LTTng 2.1 : Advanced Linux tracing for everyone

Yannick Brosseau

LCA 2013

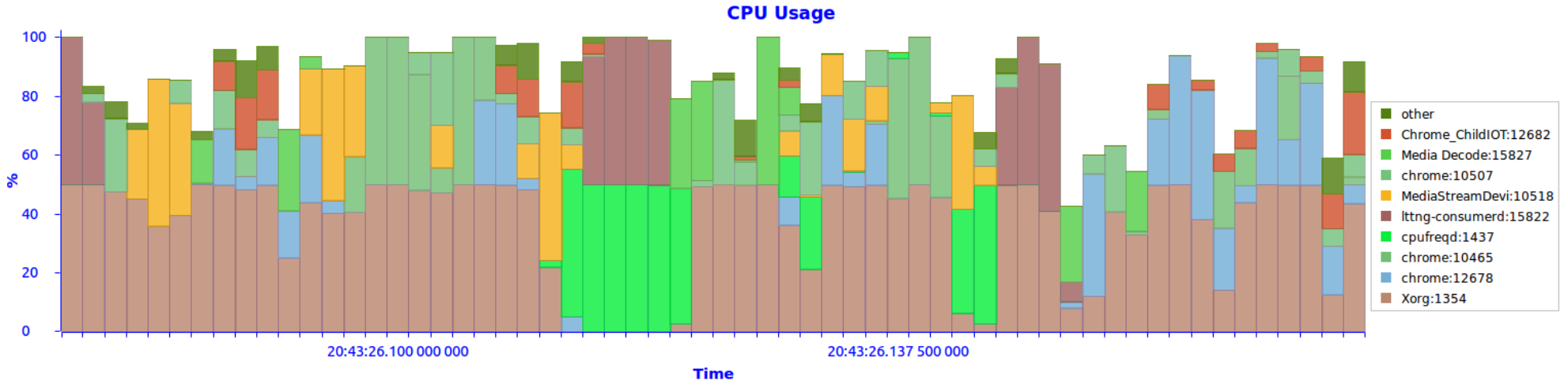
POLYTECHNIQUE
MONTRÉAL

LE GÉNIE
EN PREMIÈRE CLASSE

ct Run Window Help

Quick Access | Resource | LTTng Kernel

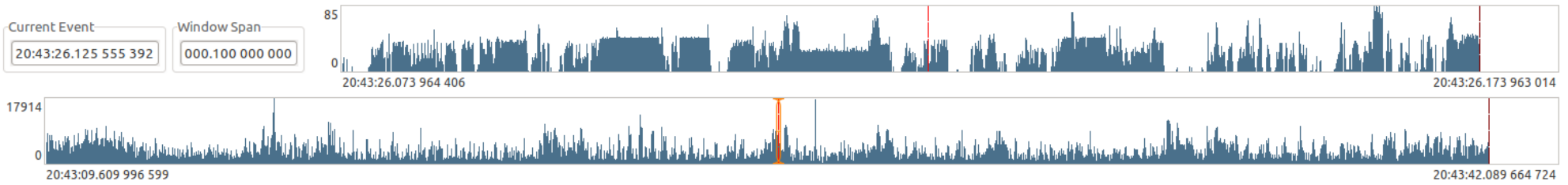
Control Flow | Resources | Statistics | CPU Usage



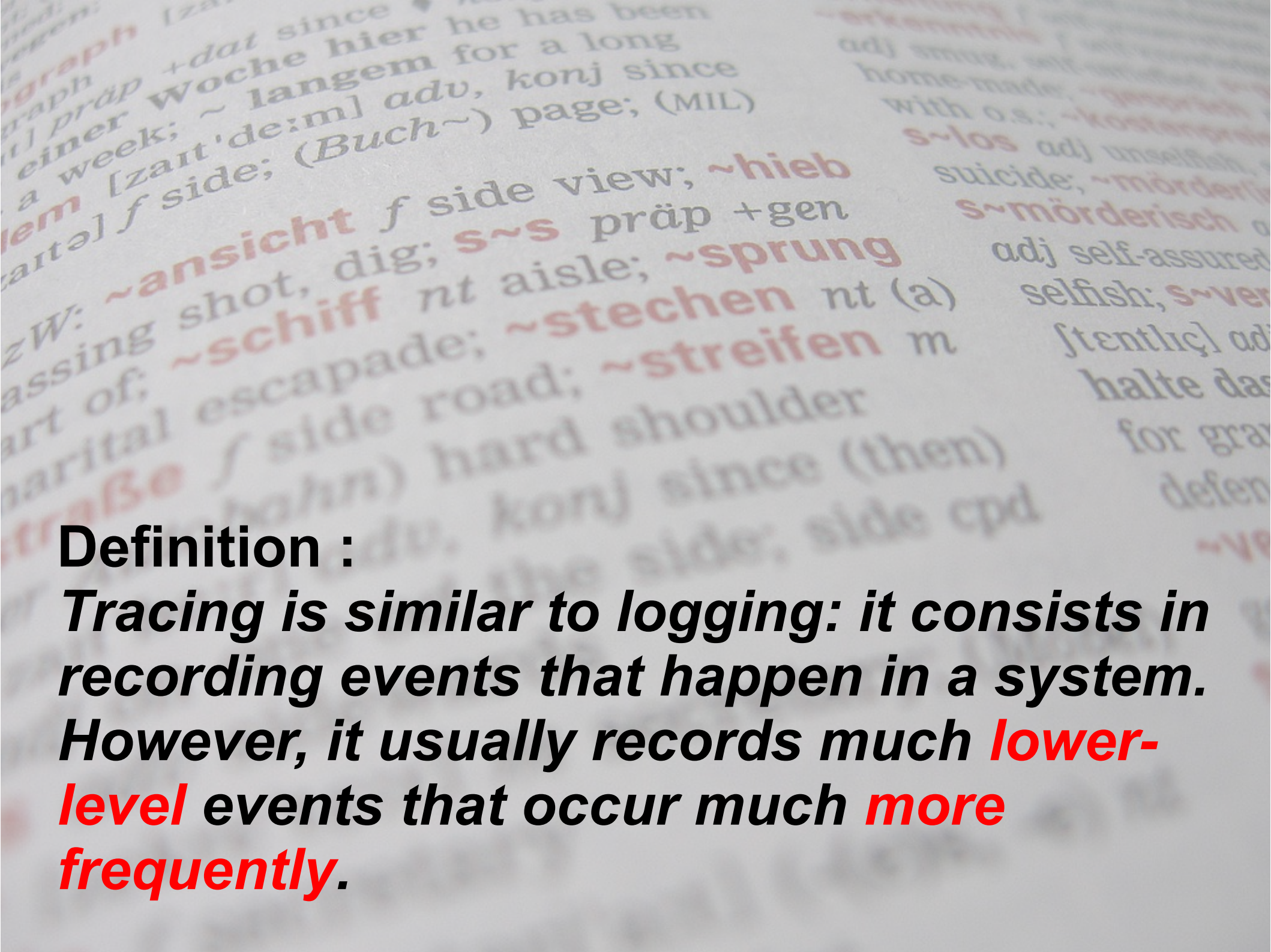
YoutubeHD

Timestamp	Channel	Event Type	Content
<srch>	<srch>	<srch>	<srch>
20:43:26.125 534 23	channel0_0	exit_syscall	ret=0
20:43:26.125 537 72	channel0_0	sys_clock_gettime	which_clock=1, tp=0xbfdb8728
20:43:26.125 540 16	channel0_0	exit_syscall	ret=0
20:43:26.125 543 72	channel0_0	sys_clock_gettime	which_clock=1, tp=0xbfdb8778
20:43:26.125 545 96	channel0_0	exit_syscall	ret=0
20:43:26.125 553 01	channel0_0	sys_clock_gettime	which_clock=1, tp=0xbfdb8758
20:43:26.125 555 39	channel0_0	exit_syscall	ret=0
20:43:26.125 559 72	channel0_0	sys_clock_gettime	which_clock=1, tp=0xbfdb8728
20:43:26.125 562 02	channel0_0	exit_syscall	ret=0
20:43:26.125 565 93	channel0_0	sys_clock_gettime	which_clock=1, tp=0xbfdb8828
20:43:26.125 568 17	channel0_0	exit_syscall	ret=0
20:43:26.125 570 05	channel0_0	sys_clock_gettime	which_clock=1, tp=0xbfdb8728
20:43:26.125 572 43	channel0_0	exit_syscall	ret=0
20:43:26.125 574 52	channel0_0	sys_clock_gettime	which_clock=1, tp=0xbfdb8778

