Multimedia arm of Norvegian newspaper ”Verdens Gang”

RFC2616 cast list

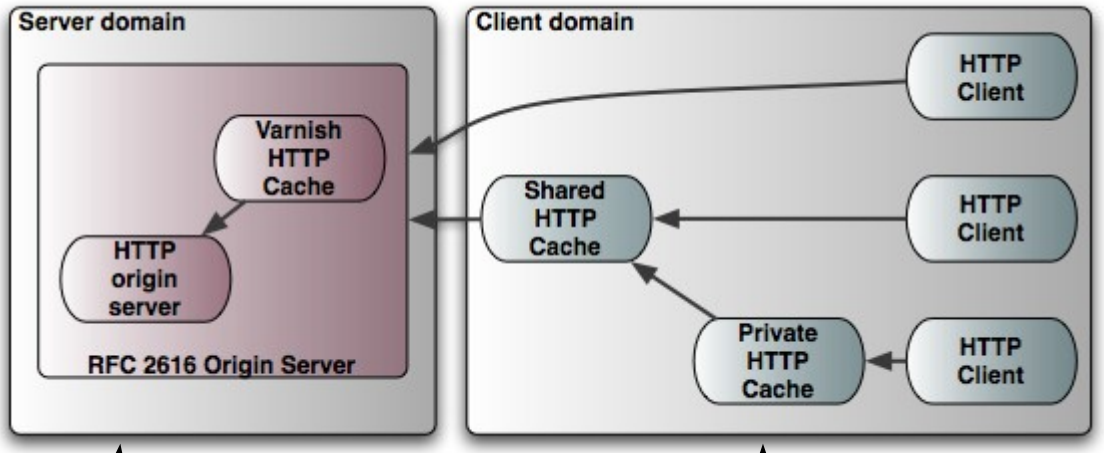
A HTTP Accelerator is not a HTTP cache.

Caching policy can be tailored to CMS system and site policies.

RFC2616 compliance as ”origin server”.

