CPE 442 : Performance of sobel program

Author(s)

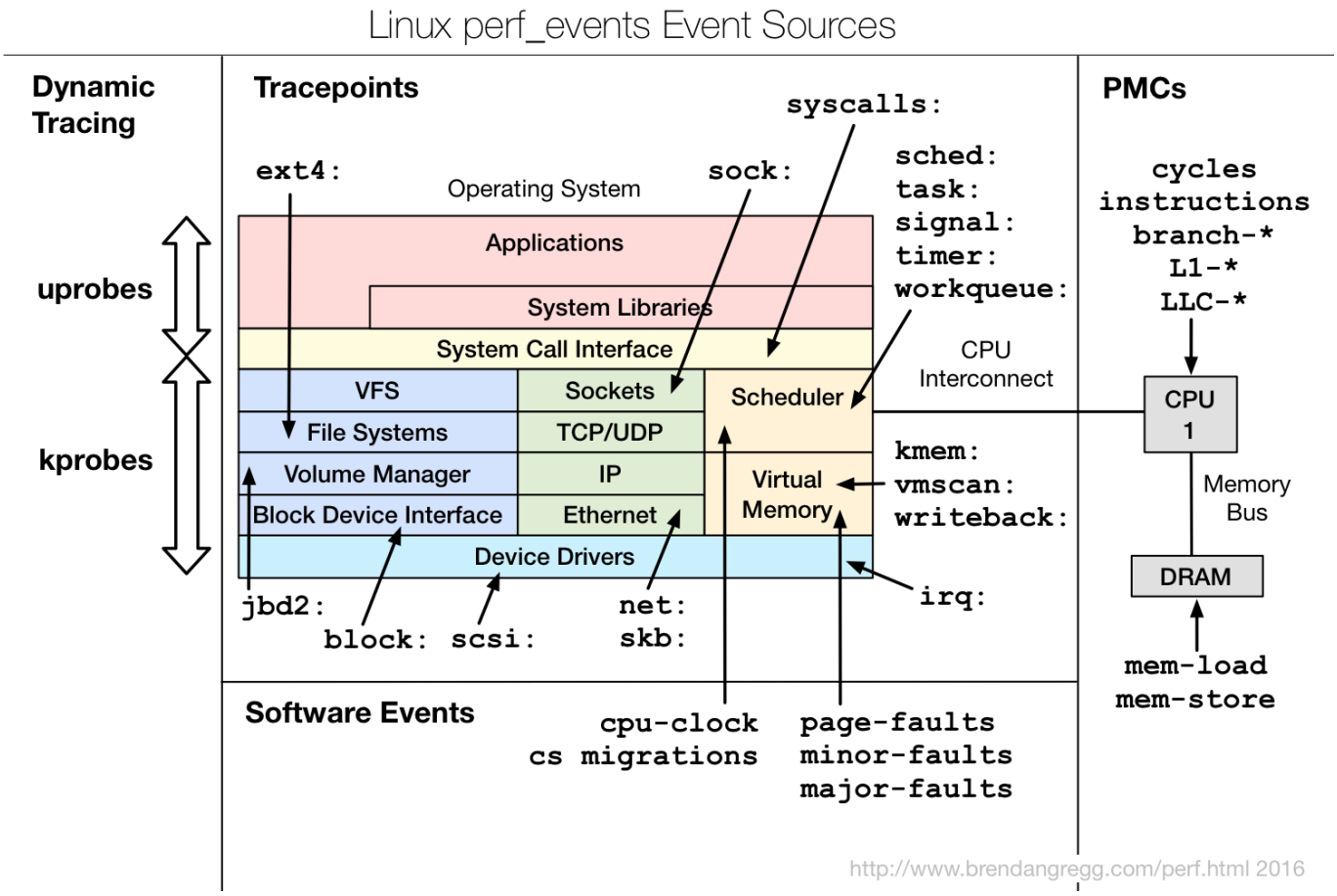
- Victor Delaplaine
- Tristan Chutka

Profiling C/C++ Applications

- This tutorials is to show how to profile an application. For this task we have chosen the Linux tool `perf`. This tool is called Performance Counters for Linux (PCL), or perf_events.

