


System factors affecting throughput

Matt Crawford
Fermilab

Context of this work

- Of the many and varied causes of poor network performance, many turn out to be faults on the end systems.
- Some are configuration issues;
- Some are inherent design problems (or unfavorable trade-offs);
- Some look exactly like a bad network.

Linux Oddities

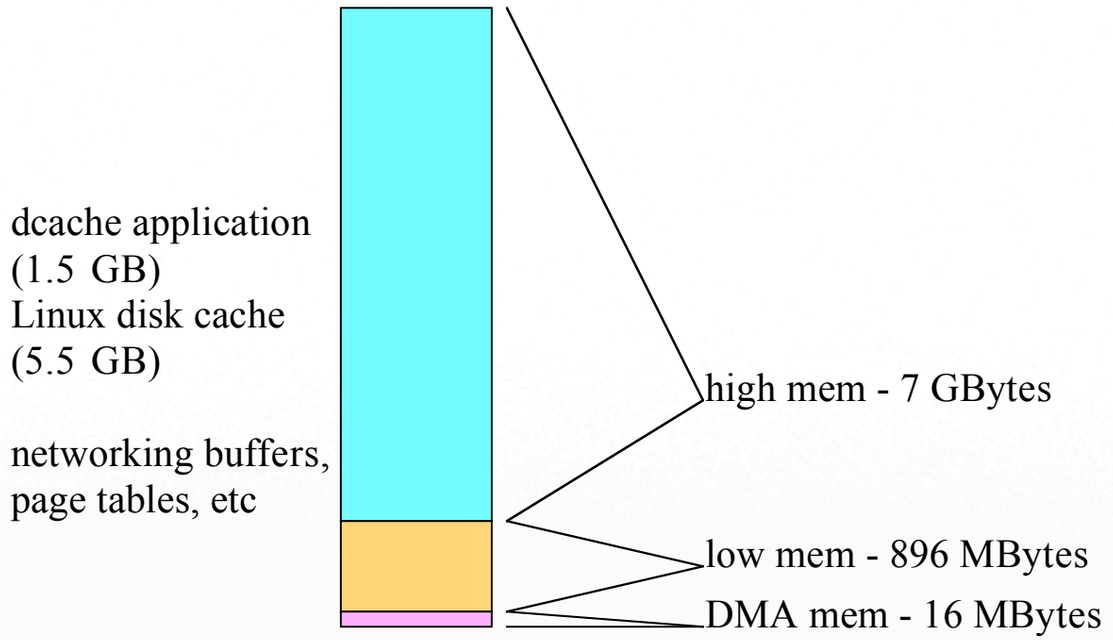
- Memory
- Interrupt Handling
- Scheduler Quirks

Memory Use

- Linux default TCP buffers suit local-area communications, not intercontinental.
- Applying the well-known BW×Delay cookbook to dCache storage systems led to frequent hangs & crashes.
- System & network wizards swing into action ...

- Dual 3.2GHz Xeon (32-bit) hosts, 8GB RAM, 2x1GE.

- Original complaint was “too many” concurrent transfers required to meet ingest goals from CERN.



- Cookbook (eg: PSC, LBNL) applied – 5MB window limits.

- Processes die; systems hang; logs show many memory allocation failures (!)

Unswappable & always mapped area “lowmem” limited to 896 MB.
Apx. 300MB fixed allocations, leaving 600MB for kmalloc.
Unused highmem→disk buffers.
Hypothesis: 4GB would serve better, due to smaller page tables.

- Single-stream TCP @ 1Gb/s FNAL–CERN (120 ms) requires ~ 15MB windows.
- But ...
  - Transfers are by GridFTP, 10-20 streams.
  - Many transfers are local or within US.
  - Found, for example, transfers to MIT consuming 10x more kernel buffer than the amount of data in flight!
- Rethink the window sizes.

```
rmem = 4096 87380 1048576
wmem = 4096 32768  131072
```

Linux is ruthless!