Histogram | Properties | Bookmarks

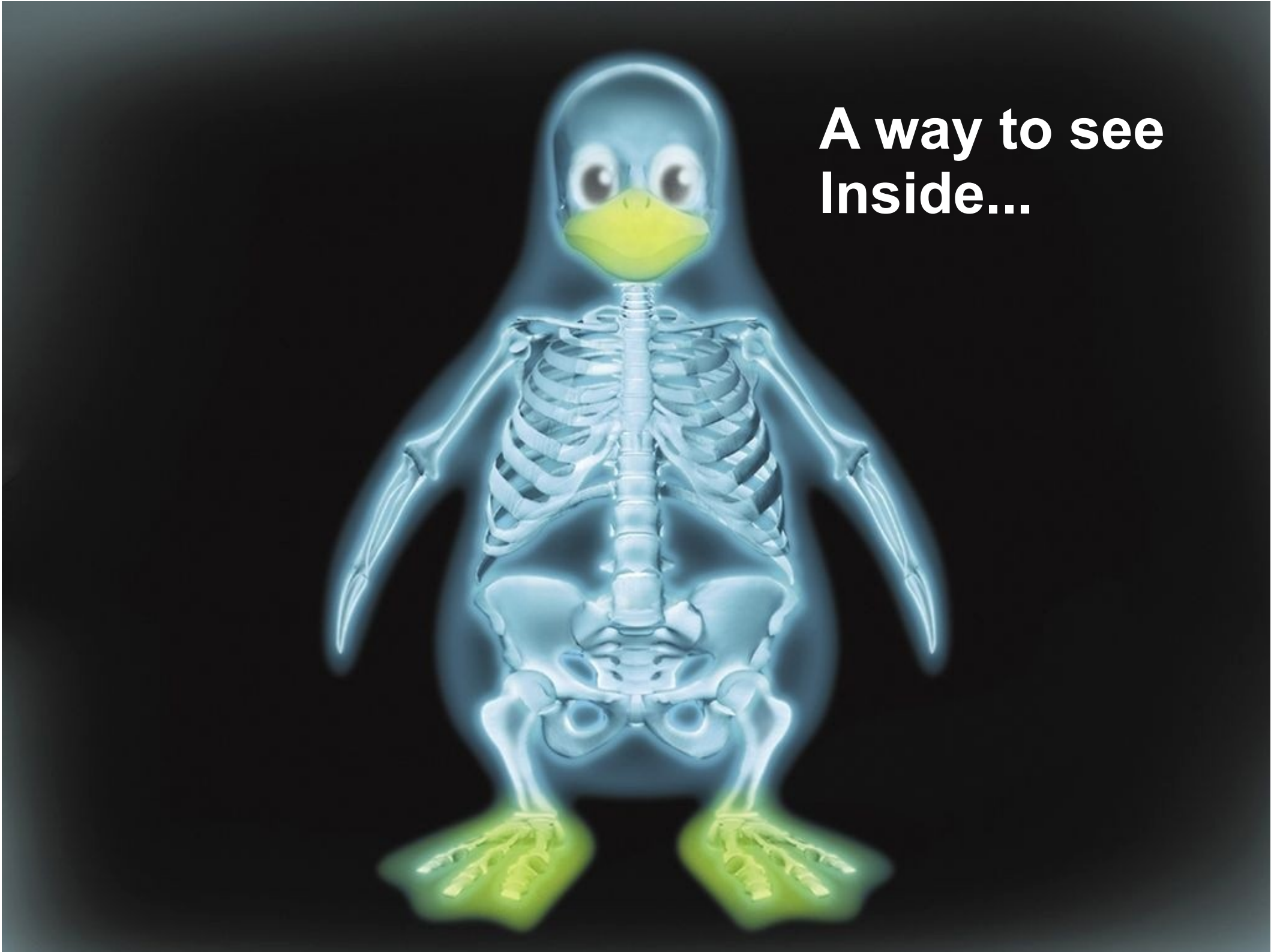


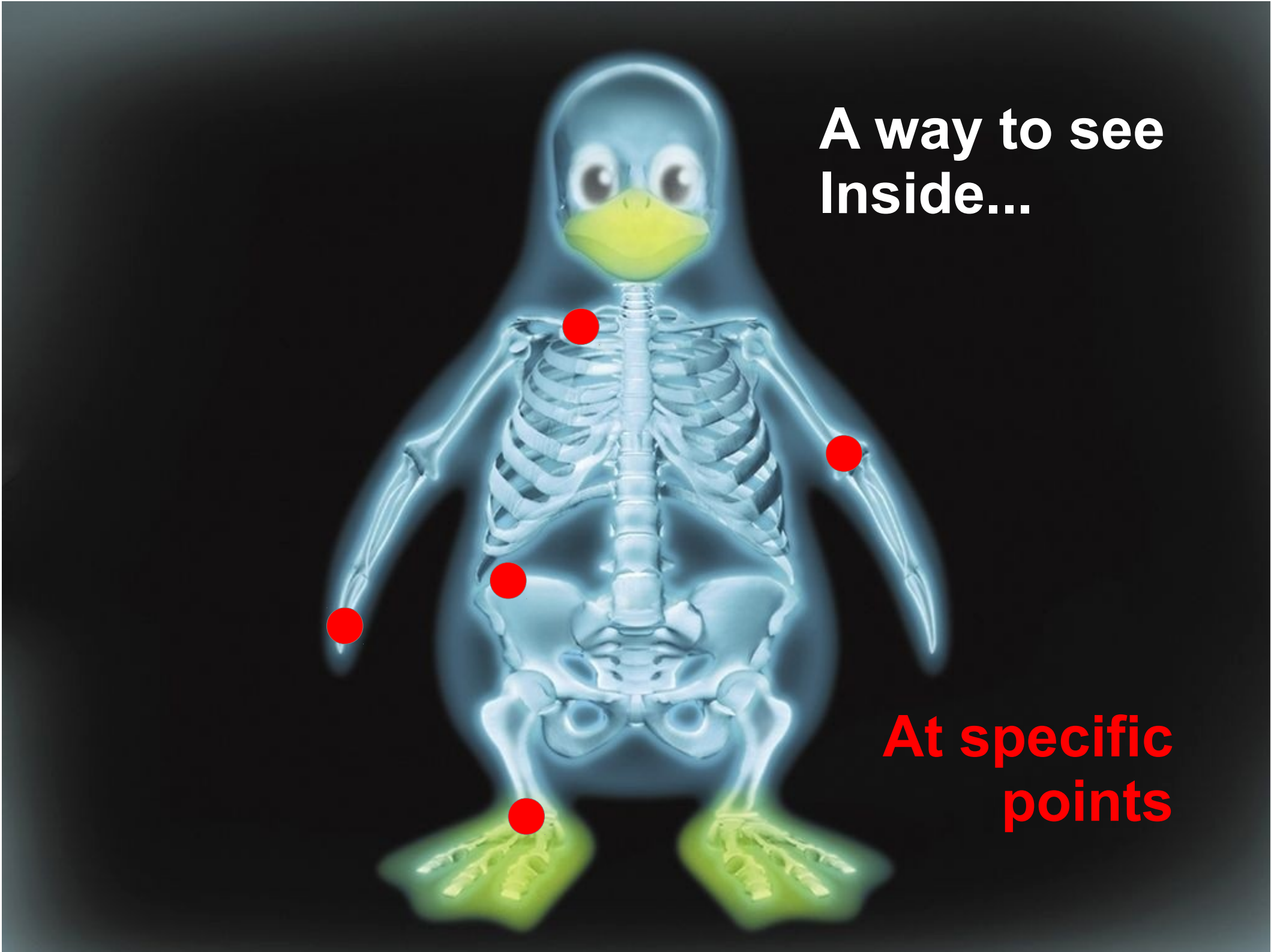

What is
Tracing?

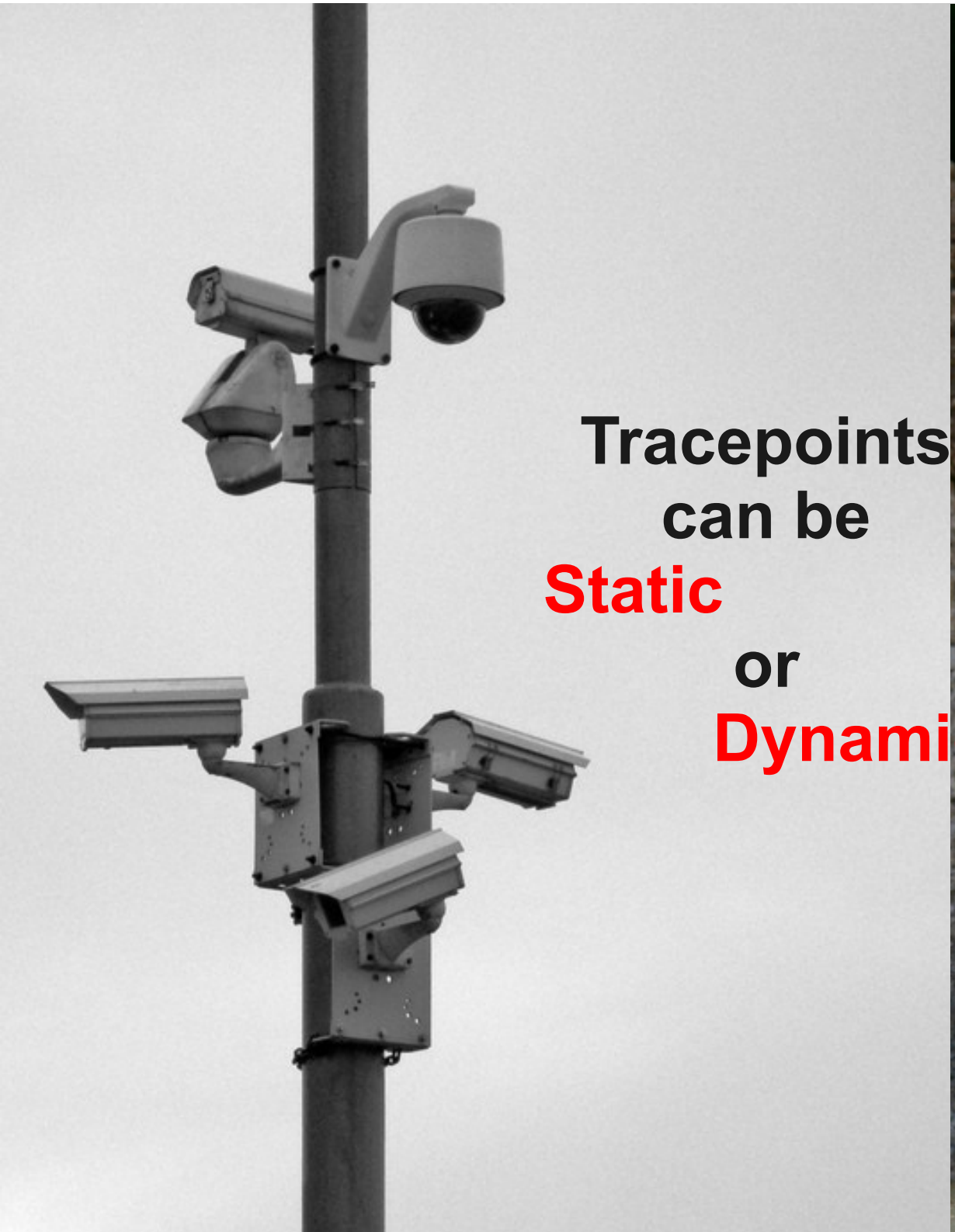


Definition :

Tracing is similar to logging: it consists in recording events that happen in a system. However, it usually records much lower-level events that occur much more frequently.


A way to see
Inside...


