Linux Perf Tools

Overview and Current Developments

Arnaldo Carvalho de Melo, Jiri Olsa

Red Hat Inc.

May 24, 2013

Overview

- Multiple events view
- Annotate GTK UI
- New 'perf mem' tool
- Per socket/core aggregation
- Diff enhancements
- Group leader sampling
- DWARF unwind
- Default precise
- Toggling events
- Kbuild integration
- Tests
- perf probe + scripting example

Multiple events without grouping

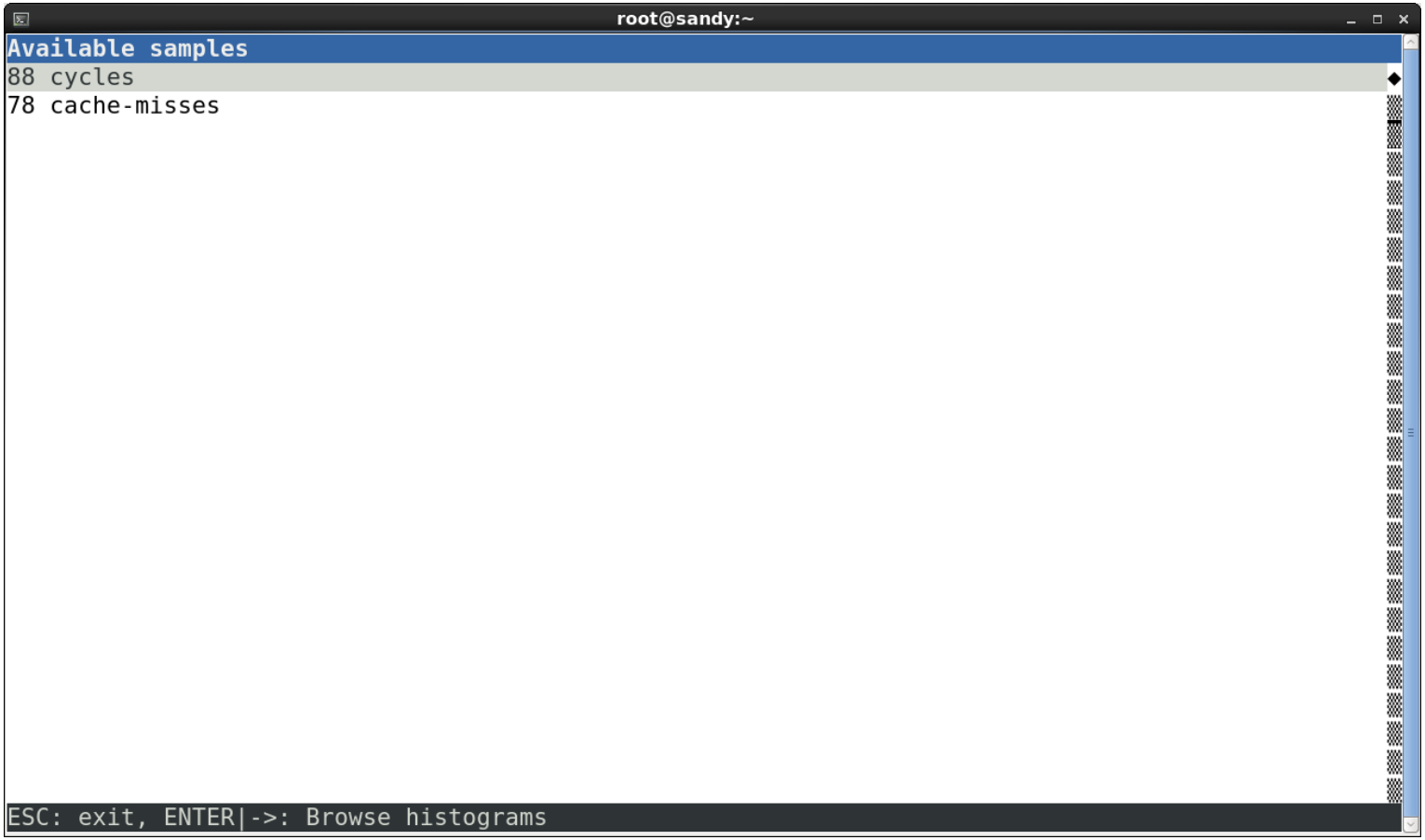
```
# perf record -e cycles,cache-misses -a usleep 1
[ perf record: Woken up 1 times to write data ]
[ perf record: Captured and wrote 0.616 MB perf.data (~26891 samples
# perf evlist
cycles
cache-misses
#
```

Multiple events grouping

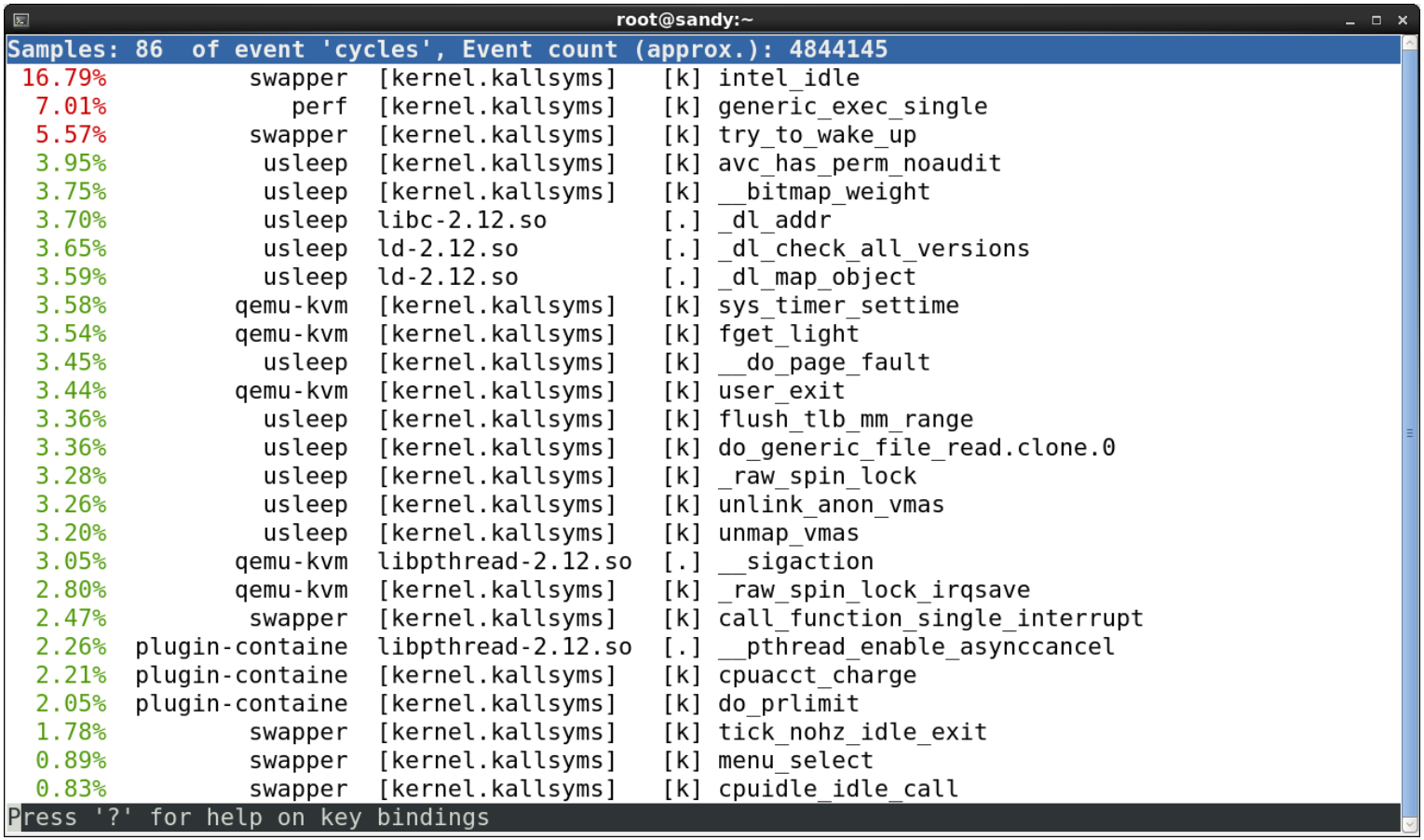
```
# perf record -e '{cycles,cache-misses}' -a usleep 1
[ perf record: Woken up 1 times to write data ]
[ perf record: Captured and wrote 0.621 MB perf.data (~27151 samples
# perf evlist
cycles
cache-misses
# perf evlist --group
{cycles,cache-misses}
#
```