Web browsers

Can never cache per-user data



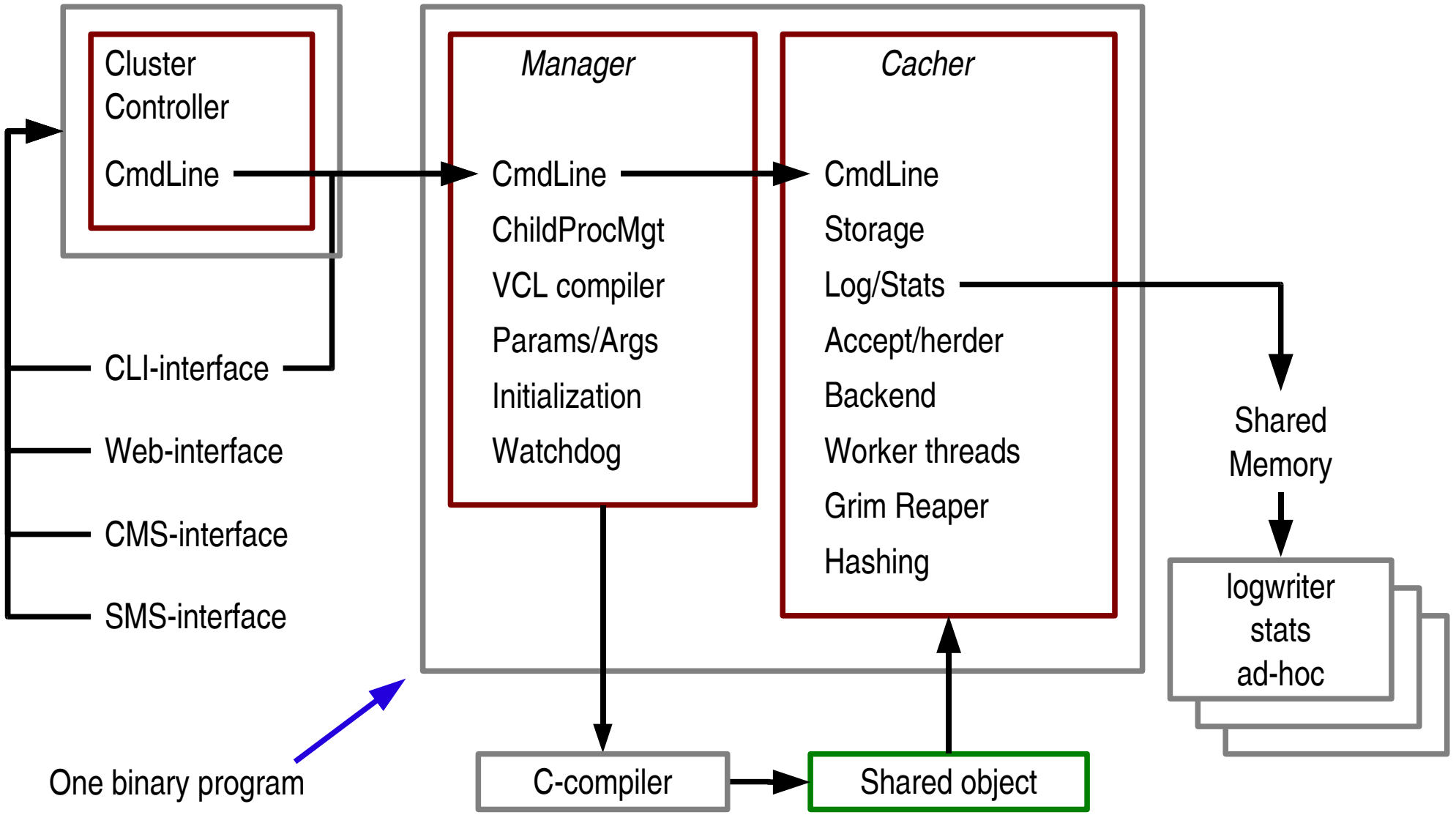
CMS system

Can sometimes cache per-user data

Starting from scratch:

- Varnish is a HTTP accelerator only.
- Better configuration.
- Better management.
- (Much) faster.
- Content Management Features

Varnish architecture



What is it about configuration files ?

```
$ cat /etc/foobar.conf
# foobard configuration file
# copied from example.conf
#         /svend 13-02-1147
# updated to new version
#         /knud 21-06-1157
# various changes
#         /valdemar 10-08-1157
# DON'T MESS WITH THIS!!!
```

```
HDXHSVVaCS=12

allocation_modulus=3.17⊕ℵ+∯f(21.4^μ•βe^-ij)

# overflow queue [default=7]
overflow_queue=7
```

```
process_backwards=if_acl_does_not

acl_set= { 1 2 3 4 6 7 21 }

invert_acls=odd

acl_reset = { 3 !8 }
```

VCL - Varnish Configuration Language

```
sub vcl_recv {
  if (req.request != "GET" && req.request != "HEAD") {
    pass;
  }
  if (req.http.Expect) {
    pipe;
  }
  if (req.http.Authenticate || req.http.Cookie) {
    pass;
  }
  lookup;
}
```

Shared memory → Cheap Real Time views

```
1304.86 /tmv11.js
 989.08 /sistenytt.html
 495.05 /include/global/art.js
 491.01 /css/hoved.css
 490.05 /gfk/ann/n.gif
 480.08 /gfk/ann/ng.gif
 468.12 /gfk/front/tipsvg.png
 352.66 /css/ufront.css
 317.75 /t.gif
 306.79 /gfk/plu2.gif
 298.84 /css/front.css
 292.84 /gfk/min2.gif
 280.94 /css/blog.css
 279.84 /
```

1000 obj/s @ 100Mbit/s

Sample	Min	Max	Median	Average	Stddev
Full	12,9	3001	16,2	71,1	338,5
90% fractile	12,9	26	15,9	16,3	1,7

(all times are in microseconds)

We...

...don't really know how fast.

...have seen 700 Mbit/s

...have heard 18k obj/s

...have not hit the limit yet.

var·nish (vär′nĭsh)

n.

1. **a.** A paint containing [...]

tr.v. **var·nished**, **var·nish·ing**, **var·nish·es**

1. To cover with varnish.

2. To give a smooth and glossy finish to.

3. To give a deceptively attractive appearance to; gloss over.

http://varnish-cache.org

Commercial support:

Linpro.no

phk@FreeBSD.org

Reference OS:
FreeBSD, Linux

Packages available:
Yes.

Portable to:
any reasonable POSIX