A way to see
Inside...
At specific
points



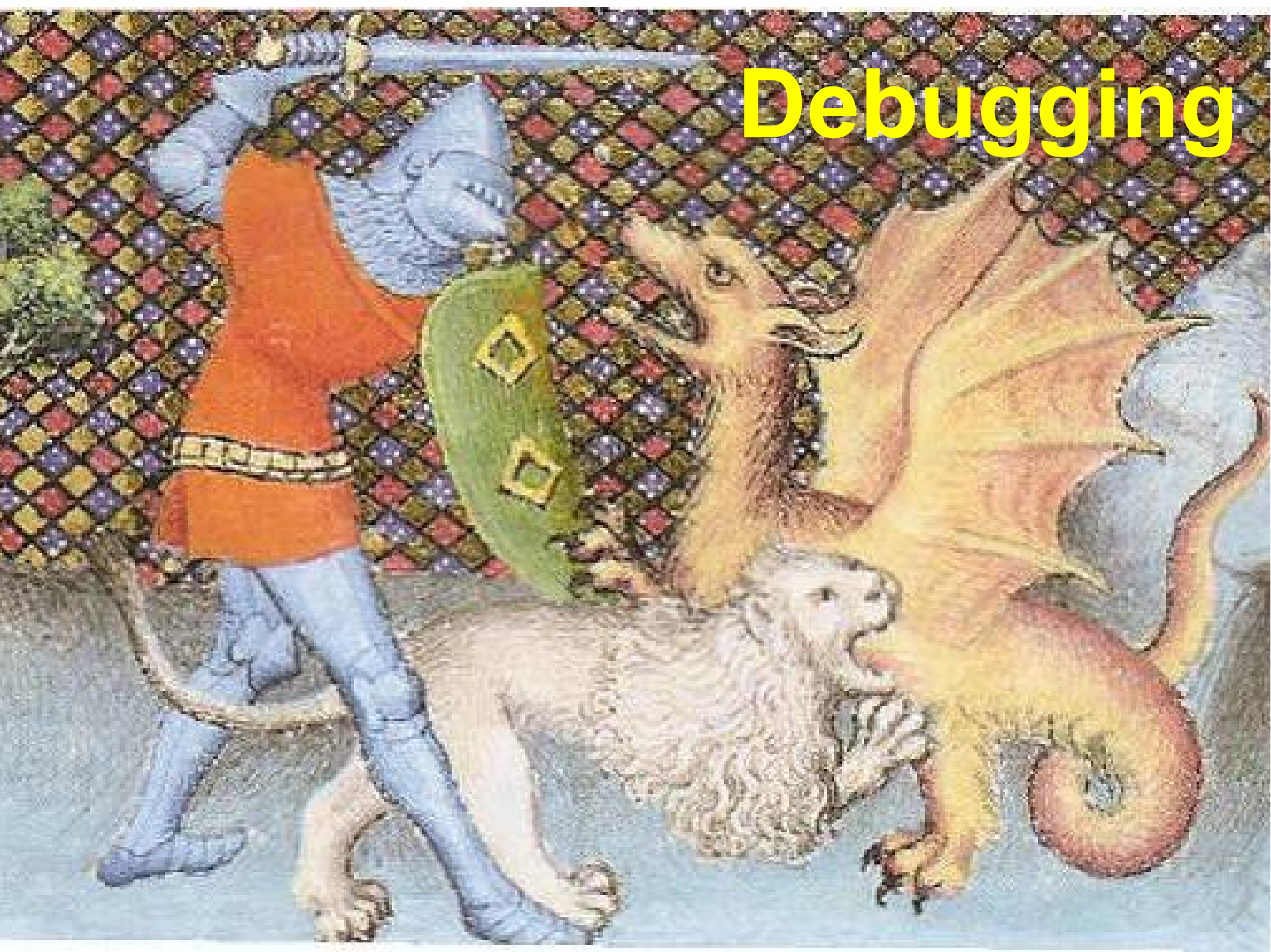
Tracepoints
can be
Static
or
Dynamic

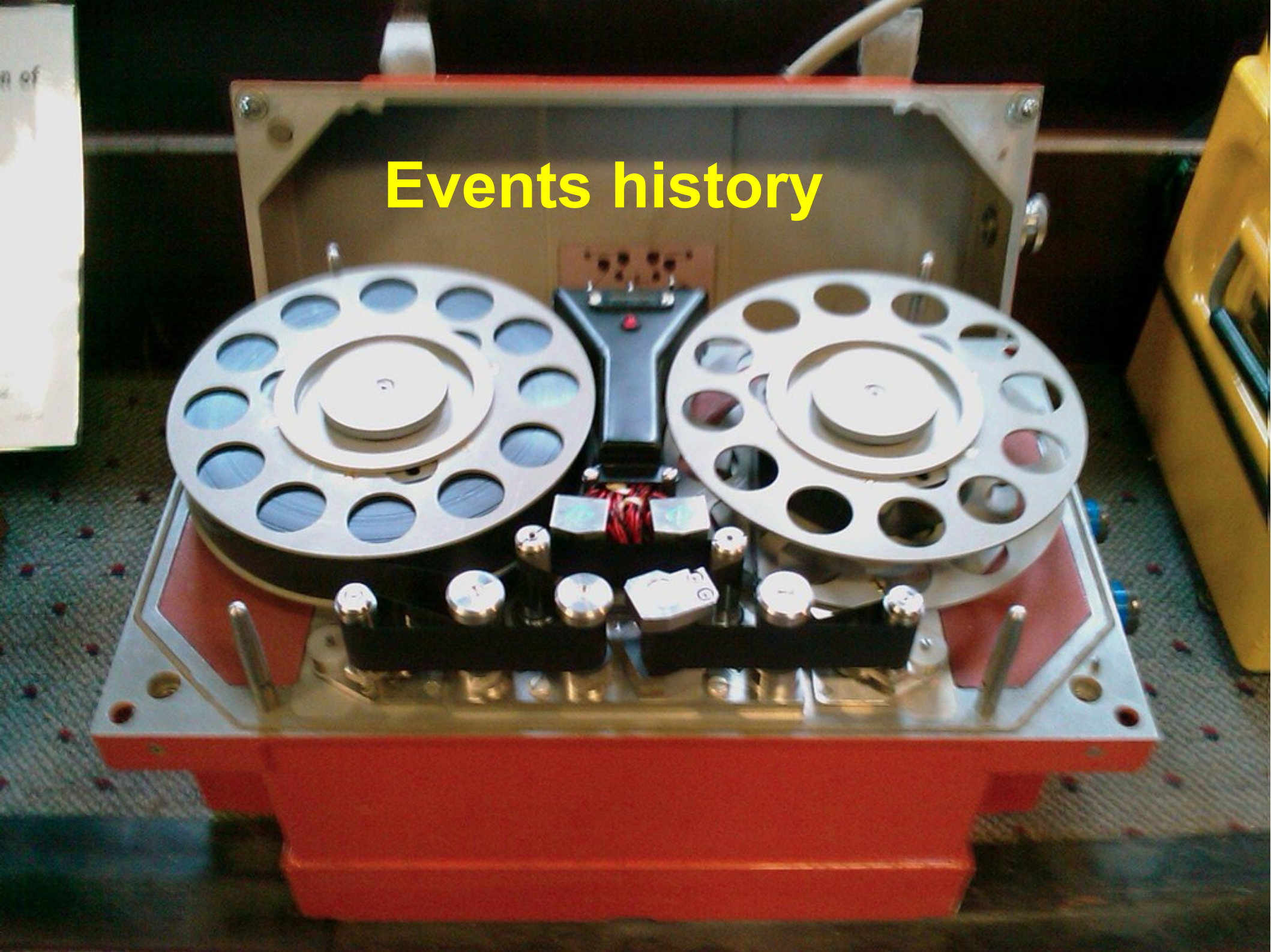
Tracepoints characteristics :

- High speed
- Low latency
- Timestamp
- Payload

What can it be used for ?


Learning


Debugging

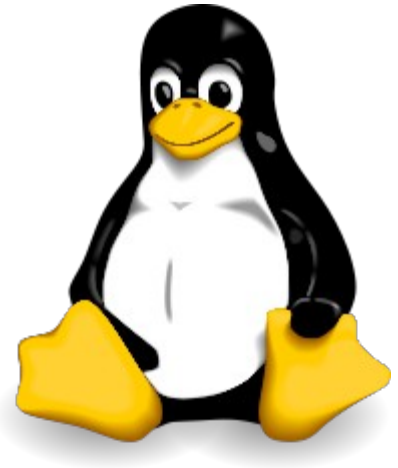

Events history

What is LTTng?

Tracers Timeline

- 1999 : LTT
- 2005 : LTTng 0.x
- 2005 : Dtrace
- 2005 : SystemTap
- 2008 : Ftrace
- 2009 : Perf
- 2012 : LTTng 2.0
- 2013 : LTTng 2.1

Userspace and kernel tracers



What's so special about LTTng 2.x?