This event tool can help us answer some of the questions:

- Why is the kernel on-CPU so much? What code-paths?
- Which code-paths are causing CPU level 2 cache misses?
- Are the CPUs stalled on memory I/O?
- Which code-paths are allocating memory, and how much?
- What is triggering TCP retransmits?
- Is a certain kernel function being called, and how often?
- What reasons are threads leaving the CPU?

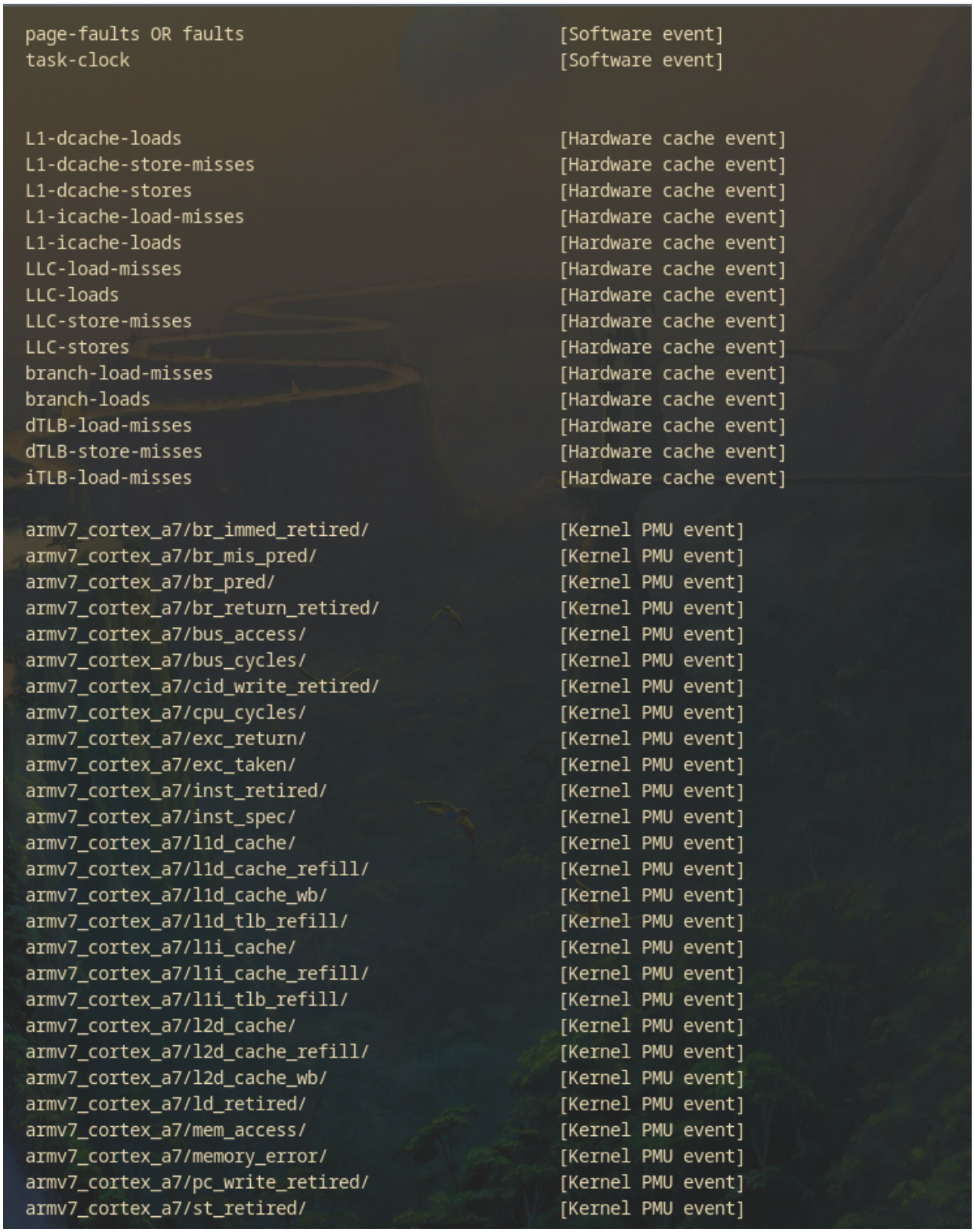


Installing libperf

```
sudo apt update sudo apt install linux-tools-$(uname -r)
```