perf report - no grouping

```
# perf report
```

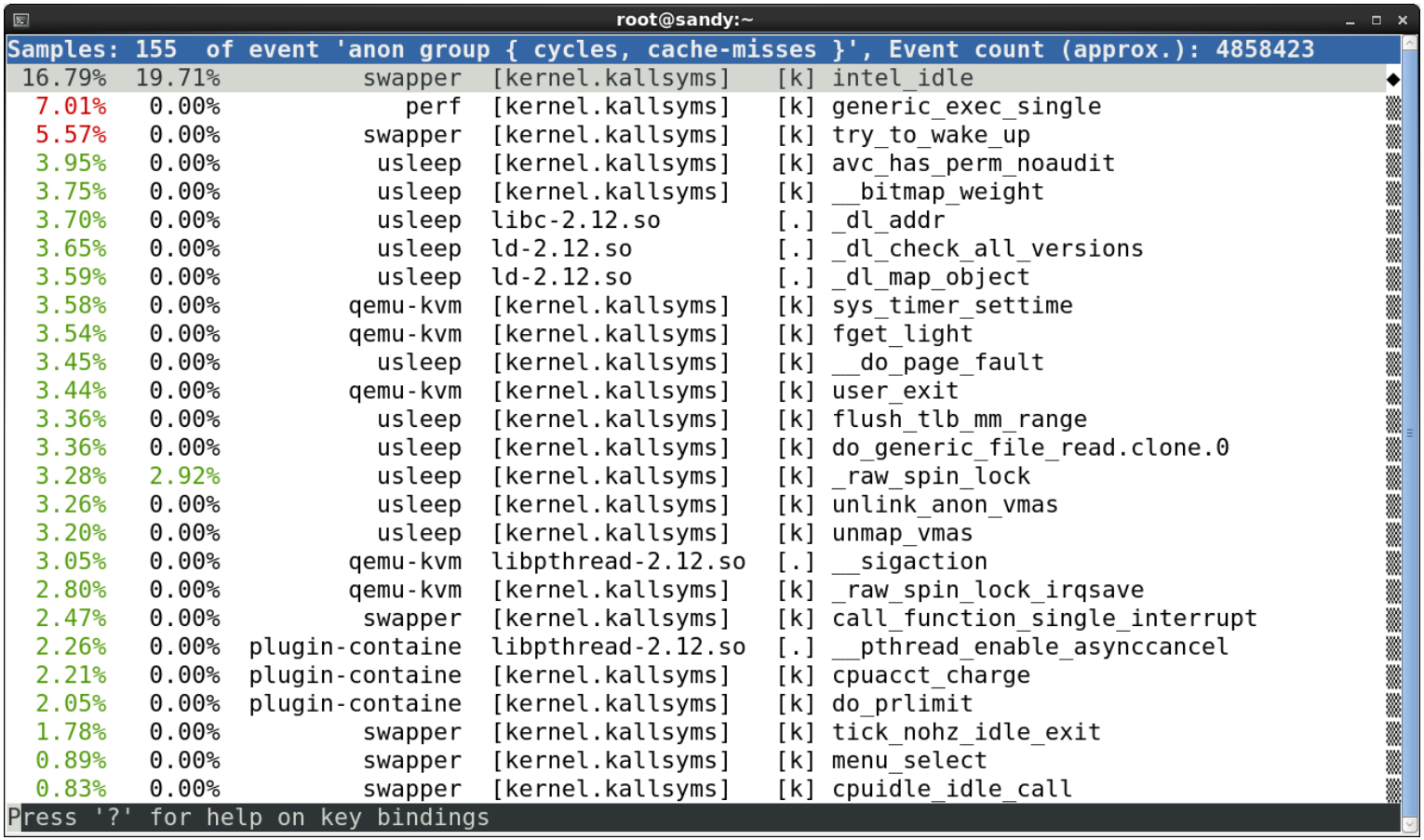


perf report - single event

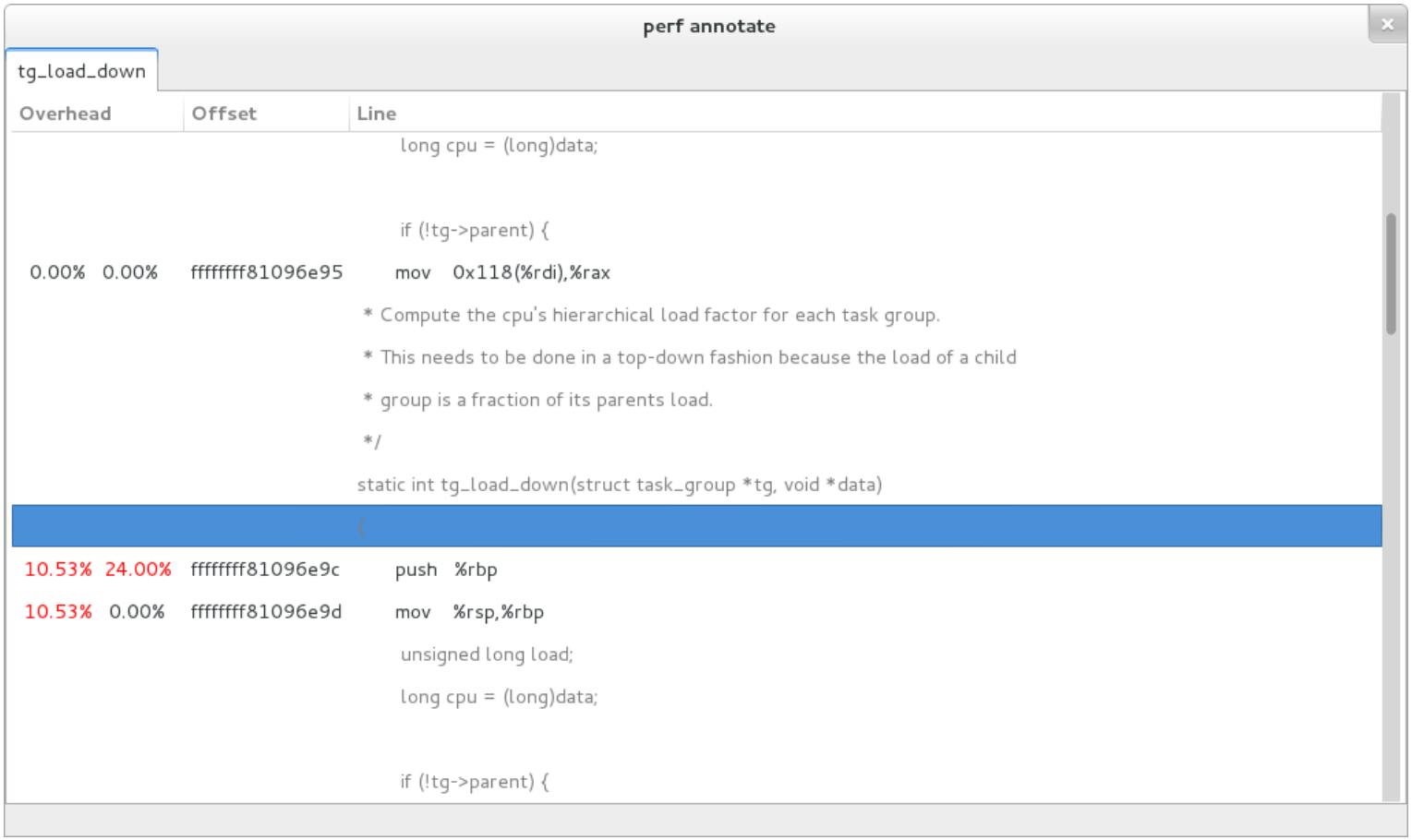


perf report - multiple events

```
# perf report --group
```



perf annotate gtk



Per socket/core aggregation

- System wide
- Per socket/core
- Helps find imbalances
- Can be combined with interval printing

```
# perf stat -I 1000 -a --per-socket -e cycles sleep 200
#           time socket cpus             counts events
     1.000097680 S0        4          5,788,785 cycles
     2.000379943 S0        4         27,361,546 cycles
     2.001167808 S0        4            818,275 cycles
^C
```

perf mem

- Memory access profiling
- PEBS/IBS
- Memory level of access: L1, L2, L3, RAM
- Access latency
- Resolves symbols to global
- More work needed to resolve to locals using DWARF

perf mem

```
# perf mem -t loads record -a usleep 10
[ perf record: Woken up 1 times to write data ]
[ perf record: Captured and wrote 0.427 MB perf.data (~18636 samples

# perf evlist
cpu/mem-stores/pp
```

perf mem report

```
# perf mem report
```

root@zoo:~

```
Samples: 2K of event 'cpu/mem-loads/pp', Event count (approx.): 34909
  2.05%  7            L1 hit                  [k] _raw_spin_lock_bh                    [kernel.kallsyms]
  1.52%  531          LFB hit                 [k] intel_idle                           [kernel.kallsyms]
  1.10%  385          LFB hit                 [.] g_main_context_check                 libglib-2.0.so.0.34
  0.81%  283          Local RAM hit           [k] ext4_nonda_switch                    [kernel.kallsyms]
  0.76%  266          L1 hit                  [.] g_main_context_find_source_by_id     libglib-2.0.so.0.34
  0.75%  262          LFB hit                 [k] unix_poll                            [kernel.kallsyms]
  0.71%  247          L3 miss                 [k] fget_light                           [kernel.kallsyms]
  0.69%  241          L2 hit                  [k] cpuidle_wrap_enter                   [kernel.kallsyms]
  0.68%  236          LFB hit                 [k] intel_idle                           [kernel.kallsyms]
  0.67%  233          L1 hit                  [.] 0x000000000028e839                   libmozjs.so
  0.66%  231          Local RAM hit           [k] kmem_cache_alloc_node                [kernel.kallsyms]