Tracing for Everyone?


easy
Installation

No kernel patch required

- Kernel tracing
  - Only kernel modules
  - Support kernel from 2.6.38
  - Runs on 2.6.32 with 3 small kernel patches
- Userspace tracing
  - Don't rely on the kernel

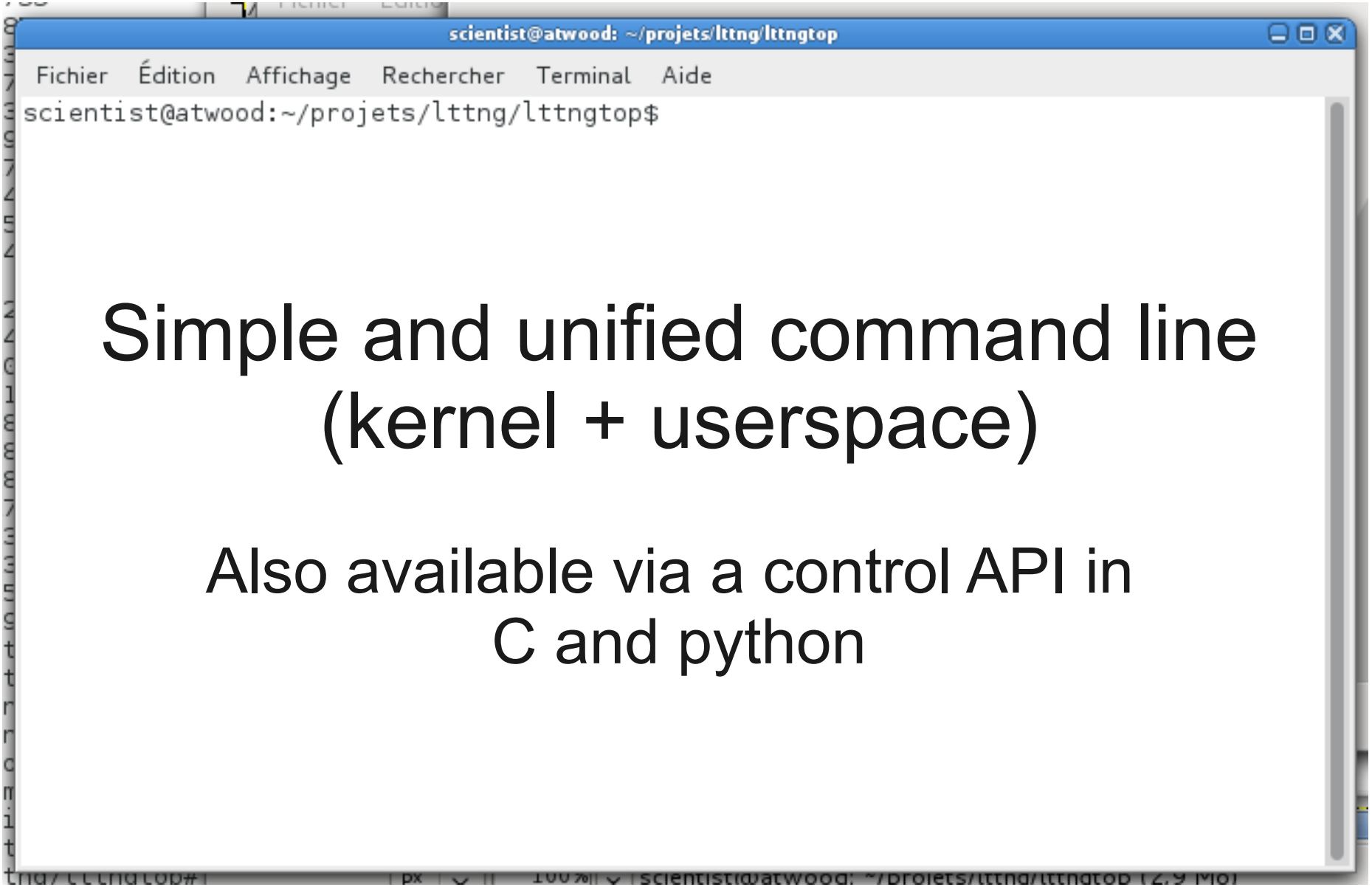
Linux distribution

- Ubuntu
- Debian
- Fedora (without kernel modules)
- Arch
- Suse
- Red Hat (pending)

Multi-arch and other OS

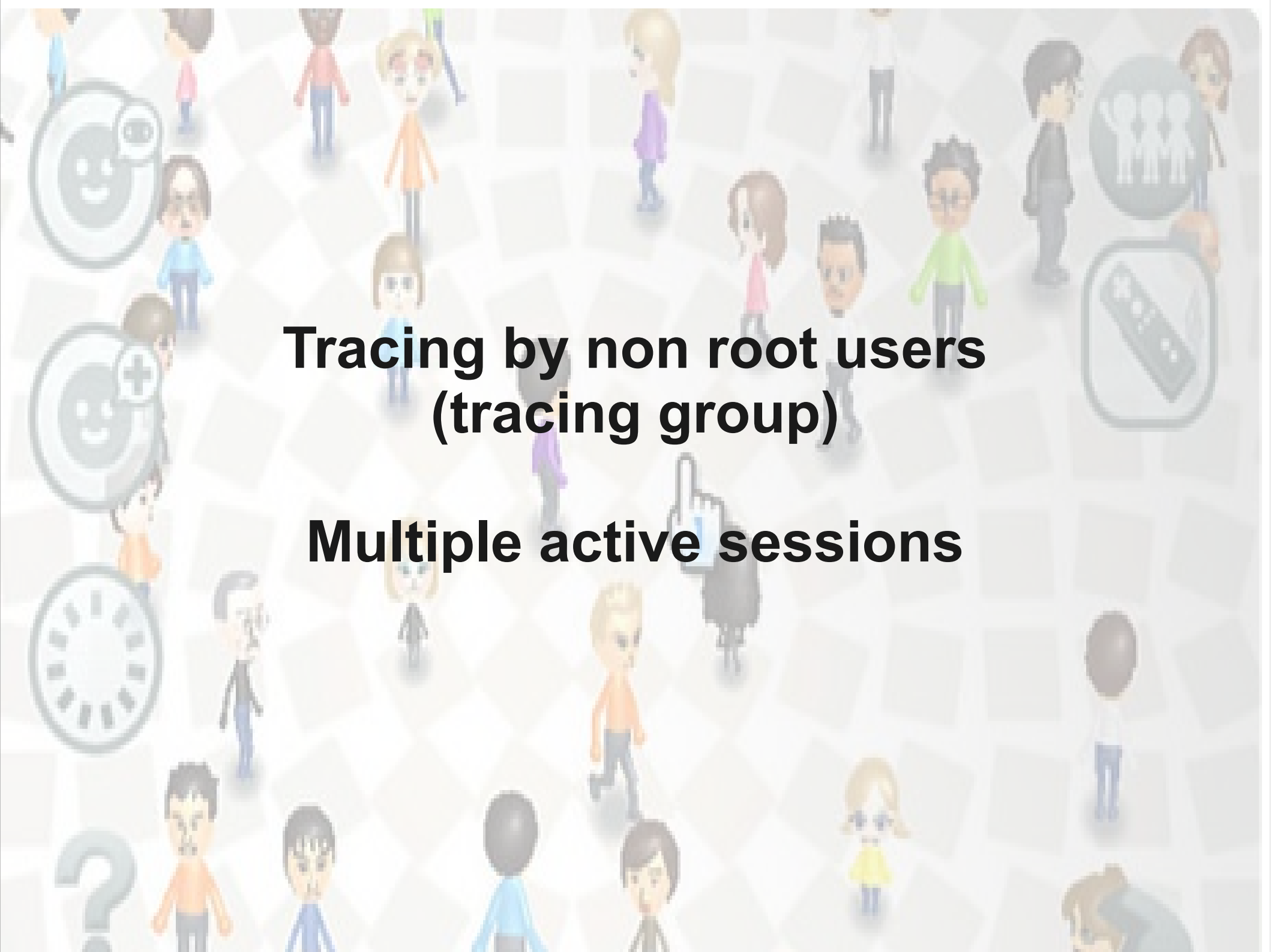
- x86 and x86-64
- ARM
- PowerPC, Sparc, Mips, Tile, Xeon Phi

- Android (Ongoing)
- FreeBSD (UST)
- Cygwin (UST)

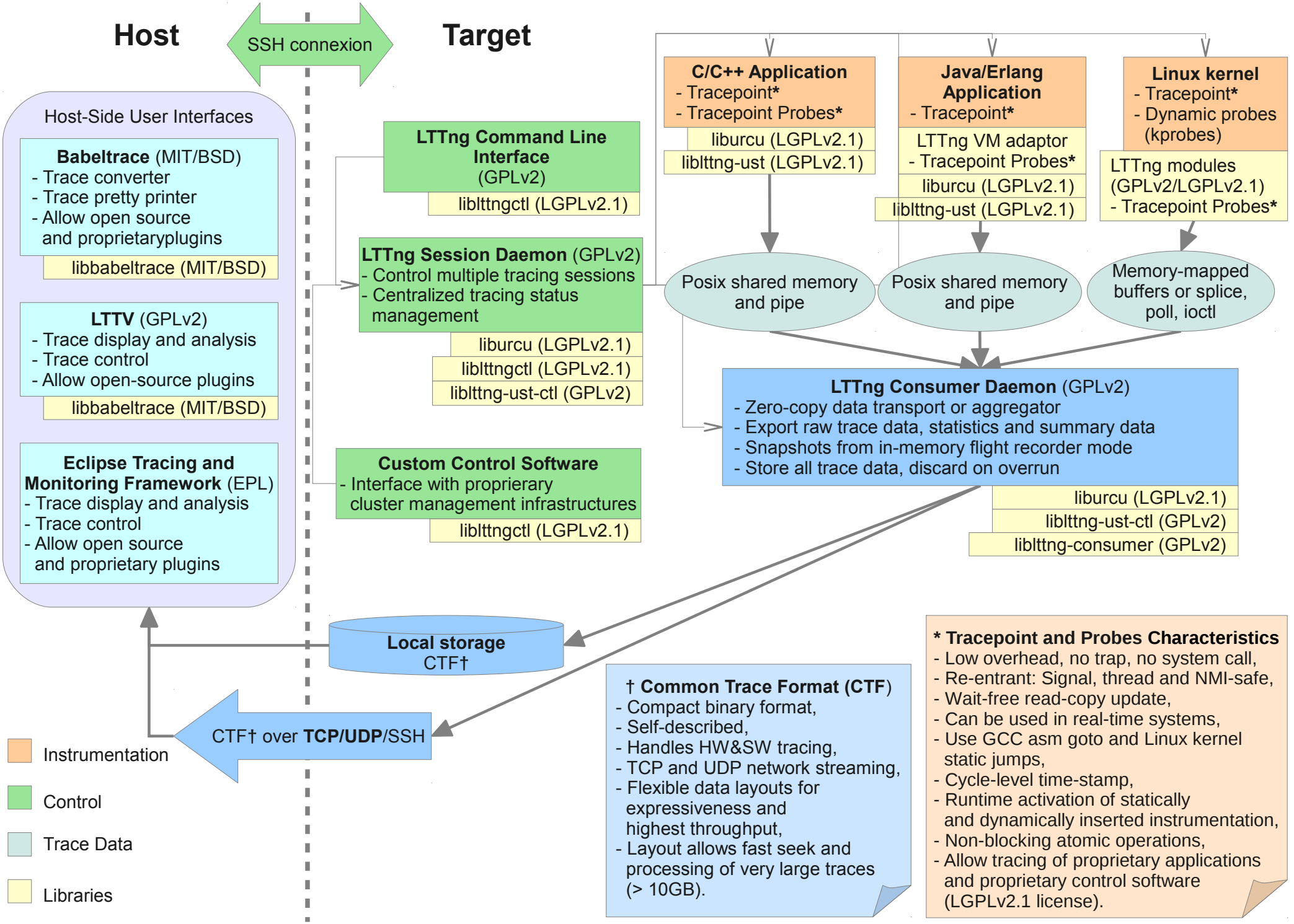

scientist@atwood: ~/projets/lttng/lttngtop
Fichier Édition Affichage Rechercher Terminal Aide
scientist@atwood:~/projets/lttng/lttngtop$
Simple and unified command line
(kernel + userspace)
Also available via a control API in
C and python

(create, enable, start, stop, view, destroy)

```
# lttng create
# lttng enable-event -k -a
# lttng enable-event -u -a
# lttng start
# lttng stop
# lttng view
# lttng destroy
```


Tracing by non root users
(tracing group)
Multiple active sessions

LTTng 2.0 Low-Overhead Tracing Architecture



Multiples data sources...