Usage

- To list all currently known events for your architecture run: `perf list`



```
page-faults OR faults                              [Software event]
task-clock                                         [Software event]


L1-dcache-loads                                    [Hardware cache event]
L1-dcache-store-misses                             [Hardware cache event]
L1-dcache-stores                                   [Hardware cache event]
L1-icache-load-misses                              [Hardware cache event]
L1-icache-loads                                    [Hardware cache event]
LLC-load-misses                                    [Hardware cache event]
LLC-loads                                          [Hardware cache event]
LLC-store-misses                                   [Hardware cache event]
LLC-stores                                         [Hardware cache event]
branch-load-misses                                 [Hardware cache event]
branch-loads                                       [Hardware cache event]
dTLB-load-misses                                   [Hardware cache event]
dTLB-store-misses                                  [Hardware cache event]
iTLB-load-misses                                   [Hardware cache event]

armv7_cortex_a7/br_immed_retired/                  [Kernel PMU event]
armv7_cortex_a7/br_mis_pred/                       [Kernel PMU event]
armv7_cortex_a7/br_pred/                           [Kernel PMU event]
armv7_cortex_a7/br_return_retired/                 [Kernel PMU event]
armv7_cortex_a7/bus_access/                        [Kernel PMU event]
armv7_cortex_a7/bus_cycles/                        [Kernel PMU event]
armv7_cortex_a7/cid_write_retired/                 [Kernel PMU event]
armv7_cortex_a7/cpu_cycles/                        [Kernel PMU event]
armv7_cortex_a7/exc_return/                        [Kernel PMU event]
armv7_cortex_a7/exc_taken/                         [Kernel PMU event]
armv7_cortex_a7/inst_retired/                      [Kernel PMU event]
armv7_cortex_a7/inst_spec/                         [Kernel PMU event]
armv7_cortex_a7/l1d_cache/                         [Kernel PMU event]
armv7_cortex_a7/l1d_cache_refill/                  [Kernel PMU event]
armv7_cortex_a7/l1d_cache_wb/                      [Kernel PMU event]
armv7_cortex_a7/l1d_tlb_refill/                    [Kernel PMU event]
armv7_cortex_a7/l1i_cache/                         [Kernel PMU event]
armv7_cortex_a7/l1i_cache_refill/                  [Kernel PMU event]
armv7_cortex_a7/l1i_tlb_refill/                    [Kernel PMU event]
armv7_cortex_a7/l2d_cache/                         [Kernel PMU event]
armv7_cortex_a7/l2d_cache_refill/                  [Kernel PMU event]
armv7_cortex_a7/l2d_cache_wb/                      [Kernel PMU event]
armv7_cortex_a7/ld_retired/                        [Kernel PMU event]
armv7_cortex_a7/mem_access/                        [Kernel PMU event]
armv7_cortex_a7/memory_error/                      [Kernel PMU event]
armv7_cortex_a7/pc_write_retired/                  [Kernel PMU event]
armv7_cortex_a7/st_retired/                        [Kernel PMU event]
```

The above case shows me running the `perf list` command on a raspberry pi 4

The idea of perf events are to use a counter to see how many of those events occur. You can also see the architecure specific counters (armv_* events).

Events

- perf_events are ways to get statistics of the events, by using internal counters for each event.
- The `perf stat -e <event1, event2, .., eventn> <command>` gives you a print out of the hardware counters for each event.

```
pi@raspberrypi:~/CPE442/assns/assn5 $ perf stat ./sobel test1.mp4
./sobel: Killed

 Performance counter stats for './sobel test1.mp4':

     125717.277132      task-clock:u (msec)       #    2.497 CPUs utilized
                0      context-switches:u        #    0.000 K/sec
                0      cpu-migrations:u          #    0.000 K/sec
          256,352      page-faults:u             #    0.002 M/sec
  139,765,200,140      cycles:u                  #    1.112 GHz
  110,890,450,594      instructions:u            #    0.79  insn per cycle
    5,531,987,789      branches:u                #   44.003 M/sec
      122,044,842      branch-misses:u           #    2.21% of all branches

      50.351623983 seconds time elapsed
```