  0.65%  226          L1 hit                  [k] mutex_lock                           [kernel.kallsyms]
  0.61%  214          LFB hit                 [.] g_str_hash                           libglib-2.0.so.0.34
  0.60%  210          LFB hit                 [k] tg_load_down                         [kernel.kallsyms]
  0.55%  191          L1 hit                  [k] ext4_da_write_begin                  [kernel.kallsyms]
  0.55%  191          LFB hit                 [.] 0x000000000104d218                   libxul.so
  0.52%  183          LFB hit                 [k] update_sd_lb_stats                   [kernel.kallsyms]
  0.52%  181          Local RAM hit           [k] start_this_handle                    [kernel.kallsyms]
  0.49%  172          Local RAM hit           [k] place_entity                         [kernel.kallsyms]
  0.46%  159          L3 hit                  [k] intel_idle                           [kernel.kallsyms]
  0.45%  158          L1 hit                  [k] update_curr                          [kernel.kallsyms]
  0.44%  9            L1 hit                  [k] delay_tsc                            [kernel.kallsyms]
  0.43%  151          LFB hit                 [.] 0x0000000000019175                   systemd-journald
  0.41%  144          L1 hit                  [k] __list_del_entry                     [kernel.kallsyms]
  0.41%  143          L1 hit                  [k] do_vfs_ioctl                         [kernel.kallsyms]
  0.40%  141          L1 hit                  [k] tg_load_down                         [kernel.kallsyms]
  0.40%  141          Local RAM hit           [.] 0x000000000124e09b                   libxul.so
  0.40%  140          Local RAM hit           [k] security_cred_free                   [kernel.kallsyms]
  0.40%  139          L3 miss                 [k] tg_load_down                         [kernel.kallsyms]
  0.40%  139          Local RAM hit           [k] mark_page_accessed                   [kernel.kallsyms]
  0.38%  134          L1 hit                  [k] cpuidle_wrap_enter                   [kernel.kallsyms]
  0.38%  133          LFB hit                 [k] memcpy                               [kernel.kallsyms]
  0.38%  132          L1 hit                  [k] add_interrupt_randomness             [kernel.kallsyms]
Press '?' for help on key bindings
```