- Kernel :
  - Static Tracepoints
  - Syscalls
  - Kprobes
  - Function tracer
  - Context (including perf counters)
- Userspace
  - Static Tracepoints
    - With filtering
  - Context

… That produce CTF data

- Common Trace Format, defined by the MultiCore Association

Kernel tracepoint anatomy

```
TRACE_EVENT(sched_process_fork,

	TP_PROTO(struct task_struct *parent, struct task_struct *child),

	TP_ARGS(parent, child),

	TP_STRUCT__entry(
		__array(	char,	parent_comm,	TASK_COMM_LEN	)
		__field(	pid_t,	parent_pid			)
		__array(	char,	child_comm,	TASK_COMM_LEN	)
		__field(	pid_t,	child_pid			)
	),

	TP_fast_assign(
		memcpy(__entry->parent_comm, parent->comm, TASK_COMM_LEN);
		__entry->parent_pid	= parent->pid;
		memcpy(__entry->child_comm, child->comm, TASK_COMM_LEN);
		__entry->child_pid	= child->pid;
	),

	TP_printk("comm=%s pid=%d child_comm=%s child_pid=%d",
		__entry->parent_comm, __entry->parent_pid,
		__entry->child_comm, __entry->child_pid)
);
```

Simple userspace tracepoint generation

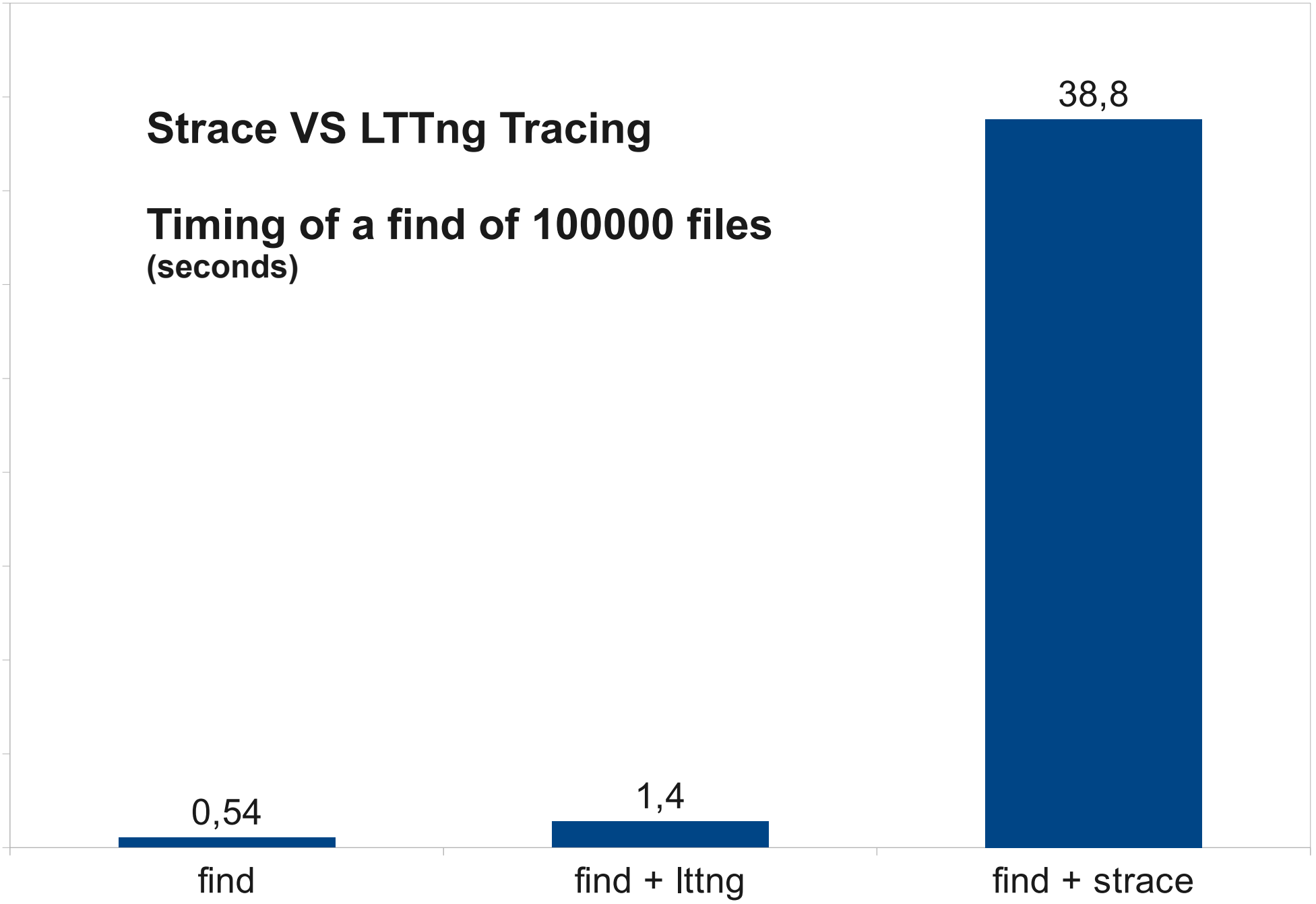
```
TRACEPOINT_EVENT(
        sample_tracepoint,
        message,
        TP_ARGS(char *, text),
        TP_FIELDS(
                ctf_string(message, text)
                )
)
```

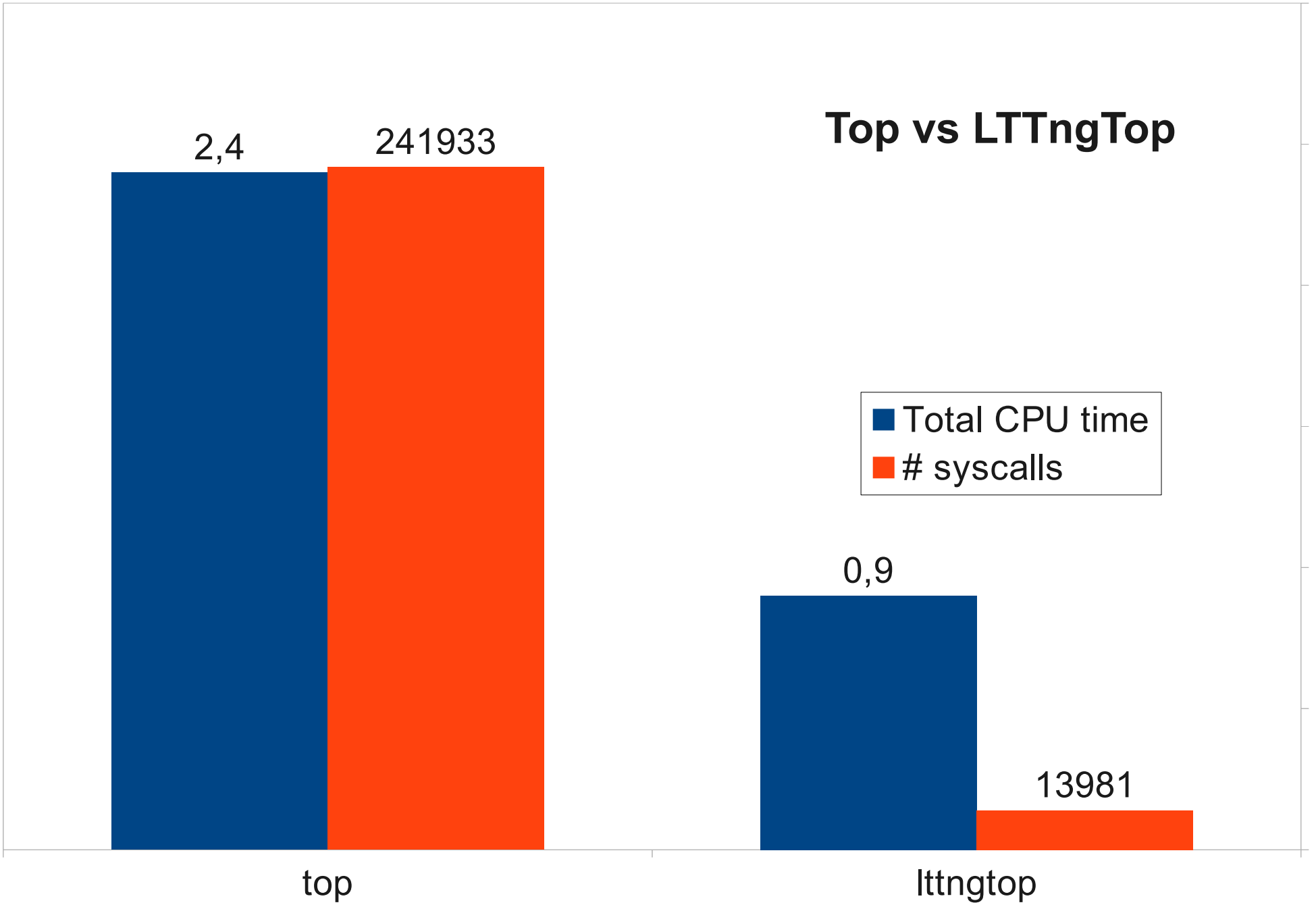
Generated by lttng-gen-tp :