The above shows you some basic events it give you by default

Cache Events

- For our case we may want to find the number of:
  - L1 data cache misses so:

```
pi@raspberrypi:~/CPE442/assns/assn5 $ perf stat -e L1-dcache-load-misses  ./sobel test1.mp4
./sobel: Killed

 Performance counter stats for './sobel test1.mp4':

       102,658,841      L1-dcache-load-misses:u

      49.177100884 seconds time elapsed
```

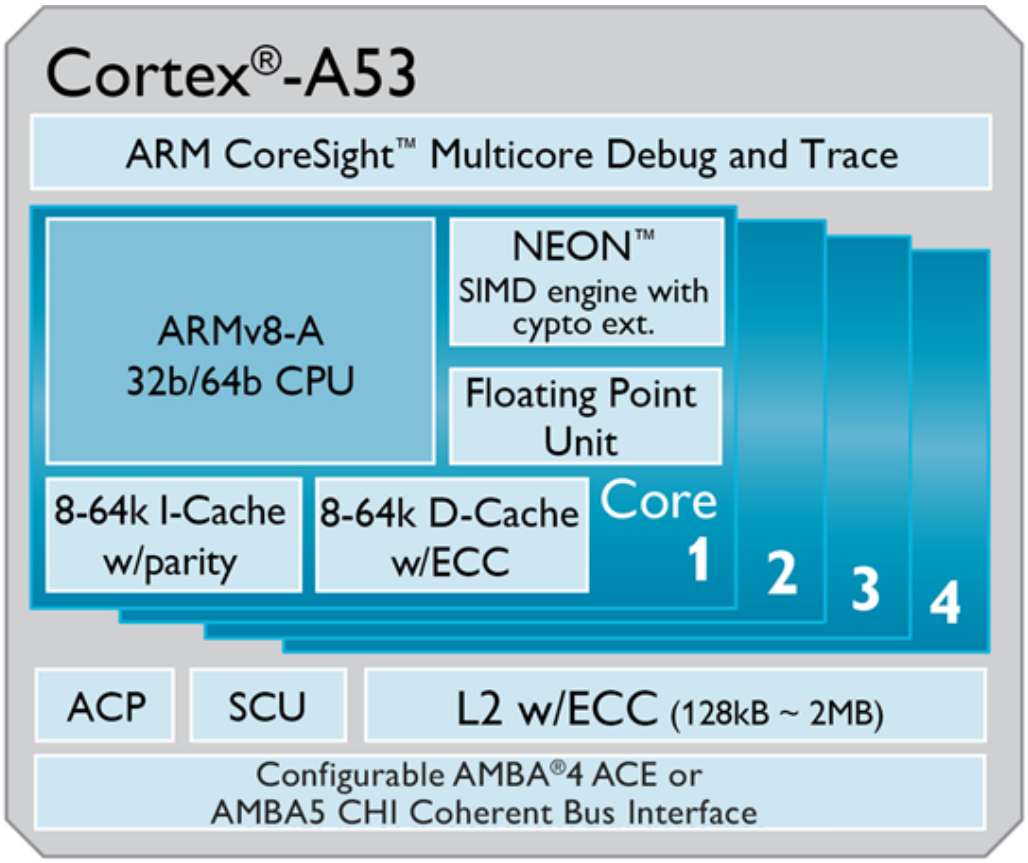
  - L2 data cache misses, but is the L2 Cache the Last Level Cache (LLC) (Look below)

```
pi@raspberrypi:~/CPE442/assns/assn5 $ perf stat -e LLC-load-misses
./sobel: Killed

 Performance counter stats for './sobel test1.mp4':

        83,358,349      LLC-load-misses:u

      60.087128730 seconds time elapsed
```



- This pictures of the `Cortex-A53` architecture helps indeicate that L2 is the LLC

Architecture specific events

- Say we want to count the number of cpu cycles it takes to run your program:

```
pi@raspberrypi:~/CPE442/assns/assn5 $ perf stat -e armv7_cortex_a7/cpu_cycles/ ./sobel test1.mp4
./sobel: Killed

 Performance counter stats for './sobel test1.mp4':

   140,698,887,289      armv7_cortex_a7/cpu_cycles/:u

      43.865236696 seconds time elapsed
```

- I found the architecture specific event in the `perf list` table
  - Specific to the armv7_cortex_a7

Reference(s) for `perf`

Linux Perf Examples