perf report mem-mode

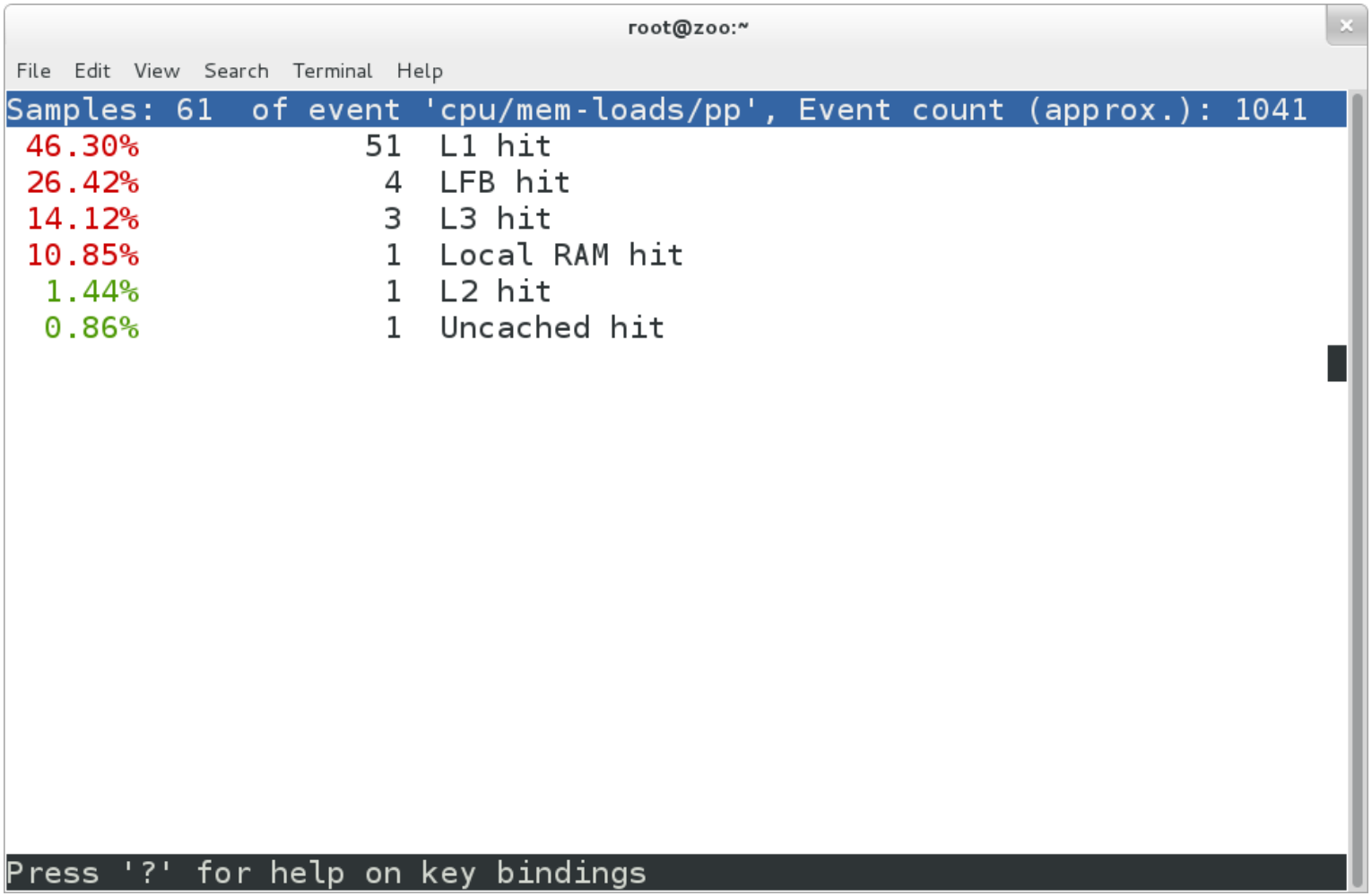
```
  --sort=mem,sym,dso,symbol_daddr,
         dso_daddr,tlb,locked
# perf report --mem-mode -s mem,symbol,symbol_daddr
```

root@zoo:~

```
Samples: 2K of event 'cpu/mem-loads/pp', Event count (approx.): 34909
  2.16%  L1 hit                  [k] _raw_spin_lock_bh                  [k] 0xffff880260a89ff8
  1.52%  LFB hit                 [k] intel_idle                         [k] 0xffff88026395de20
  1.10%  LFB hit                 [.] g_main_context_check               [.] 0x00007f3a9c039ad0
  0.81%  Local RAM hit           [k] ext4_nonda_switch                  [k] 0xffff88025ebf80e0
  0.76%  L1 hit                  [.] g_main_context_find_source_by_id   [.] 0x00007fff983dca28
  0.75%  LFB hit                 [k] unix_poll                          [k] 0xffff88026316bb8c
  0.71%  L3 miss                 [k] fget_light                         [k] 0xffff88024931b245
  0.69%  L2 hit                  [k] cpuidle_wrap_enter                 [k] 0xffff88026395fe90
  0.68%  LFB hit                 [k] intel_idle                         [k] 0xffffffff81c01e18
  0.67%  L1 hit                  [.] 0x000000000028e839                 [.] 0x00007f189f7188d6
  0.66%  Local RAM hit           [k] kmem_cache_alloc_node              [k] 0xffff88026037f400
  0.65%  L1 hit                  [k] mutex_lock                         [k] 0xffff88021d1ddac8
  0.61%  LFB hit                 [.] g_str_hash                         [.] 0x000000375293702f
  0.60%  LFB hit                 [k] tg_load_down                       [k] 0xffff8802492d9848
  0.55%  L1 hit                  [k] ext4_da_write_begin                [k] 0xffffea00071ae300
  0.55%  LFB hit                 [.] 0x000000000104d218                 [.] 0x00007fff8cd76168
  0.52%  LFB hit                 [k] update_sd_lb_stats                 [k] 0xffff88026f214050
  0.52%  Local RAM hit           [k] start_this_handle                  [k] 0xffff88025d1f3a0c
  0.49%  Local RAM hit           [k] place_entity                       [k] sysctl_sched_latency+0x0
  0.46%  L3 hit                  [k] intel_idle                         [k] lapic_timer_reliable_states+0
  0.45%  L1 hit                  [k] update_curr                        [k] 0xffff88021d1dd8d8
  0.44%  L1 hit                  [k] delay_tsc                          [k] 0xffff88026f20b034
  0.43%  LFB hit                 [.] 0x0000000000019175                 [.] 0x00007f90e88fb070
  0.41%  L1 hit                  [k] __list_del_entry                   [k] 0xffff880260e446c8
  0.41%  L1 hit                  [k] do_vfs_ioctl                       [k] 0xffffffffa00e0e60
  0.40%  L1 hit                  [k] tg_load_down                       [k] 0xffff880260820600
  0.40%  Local RAM hit           [.] 0x000000000124e09b                 [.] 0x00007f18b42e6680
  0.40%  Local RAM hit           [k] security_cred_free                 [k] selinux_ops+0x248
  0.40%  L3 miss                 [k] tg_load_down                       [k] 0xffff880260848840
  0.40%  Local RAM hit           [k] mark_page_accessed                 [k] 0xffffea000979cb40
  0.38%  L1 hit                  [k] cpuidle_wrap_enter                 [k] 0xffff88026f244010
  0.38%  LFB hit                 [k] memcpy                             [k] 0xffff88026310603f
  0.38%  L1 hit                  [k] add_interrupt_randomness           [k] 0xffff88026f290a00
Press '?' for help on key bindings
```