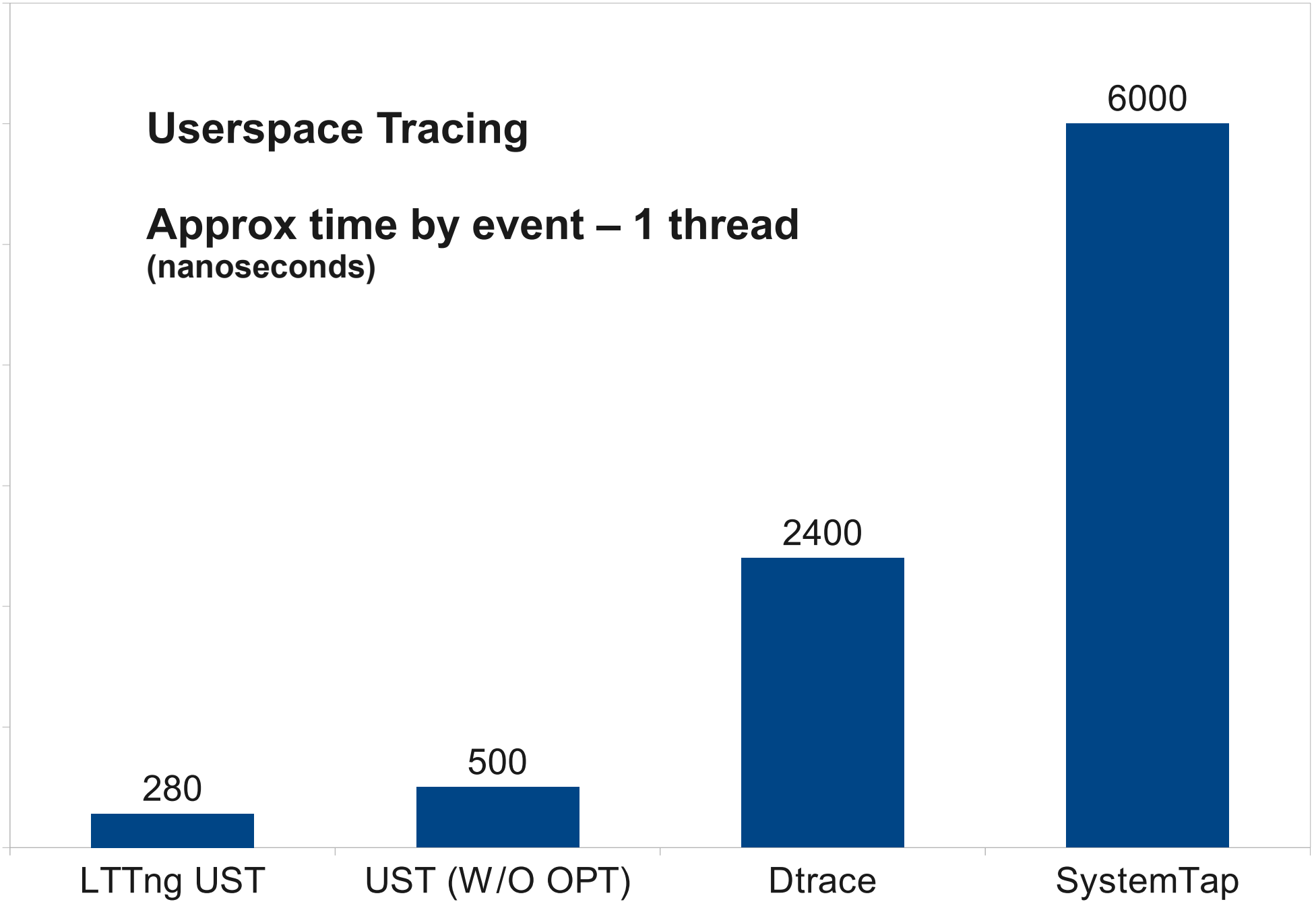
In C :

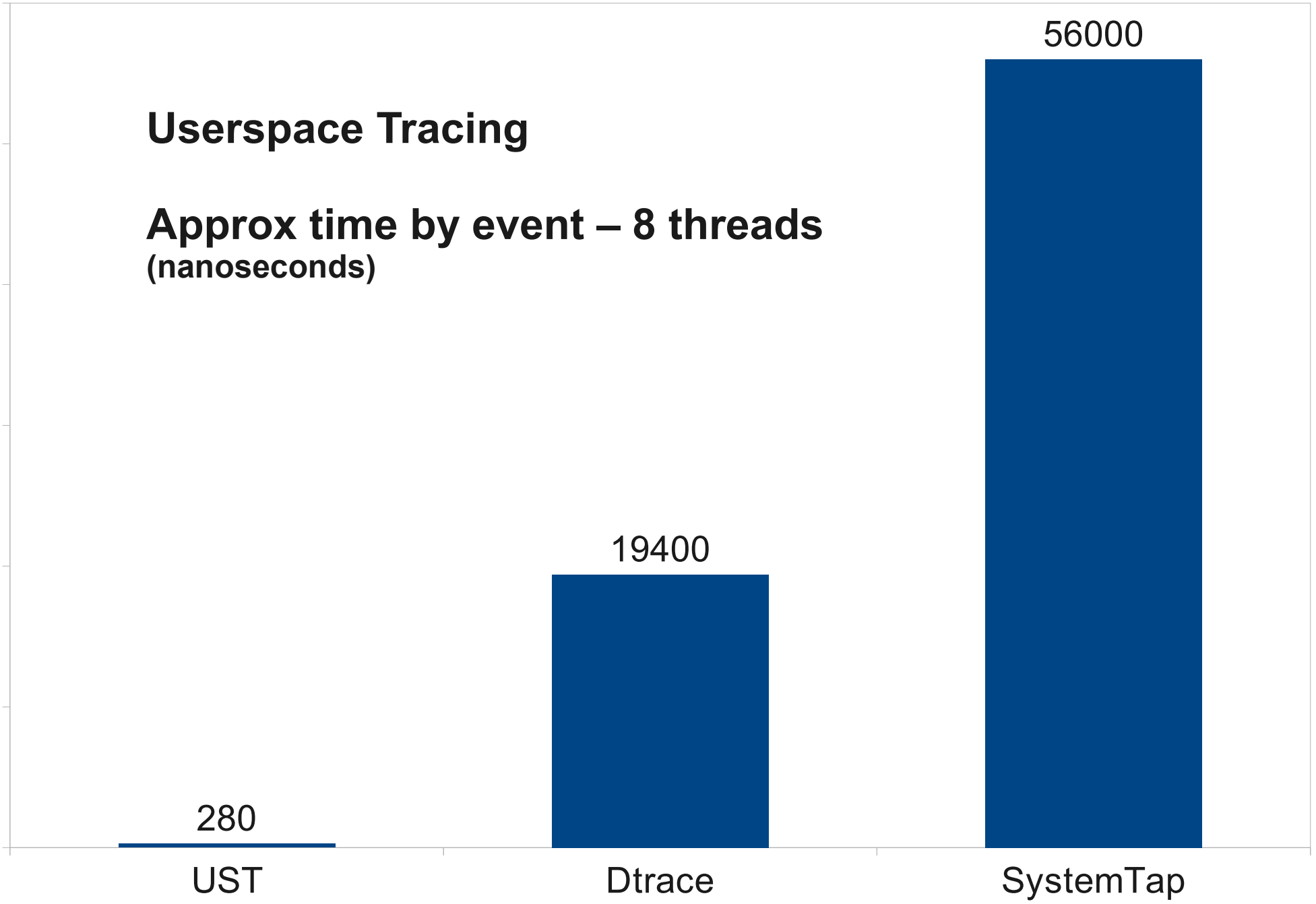
```
#include "sample_tracepoint.h"
tracepoint(sample_tracepoint, message,
                    "Hello World\n");
```


FAST
heart heart
96.1
96.1


Strace VS LTTng Tracing
Timing of a find of 100000 files
(seconds)
38,8
0,54
1,4
find
find + lttng
find + strace


