

# Sam: a tiny, concurrent sampling profiler for Clozure Common Lisp

Max Rottenkolber <max@mr.gy>

Wednesday, 27 July 2016

I have recently been in need of a profiler for my favorite Common Lisp implementation, *Clozure Common Lisp* (<http://ccl.clozure.com/>), but sadly its *mechanism for generating a Lisp image for use with oprofile* (<http://ccl.clozure.com/docs/ccl.html#using-the-linux-oprofile-system-level-profiler>) no longer works with recent versions of Linux/*oprofile*. *rme* (<http://www.thoughtstuff.com/>) explained it to me in #ccl on Freenode:

*“Recent kernels and perf versions are not fooled by our trick of making a heap image look sort of like an ELF file. Also, they only consider mapped memory from ELF files (and not just some random file, like a heap image) as a place where functions can reside.”*

That was disappointing, I had waited for the new Slackware release to get a compatible version of *libelf*, after all. I also looked into the deterministic profiler for CCL by Hans Huebner that lives in `contrib/`, but that one bit-rotted, sadly.



Still from the 2009 movie “Moon”. The protagonist’s name is Sam as well.

Still, my search was not in vain. *rme* sent me a very helpful *paste* (<http://paste.lisp.org/display/319875>), whose contents he described as “a crude userland-based sampling profiler”. Here is how it works: it peeks into the call stack of a running process repeatedly to find out which function is being run, and counts how often it finds the process busy with a given function. With the collected information it can then report an estimate of where time is spent.

After having given some love to this tiny profiler's report output, I figured why not share it for others to use, and rme kindly agreed. Meet *Sam* (<https://github.com/eugeneia/sam>), it supports profiling forms as well as running processes, and is reasonably well *documented* (<http://mr.gy/software/sam/api.html>). You can use it to profile arbitrary forms like this:

```
CL-USER> (sam:profile ()
           (loop for i from 1 to 1e8 sum i))
39% CCL::>-2 <no source>
19% CCL::FIXNUM-SFLOAT-COMPARE <no source>
15% #<Anonymous Function #x30200B10843F> <no source>
12% SAM:CALL-WITH-SAMPLING (defun call-with-sampling (fn ...
11% CCL::INTEGER-DECODE-SHORT-FLOAT <no source>
 5% CCL::FIXNUM-DECODE-SHORT-FLOAT <no source>
```

...or profile a different process:

```
CL-USER> (sam:profile-process (fourth (all-processes)) 1)
100% CCL::FD-READ <no source>
```

May the search engines index this, and spare others from repeating my search (but not from reading about it.) I hope Sam can be of use to you, and let me know if you find any bugs!