perf report mem-mode

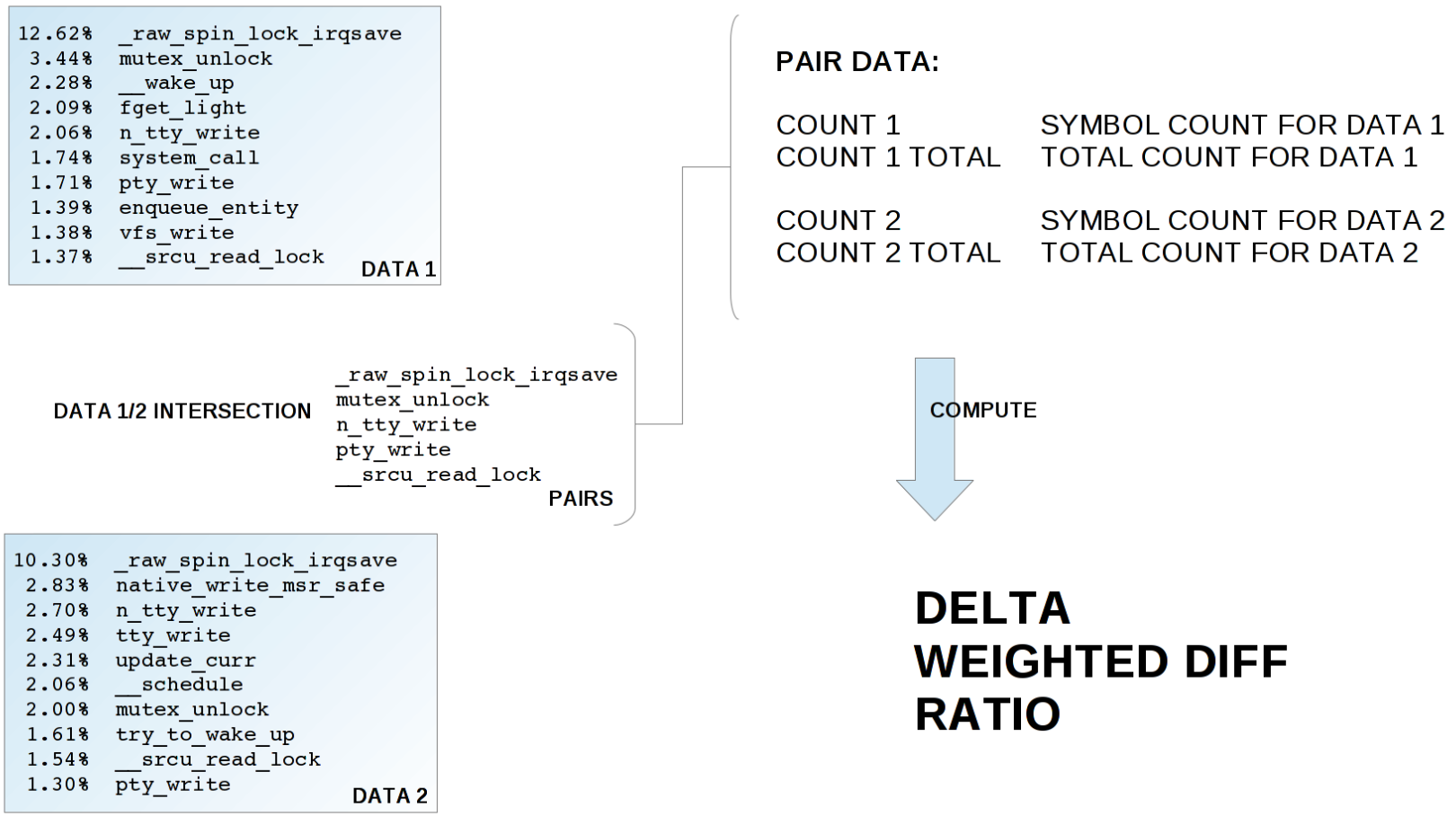
```
# perf mem report -s mem
```



Diff enhancements

- compare methods: delta, weighted diff, ratio
  (already in)
- Paul E.McKenney - Differential Profiling
- multiple data files
  (soon to be merged)

Diff enhancements - basics



Diff enhancements - delta

%1 = (COUNT1 * 100) / COUNT1 TOTAL
%2 = (COUNT2 * 100) / COUNT2 TOTAL

DELTA = %2 - %1

```
$ perf  diff -c delta
# Event 'cycles'
#
# Baseline    Delta      Shared Object                                 Symbol
# ........  .......  ..................  ....................................
#
    12.62%   -2.32%  [kernel.kallsyms]  [k] _raw_spin_lock_irqsave
     3.44%   -1.44%  [kernel.kallsyms]  [k] mutex_unlock
     2.06%   +0.64%  [kernel.kallsyms]  [k] n_tty_write
     1.71%   -0.42%  [kernel.kallsyms]  [k] pty_write
     1.37%   +0.17%  [kernel.kallsyms]  [k] __srcu_read_lock
...
```

Diff enhancements - weighted diff

WEIGHT1 = USER DEFINED
WEIGHT2 = USER DEFINED

WEIGHTED DIFF = COUNT2 * WEIGHT1 – COUNT1 * WEIGHT2

```
$ perf  diff -c wdiff:1,2
# Event 'cycles'
#
# Baseline   Weighted diff      Shared Object                                 Symbol
# ........  ..............  ..................  ....................................
#
    12.62%       100376692  [kernel.kallsyms]  [k] _raw_spin_lock_irqsave
     3.44%        17128216  [kernel.kallsyms]  [k] mutex_unlock
     2.06%        29267199  [kernel.kallsyms]  [k] n_tty_write
     1.71%        12346582  [kernel.kallsyms]  [k] pty_write
     1.37%        16196601  [kernel.kallsyms]  [k] __srcu_read_lock
...
```

Diff enhancements - ratio

RATIO = COUNT2 / COUNT1

```
$ perf  diff -c ratio
# Event 'cycles'
#
# Baseline           Ratio      Shared Object                                  Symbol
# ........  ..............  ..................  ....................................
#
    12.62%        2.168020  [kernel.kallsyms]  [k] _raw_spin_lock_irqsave
     3.44%        1.542882  [kernel.kallsyms]  [k] mutex_unlock
     2.06%        3.477702  [kernel.kallsyms]  [k] n_tty_write
     1.71%        2.010982  [kernel.kallsyms]  [k] pty_write
     1.37%        2.986062  [kernel.kallsyms]  [k] __srcu_read_lock
...
```

Diff enhancements - multiple data files - example

```
$ perf diff -b ./perf.data.[123456]
# Event 'cycles'
#
# Data files:
#  [0] ./perf.data.1 (Baseline)
#  [1] ./perf.data.2
#  [2] ./perf.data.3
#  [3] ./perf.data.4
#  [4] ./perf.data.5
#  [5] ./perf.data.6
#
# Baseline/0  Delta/1  Delta/2  Delta/3  Delta/4  Delta/5      Shared Object                              Symbol
# ..........  .......  .......  .......  .......  .......  ..................  ..................................
#
     36.44%   +0.27%   +7.81%   +1.18%   +0.72%   +0.74%  libc-2.15.so        [.] _IO_file_xsputn@@GLIBC_2.2.5
     32.70%   -2.74%  -12.76%   -0.90%   -2.16%   -1.11%  yes                 [.] 0x000000000000140b
     15.01%   +1.75%   +0.50%   +1.03%   +1.80%   +0.13%  libc-2.15.so        [.] __strlen_sse2
     14.88%   +0.45%   +4.45%   -1.38%   -0.64%   +0.11%  libc-2.15.so        [.] fputs_unlocked
      0.25%   +0.31%   -0.08%   +0.04%   +0.33%   +0.03%  yes                 [.] fputs_unlocked@plt
      0.11%   -0.05%   -0.02%   -0.05%   -0.06%           [kernel.kallsyms]   [k] __srcu_read_lock
      0.06%   -0.03%   -0.05%   -0.05%   -0.04%   -0.02%  [kernel.kallsyms]   [k] fget_light
      0.05%   -0.03%            -0.02%   -0.02%           [kernel.kallsyms]   [k] native_write_msr_safe
      0.05%   -0.01%   -0.01%   +0.01%   +0.06%           [kernel.kallsyms]   [k] system_call
      0.05%            +0.02%            -0.03%           [kernel.kallsyms]   [k] __audit_syscall_exit
      0.05%            -0.04%   -0.03%                    [kernel.kallsyms]   [k] sysret_check
      0.03%   -0.02%            -0.02%            -0.02%  libc-2.15.so        [.] _IO_file_overflow@@GLIBC_2.2.5
      0.03%                     -0.02%   -0.02%   -0.02%  [kernel.kallsyms]   [k] security_file_permission
      0.03%            -0.02%                             [kernel.kallsyms]   [k] fsnotify
      0.03%                                               [kernel.kallsyms]   [k] __audit_syscall_entry
...
```