Top vs LTTngTop
2,4
241933
0,9
13981
Total CPU time
syscalls
top
lttngtop

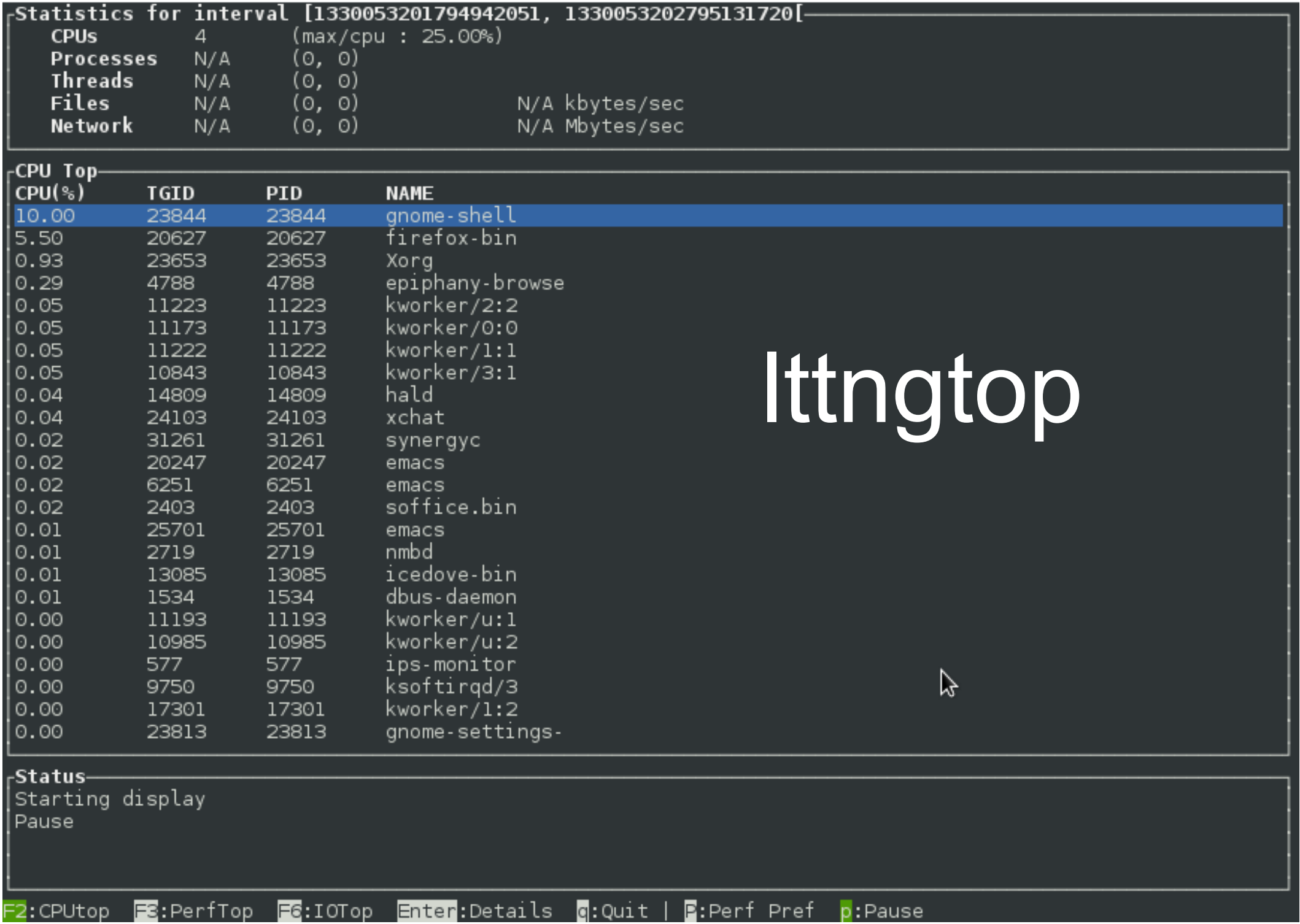

Userspace Tracing
Approx time by event – 1 thread
(nanoseconds)
280
500
2400
6000
LTTng UST
UST (W/O OPT)
Dtrace
SystemTap


Userspace Tracing
Approx time by event – 8 threads
(nanoseconds)
280
19400
56000
UST
Dtrace
SystemTap

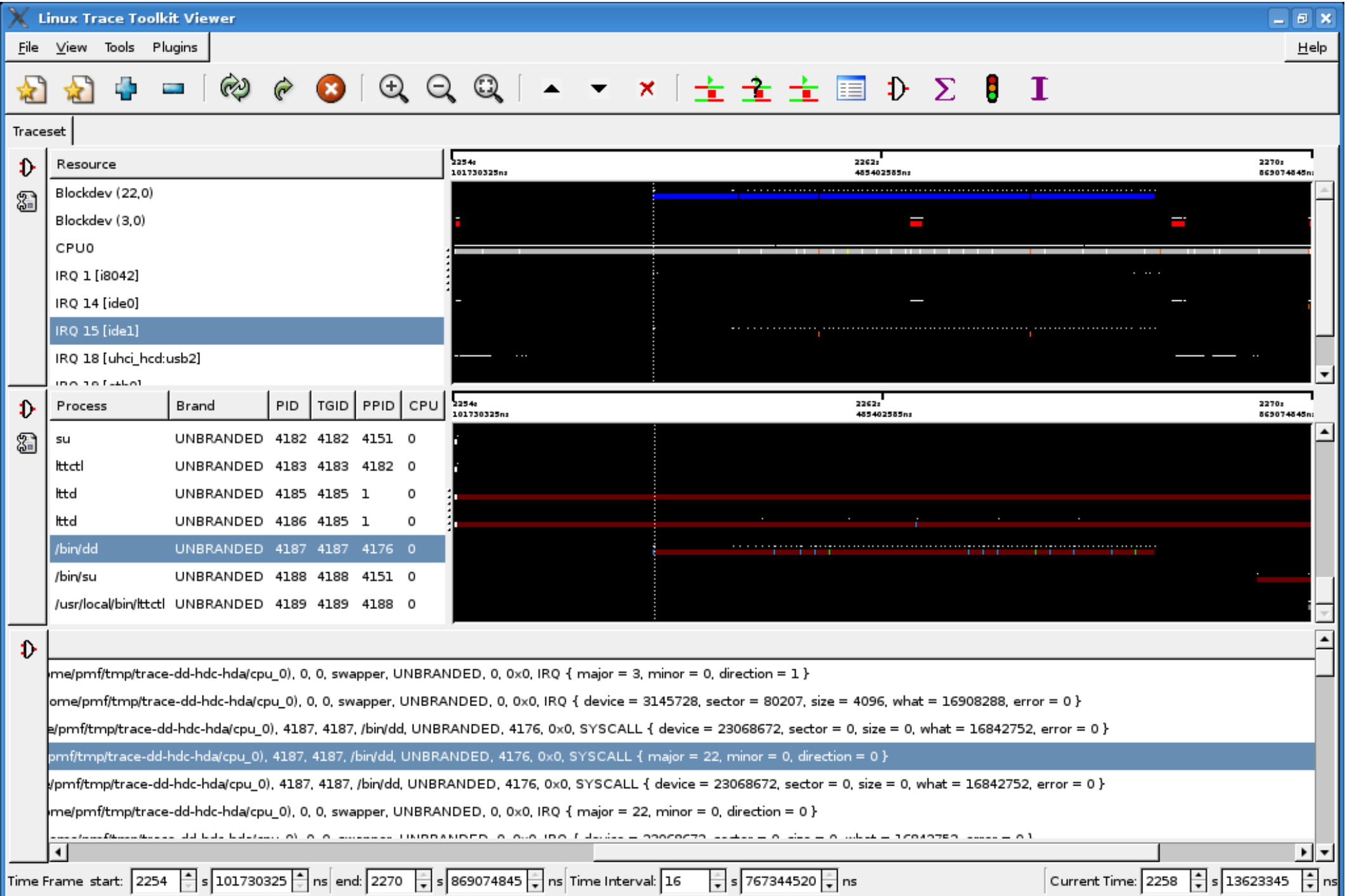
Looking at the trace data

Babeltrace

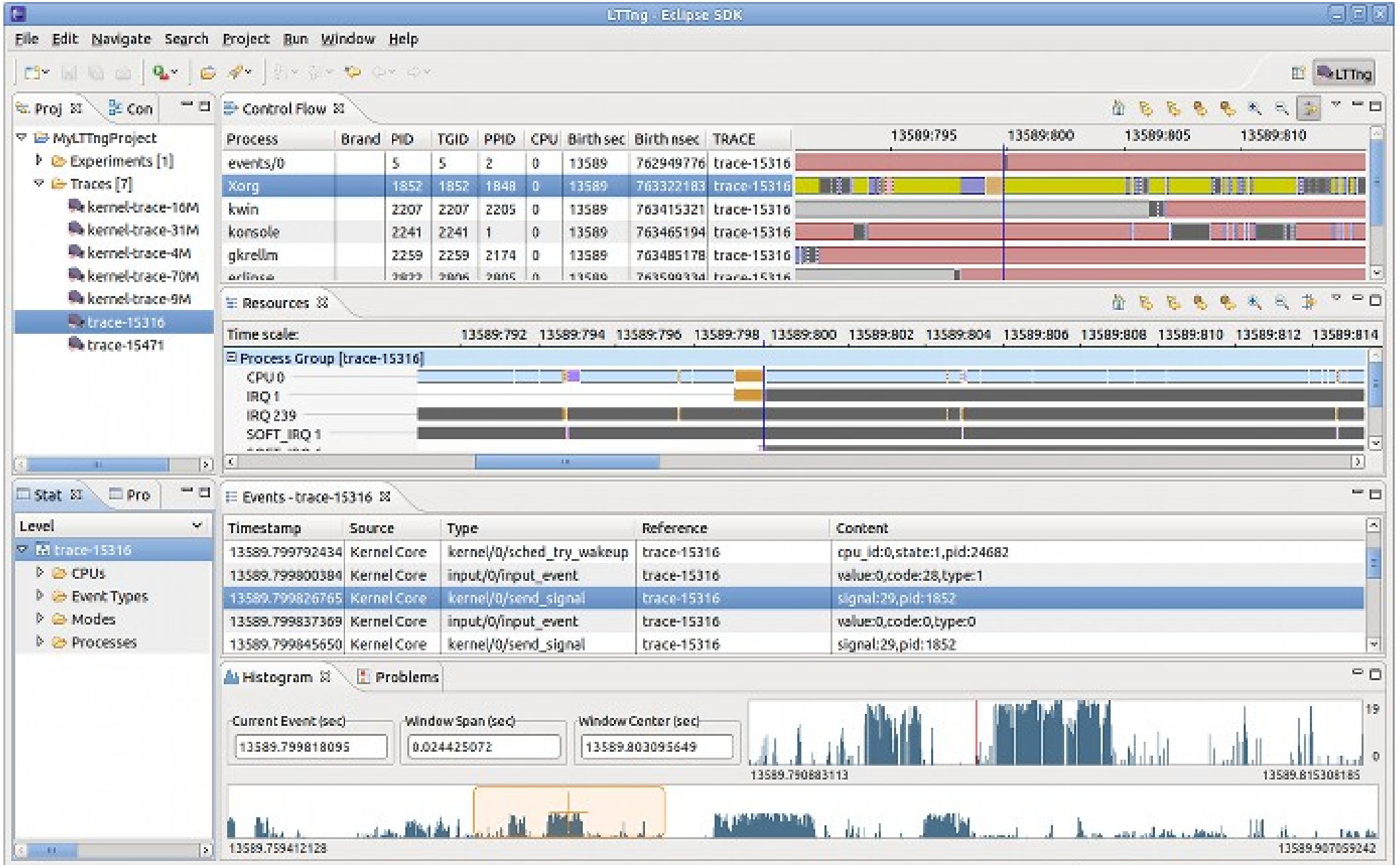
```
[13:58:29.128909723] (+0.000002475) sys_read: { 0 }, { "firefox-bin", 3363 }, { fd = 5, buf =
 count = 16 }
[13:58:29.128911513] (+0.000001790) exit_syscall: { 0 }, { "firefox-bin", 3363 }, { ret = -11
[13:58:29.128919672] (+0.000008159) sys_write: { 0 }, { "firefox-bin", 3363 }, { fd = 5, buf
, count = 8 }
[13:58:29.128921404] (+0.000001732) exit_syscall: { 0 }, { "firefox-bin", 3363 }, { ret = 8 }
[13:58:29.128922884] (+0.000001480) sys_read: { 0 }, { "firefox-bin", 3363 }, { fd = 19, buf
, count = 1 }
[13:58:29.128925765] (+0.000002881) exit_syscall: { 0 }, { "firefox-bin", 3363 }, { ret = 1 }
[13:58:29.128928120] (+0.000002355) sys_write: { 0 }, { "firefox-bin", 3363 }, { fd = 5, buf
, count = 8 }
[13:58:29.128929552] (+0.000001432) exit_syscall: { 0 }, { "firefox-bin", 3363 }, { ret = 8 }
[13:58:29.129020005] (+0.000090453) exit_syscall: { 0 }, { "acpid", 1536 }, { ret = 1 }
[13:58:29.129025587] (+0.000005582) sys_rt_sigprocmask: { 0 }, { "acpid", 1536 }, { how = 0,
 oset = 0x0, sigsetsize = 8 }
[13:58:29.129027993] (+0.000002406) exit_syscall: { 0 }, { "acpid", 1536 }, { ret = 0 }
[13:58:29.129030188] (+0.000002195) sys_poll: { 0 }, { "acpid", 1536 }, { ufds = 0x7FFF2A055D
meout_msecs = 0 }
[13:58:29.129032570] (+0.000002382) exit_syscall: { 0 }, { "acpid", 1536 }, { ret = 0 }
[13:58:29.129033929] (+0.000001359) sys_rt_sigprocmask: { 0 }, { "acpid", 1536 }, { how = 1,
 oset = 0x0, sigsetsize = 8 }
[13:58:29.129035144] (+0.000001215) exit_syscall: { 0 }, { "acpid", 1536 }, { ret = 0 }
[13:58:29.129037520] (+0.000002376) sys_read: { 0 }, { "acpid", 1536 }, { fd = 4, buf = 0x7FF
 = 24 }
```