- Swapper tries (mostly in vain) to free memory for more network buffers; cycles would be better spent draining buffers.
- Linux grants memory reservation requests “optimistically,” since many large requests a for a process fork which will exec before consuming many new pages.
- When optimism proves unjustified, the “OOM Killer” brutally reclaims space.

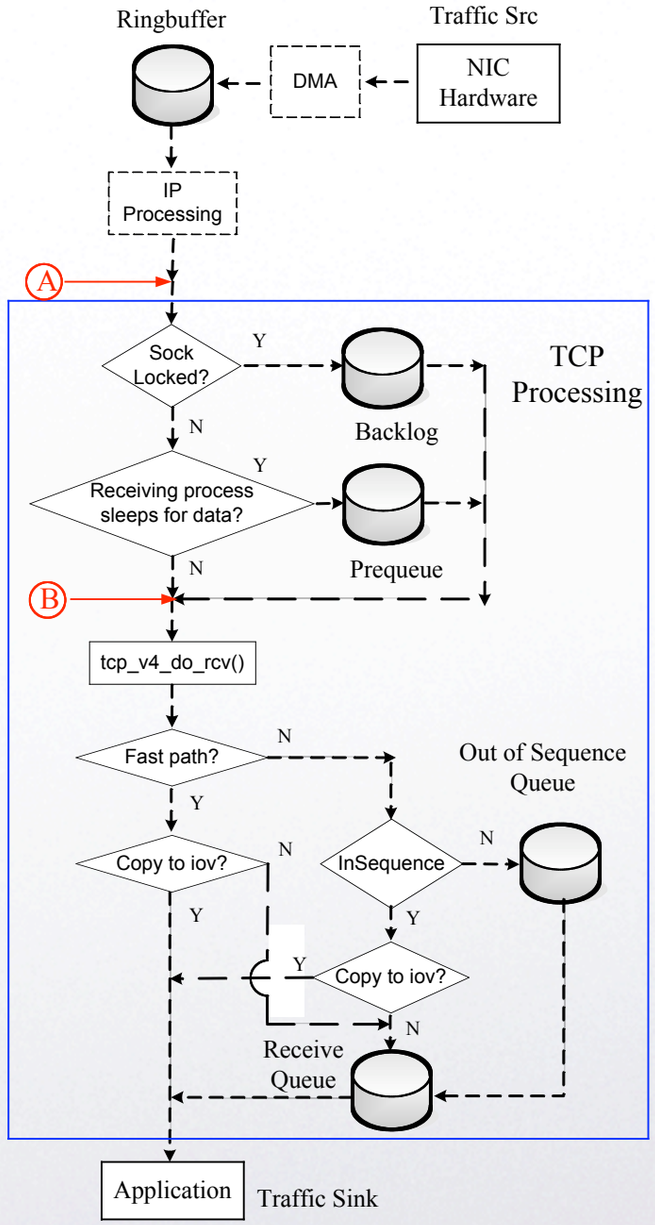
Interrupt Handling

- NIC driver "NAPI" mode (New API) handles multiple input packets per hardware interrupt; most work happens at software interrupt ("softirq") level.
- Host stack does limited work per softirq.
- *Transmit has higher priority than receive.*

⇒ Packets dropped inside host, due to lack of servicing the receive ring buffer.

Preemption & Locks

- If the process is suspended while receiving, the socket data structures are locked.
- Packets arriving during that time go on "backlog" queue; are *not* processed by TCP!
- Delay can be N × 100ms.



Fun with Scheduling

- Linux scheduler rewards interactive processes in two ways: priority boost and extra time slices.
- Interactivity measured by accumulation of interruptible sleep time > run time.
- A process receiving a *relatively* slow TCP stream fits that criterion all too well – can starve other processes.
- And, in fact, a slow stream will get better service than a fast stream.

Taming the pseudo-interactive receiver