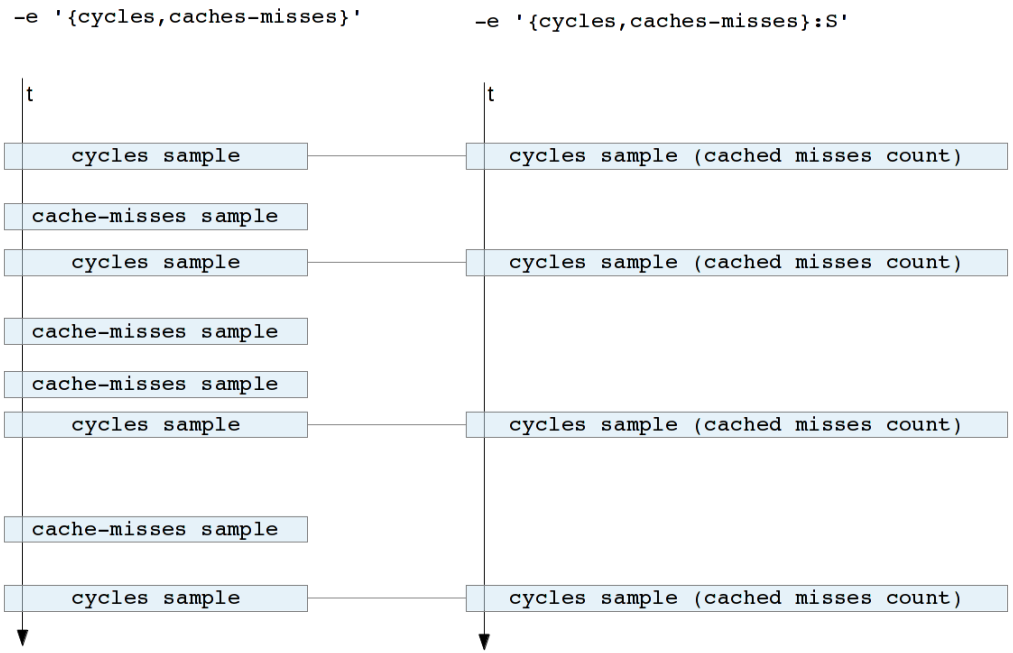
Diff enhancements - multiple data files - example

```
$ perf diff -b -o 1 -c ratio /perf.data.[1234]
# Event 'cycles'
#
# Data files:
#  [0] ./perf.data.1 (Baseline)
#  [1] ./perf.data.2
#  [2] ./perf.data.3
#  [3] ./perf.data.4
#
# Baseline/0         Ratio/1         Ratio/2         Ratio/3      Shared Object                                Symbol
# ..........  ..............  ..............  ..............  ...................  ....................................
#
       0.25%        2.911074        0.996950        2.366541  yes                  [.] fputs_unlocked@plt
       0.03%        2.033060        3.077690        2.589222  libc-2.15.so         [.] __GI___libc_write
       0.02%        2.024182        2.978838        1.005138  libc-2.15.so         [.] new_do_write
      15.02%        1.447379        1.487681        2.198029  libc-2.15.so         [.] __strlen_sse2
      14.88%        1.335010        1.870979        1.865701  libc-2.15.so         [.] fputs_unlocked
      36.44%        1.305951        1.748771        2.123142  libc-2.15.so         [.] _IO_file_xsputn@@GLIBC_2.2.5
      32.71%        1.187609        0.878003        1.999755  yes                  [.] 0x000000000000140b
       0.01%        1.057943                        1.055648  [kernel.kallsyms]    [k] unroll_tree_refs
       0.05%        1.023360        1.992786        2.052029  [kernel.kallsyms]    [k] __audit_syscall_exit
       0.03%        1.019583        1.006175        1.547110  [kernel.kallsyms]    [k] __audit_syscall_entry
       0.05%        0.992093        0.995756        2.688763  [kernel.kallsyms]    [k] system_call
       0.03%        0.986661        0.502363        2.520055  [kernel.kallsyms]    [k] fsnotify
       0.06%        0.750137        0.256142        0.257698  [kernel.kallsyms]    [k] fget_light
       0.11%        0.722499        1.139460        1.172109  [kernel.kallsyms]    [k] __srcu_read_lock
       0.03%        0.507413        1.008720        0.507093  libc-2.15.so         [.] _IO_file_overflow@@GLIBC_2.2.5
...
```

Group leader sampling

- leader sampling
- :S modifier
- `-e cycles:S`
- `-e '{cycles,cache-misses}:S'`
- attach rest of the group data to sample
- report group view by Namhyung Kim
- soon to be merged

Group leader sampling



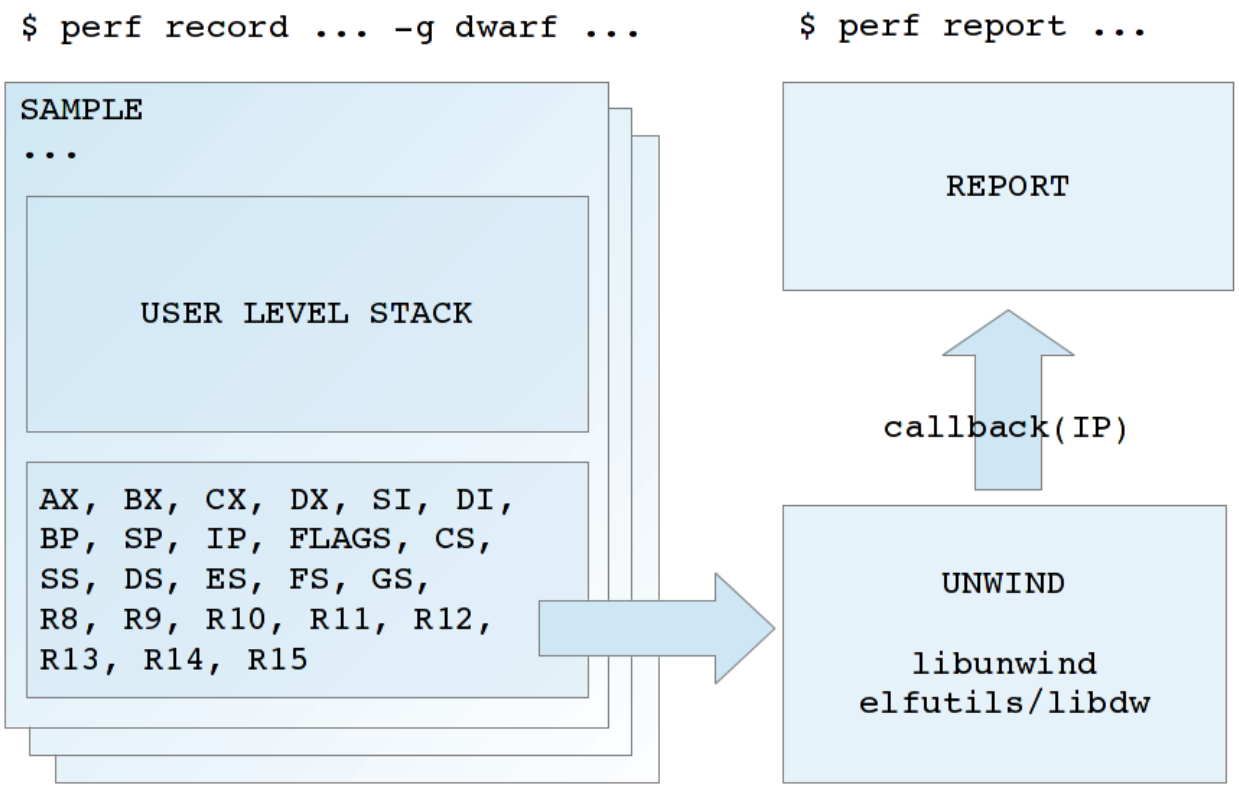
Group leader sampling - example

```
$ perf record -e '{cycles,cache-misses}:S' yes > /dev/null
^C[ perf record: Woken up 3 times to write data ]
[ perf record: Captured and wrote 0.692 MB perf.data (~30242 samples) ]
yes: Interrupt
$ perf report --group --show-total-period --stdio
# group: {cycles,cache-misses}
# ========
#
# Samples: 23K of event 'anon group { cycles, cache-misses }'
# Event count (approx.): 15458572434
#
#         Overhead                      Period  Command      Shared Object                                      Symbol
# ..............  ..........................  .......  .................  ...........................................
#
    32.52%  10.06%    5027754214          500      yes  libc-2.14.90.so    [.] _IO_file_xsputn@@GLIBC_2.2.5
    18.75%   5.84%    2898593210          290      yes  yes                [.] main
    16.84%   0.60%    2603347064           30      yes  libc-2.14.90.so    [.] __strlen_sse2
    15.39%   8.90%    2378987098          442      yes  libc-2.14.90.so    [.] fputs_unlocked
     3.50%   3.50%     540291244          174      yes  yes                [.] fputs_unlocked@plt
     2.14%   0.02%     331186976            1      yes  [kernel.kallsyms]  [k] __lock_acquire
     0.98%   0.18%     150828592            9      yes  [kernel.kallsyms]  [k] sched_clock_local
     0.96%   0.32%     148200260           16      yes  [kernel.kallsyms]  [k] debug_smp_processor_id
     0.81%  23.27%     125803028         1156      yes  [kernel.kallsyms]  [k] lock_release
     0.76%   0.00%     117272260            0      yes  [kernel.kallsyms]  [k] native_sched_clock
     0.55%   0.00%      84947064            0      yes  [kernel.kallsyms]  [k] intel_pmu_disable_all
     0.49%   0.02%      76024090            1      yes  [kernel.kallsyms]  [k] perf_event_task_tick
     0.48%   0.00%      73442096            0      yes  [kernel.kallsyms]  [k] local_clock
     0.38%   0.00%      57982898            0      yes  [kernel.kallsyms]  [k] lock_acquired
     0.33%   0.56%      51541448           28      yes  [kernel.kallsyms]  [k] lock_acquire
     0.33%  14.81%      51096950          736      yes  [kernel.kallsyms]  [k] lock_release_holdtime.part.20
     0.31%   0.00%      47667534            0      yes  [kernel.kallsyms]  [k] mark_lock
  ...
```