Statistics for interval [1330053201794942051, 1330053202795131720[
CPUs 4 (max/cpu : 25.00%)
Processes N/A (0, 0)
Threads N/A (0, 0)
Files N/A (0, 0) N/A kbytes/sec
Network N/A (0, 0) N/A Mbytes/sec
CPU Top
CPU(%) TGID PID NAME
10.00 23844 23844 gnome-shell
5.50 20627 20627 firefox-bin
0.93 23653 23653 Xorg
0.29 4788 4788 epiphany-browse
0.05 11223 11223 kworker/2:2
0.05 11173 11173 kworker/0:0
0.05 11222 11222 kworker/1:1
0.05 10843 10843 kworker/3:1
0.04 14809 14809 hald
0.04 24103 24103 xchat
0.02 31261 31261 synergyc
0.02 20247 20247 emacs
0.02 6251 6251 emacs
0.02 2403 2403 soffice.bin
0.01 25701 25701 emacs
0.01 2719 2719 nmbd
0.01 13085 13085 icedove-bin
0.01 1534 1534 dbus-daemon
0.00 11193 11193 kworker/u:1
0.00 10985 10985 kworker/u:2
0.00 577 577 ips-monitor
0.00 9750 9750 ksoftirqd/3
0.00 17301 17301 kworker/1:2
0.00 23813 23813 gnome-settings-
lttngtop
Status
Starting display
Pause
F2:CPUtop F3:PerfTop F6:IOTop Enter:Details q:Quit | P:Perf Pref p:Pause

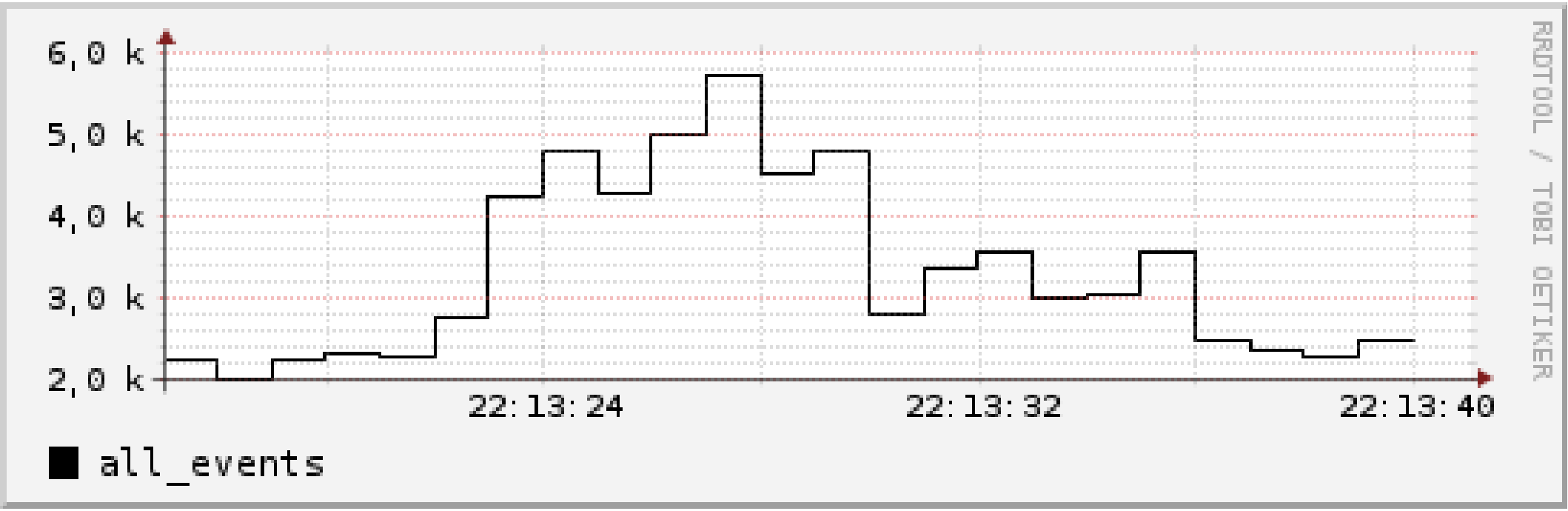
lttv


Linux Trace Toolkit Viewer
File View Tools Plugins
Help
Traceset
Resource
Blockdev (22,0)
Blockdev (3,0)
CPU0
IRQ 1 [i8042]
IRQ 14 [ide0]
IRQ 15 [ide1]
IRQ 18 [uhci_hcd:usb2]
Process Brand PID TGID PPID CPU
su UNBRANDED 4182 4182 4151 0
lttctl UNBRANDED 4183 4183 4182 0
lttd UNBRANDED 4185 4185 1 0
lttd UNBRANDED 4186 4185 1 0
/bin/dd UNBRANDED 4187 4187 4176 0
/bin/su UNBRANDED 4188 4188 4151 0
/usr/local/bin/lttctl UNBRANDED 4189 4189 4188 0
2254s 101730325ns
2262s 485402585ns
2270s 869074845ns
me/pmf/tmp/trace-dd-hdc-hda/cpu_0), 0, 0, swapper, UNBRANDED, 0, 0x0, IRQ { major = 3, minor = 0, direction = 1 }
ome/pmf/tmp/trace-dd-hdc-hda/cpu_0), 0, 0, swapper, UNBRANDED, 0, 0x0, IRQ { device = 3145728, sector = 80207, size = 4096, what = 16908288, error = 0 }
e/pmf/tmp/trace-dd-hdc-hda/cpu_0), 4187, 4187, /bin/dd, UNBRANDED, 4176, 0x0, SYSCALL { device = 23068672, sector = 0, size = 0, what = 16842752, error = 0 }
pmf/tmp/trace-dd-hdc-hda/cpu_0), 4187, 4187, /bin/dd, UNBRANDED, 4176, 0x0, SYSCALL { major = 22, minor = 0, direction = 0 }
/pmf/tmp/trace-dd-hdc-hda/cpu_0), 4187, 4187, /bin/dd, UNBRANDED, 4176, 0x0, SYSCALL { device = 23068672, sector = 0, size = 0, what = 16842752, error = 0 }
me/pmf/tmp/trace-dd-hdc-hda/cpu_0), 0, 0, swapper, UNBRANDED, 0, 0x0, IRQ { major = 22, minor = 0, direction = 0 }
Time Frame start: 2254 s 101730325 ns end: 2270 s 869074845 ns Time Interval: 16 s 767344520 ns Current Time: 2258 s 13623345 ns

Eclipse Linux Tools



lttng-graph


6,0 k
5,0 k
4,0 k
3,0 k
2,0 k
22:13:24
22:13:32
22:13:40
all_events
RRDTOOL / TOBI OETIKER

The Road Ahead

Live tracing

Per user global buffers

Remote tracing

Filtering (kernel)

uprobes

Other language binding

lttv

What about research?

- Depency analysis
- Multi level event abstraction
- Time synchronisation
- Integration of hardware tracing
- Tracing the cloud!

Demo!

Contact

- Me :
  - http://yannickbrosseau.com
  - yannick.brosseau@gmail.com
  - @greenscientist
- LTTng project :
  - http://lttng.org
  - lttng-dev@lists.lttng.org
  - irc.oftc.net #lttng

http://lttng.org/download

Image references

https://secure.flickr.com/photos/andreweadie/3746534415
https://secure.flickr.com/photos/-bast-/34949798
https://secure.flickr.com/photos/jwyg/3746351826
https://secure.flickr.com/photos/pedromourapinheiro/4270640863
https://secure.flickr.com/photos/puuikibeach/4781504955
https://secure.flickr.com/photos/b4b2/3954106061
https://secure.flickr.com/photos/edvvc/1972546648
https://secure.flickr.com/photos/anirudhkoul/2632880868
https://secure.flickr.com/photos/iita-media-library/6031540791
https://secure.flickr.com/photos/38471709@N02/5353281594
https://commons.wikimedia.org/wiki/File:Yvain-drgaon.jpg
https://commons.wikimedia.org/wiki/File:Chevrolet_Volt_under_the_hood_WAS_2011_1107.jpg
http://www.flickr.com/photos/15708236@N07/2754478731/
https://commons.wikimedia.org/wiki/File:Mount_Royal_Montreal_Lookout.jpg