Callchains DWARF unwind

- kernel support for x86 only
- user level remote unwind support
- `libunwind` - no maintainer for Fedora/RHEL
- `elfutils` remote DWARF unwind support by Jan Kratochvil, pending review
- testable perf support ready

DWARF unwind



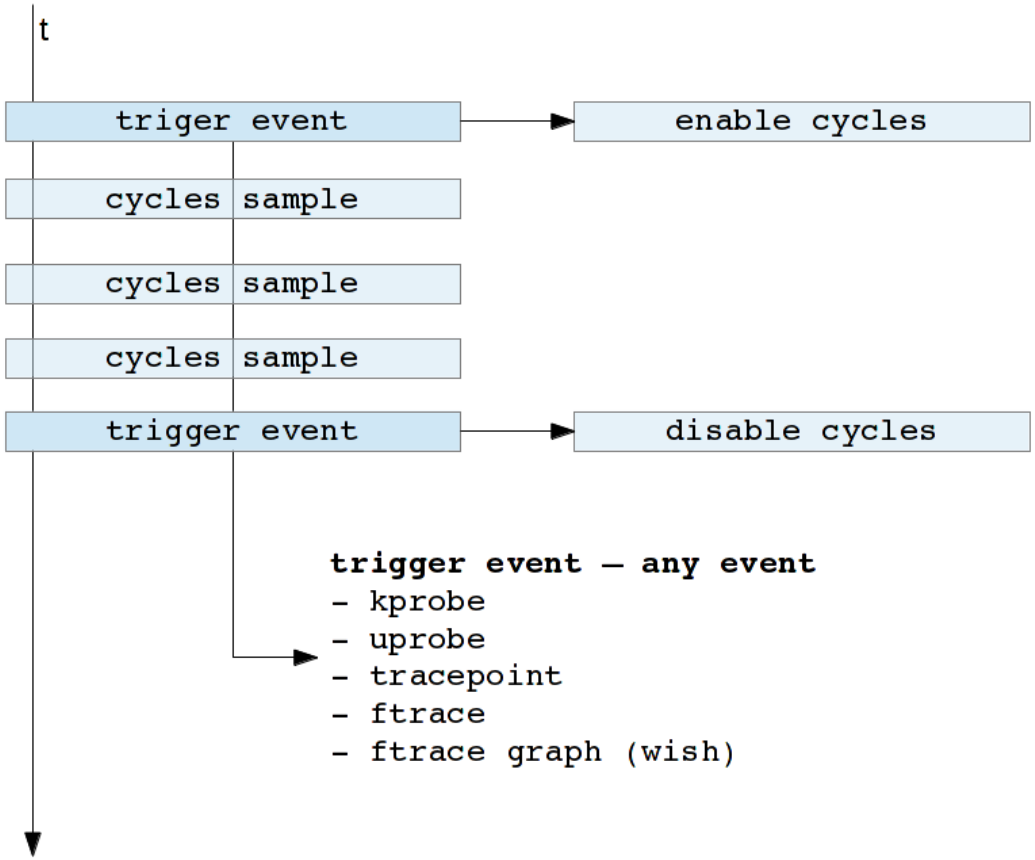
Default precise event

- different level of precise in CPUs (x86)
- sysfs exports precise level:
  `/sys/bus/event_source/devices/cpu/precise`
- `precise event by default (:p)`
- `RFC state`

Toggling events

- **want to have** feature
- Configure event to trigger another event
- Trigger - enable/disable
- Initial patchset sent by Frederic Weisbecker
- Jiri Olsa has it in testing state for recent kernels

Toggling events



Kbuild integration

- .config/Kconfig features setup
- `make *config` targets support
- Easy features/libs config
- RFC sent by Jiri Olsa

tests

- perf test getting bigger
- Ideally add a test before each fix or new feature
- Vince Weaver's tests suite
  (overflow tests ported, testing state)

Current Tests

```
[root@zoo ~]# perf test
 1: vmlinux symtab matches kallsyms                          : Ok
 2: detect open syscall event                                : Ok
 3: detect open syscall event on all cpus                    : Ok
 4: read samples using the mmap interface                    : Ok
 5: parse events tests                                       : Ok
 6: x86 rdpmc test                                           : Ok
 7: Validate PERF_RECORD_* events & perf_sample fields       : Ok
 8: Test perf pmu format parsing                             : Ok
 9: Test dso data interface                                  : Ok
10: roundtrip evsel->name check                              : Ok
11: Check parsing of sched tracepoints fields                : Ok
12: Generate and check syscalls:sys_enter_open event fields: Ok
13: struct perf_event_attr setup                             : Ok
14: Test matching and linking multiple hists                 : Ok
15: Try 'use perf' in python, checking link problems         : Ok
16: Test breakpoint overflow signal handler                  : FAILED!
17: Test breakpoint overflow sampling                        : FAILED!
18: Test number of exit event of a simple workload           : Ok
19: Test software clock events have valid period values      : Ok
[root@zoo ~]#
```

Scripting

1. Use scripting languages to process events
2. Python and Perl
3. Allows tapping into tons of language libraries
4. Several scripts available
5. Generate scripts from perf.data

Available Scripts

```
[root@aninha ~]# perf script --list
List of available trace scripts:
 syscall-counts-by-pid [comm]   system-wide syscall counts
 sctop [comm] [interval]        syscall top
 failed-syscalls-by-pid [comm]  system-wide failed syscalls
 net_dropmonitor                shows table of dropped frames
 sched-migration                sched migration overview
 netdev-times [tx] [rx] [dev=]  packet processing time
 futex-contention               futex contention measurement
 syscall-counts [comm]          system-wide syscall counts
 rw-by-pid                      system-wide r/w activity
 rwtop [interval]               system-wide r/w top
 workqueue-stats                ins/exe/create/destroy
 rw-by-file <comm>              r/w activity for a program
 failed-syscalls [comm]         system-wide failed syscalls
 wakeup-latency                 system-wide min/max/avg
[root@aninha ~]#
```

Generate Scripts

1. From the events found in perf.data file
2. Quickly start writing event handling
3. Creates function skeletons for each trace event
4. With a common set of parameters
5. Plus event specific parameters
6. Calls methods at init, exit and for unhandled events
7. Comes with library of tracing specific methods

Listing Possible probe points

```
[root@ana icmp]# perf probe -L icmp_rcv
<icmp_rcv:0>
      0  int icmp_rcv(struct sk_buff *skb)
      1  {

     59          if (rt->rt_flags & (RTCF_BROADCAST | RTCF_MULTICAST)
                         /*
                          * RFC 1122: 3.2.2.6 An ICMP_ECHO to broadca
                          * silently ignored (we let user decide with
                          * RFC 1122: 3.2.2.8 An ICMP_TIMESTAMP MAY b
                          * discarded if to broadcast/multicast.
                          */
     66                  if ((icmph->type == ICMP_ECHO ||
                               icmph->type == ICMP_TIMESTAMP) &&
                              net->ipv4.sysctl_icmp_echo_ignore_broadc
                                  goto error;
                         }
     71                  if (icmph->type != ICMP_ECHO &&
```

Listing variables that can be collected

```
[root@ana ~]# perf probe -V icmp_rcv:66
Available variables at icmp_rcv:66
        @<icmp_rcv+343>
                struct icmphdr* icmph
                struct net*     net
                struct rtable*  rt
                struct sk_buff* skb
[root@ana ~]#
```

Adding a probe

```
[root@ana icmp]# perf probe icmp_rcv:66 'type=icmph->type'
Add new event:
  probe:icmp_rcv     (on icmp_rcv:66 with type=icmph->type)

You can now use it on all perf tools, such as:

perf record -e probe:icmp_rcv -aR sleep 1

[root@ana ~]# perf probe --list
  probe:icmp_rcv (on icmp_rcv:66@net/ipv4/icmp.c with type)

[root@ana icmp]# perf record -a -g -e probe:icmp_rcv
^C[ perf record: Woken up 1 times to write data ]
[ perf record: Captured and wrote 0.324 MB perf.data ]
```

Generating a python script from perf.data

```
[root@ana icmp]# perf script -g python
generated Python script: perf-script.py

[root@ana icmp]# cat perf-script.py

def trace_begin():
        print "in trace_begin"

def trace_end():
        print "in trace_end"

def probe__icmp_rcv(evname, cpu, secs, nsecs, pid, comm,
                    probe_ip, type):
        print "%s %u.%u type=%u" % (evname, secs, nsecs, type)
```

Running python script

```
[root@ana icmp]# perf script -s perf-script.py
in trace_begin
probe__icmp_rcv 71171.964568380 type=8
probe__icmp_rcv 71177.792382154 type=8
probe__icmp_rcv 71178.792236953 type=8
in trace_end
[root@ana icmp]#
```

Backtraces from probes

```
[root@ana ~]# perf report --stdio
# Events: 2
#
# Overhead  Command      Shared Object     Symbol
# ........  .......  ..................  ........
#
   100.00%     ping  [kernel.kallsyms]  [k] icmp_rcv
               |
               --- icmp_rcv
                   ip_local_deliver_finish
                   NF_HOOK.clone.1
                   ip_local_deliver
                   ip_rcv_finish
                   NF_HOOK.clone.1
                   ip_rcv
                   __netif_receive_skb
                   process_backlog
                   net_rx_action
                   __do_softirq
                   0xb7707424

[root@ana ~]#
```

Listing probeable functions in userspace DSO

```
# perf probe -F /lib64/libc-2.12.so|grep ^m|head -10
madvise
malloc
malloc@plt
malloc_info
mblen
mbstowcs
mbtowc
mcheck
mcheck_check_all
mcheck_pedantic
[root@sandy ~]#
```

Adding userspace probe

```
[root@sandy ~]# perf probe -x /lib64/libc-2.12.so malloc
Added new event:
  probe_libc:malloc    (on 0x79b80)

You can now use it in all perf tools, such as:

perf record -e probe_libc:malloc -aR sleep 1

[root@sandy ~]#
```

Collecting callchains with stack chunks

```
# perf record -e probe_libc:* -g dwarf,1024 sleep 2
[ perf record: Woken up 1 times to write data ]
[ perf record: Captured and wrote 0.058 MB perf.data (~2547
#
```

Report snapshot

```
[root@sandy ~]# cat perf.hist.5
-  100.00%  sleep  libc-2.12.so  [.] malloc
   - malloc
      - 45.16% __strdup
         + 85.71% setlocale
         + 7.14% _nl_load_locale_from_archive
         + 7.14% __textdomain
      + 38.71% _nl_intern_locale_data
      + 6.45% _nl_normalize_codeset
      + 3.23% _nl_load_locale_from_archive
      - 3.23% new_composite_name
           setlocale
           0x4014ec
           __libc_start_main
           0x4011f9
      + 3.23% set_binding_values
[root@sandy ~]#
```

Verbose report snapshot

```
[root@sandy ~]# cat perf.hist.6
-  100.00%  sleep  libc-2.12.so  [.] malloc
   - malloc libc-2.12.so
      - 45.16% __strdup libc-2.12.so
         + 85.71% setlocale libc-2.12.so
         + 7.14% _nl_load_locale_from_archive libc-2.12.so
         + 7.14% __textdomain libc-2.12.so
      + 38.71% _nl_intern_locale_data libc-2.12.so
      + 6.45% _nl_normalize_codeset libc-2.12.so
      + 3.23% _nl_load_locale_from_archive libc-2.12.so
      - 3.23% new_composite_name libc-2.12.so
           setlocale libc-2.12.so
           0x4014ec sleep
           __libc_start_main libc-2.12.so
           0x4011f9 sleep
      + 3.23% set_binding_values libc-2.12.so
[root@sandy ~]# rpm -qf `which sleep`
coreutils-8.4-19.el6.x86_64
[root@sandy ~]# rpm -q coreutils-debuginfo
package coreutils-debuginfo is not installed
[root@sandy ~]# rpm -q glibc-debuginfo
glibc-debuginfo-2.12-1.80.el6_3.4.x86_64
[root@sandy ~]#
```

That is all folks!

Thanks!

Arnaldo Carvalho de Melo

acme@infradead.org

acme@redhat.com

Jiri Olsa - jolsa@redhat.com

linux-perf-users@vger.kernel.org