

Trace Compass Incubator

Progress Report Meeting,
École Polytechnique de Montréal
December 7, 2017

Geneviève Bastien
Research Associate

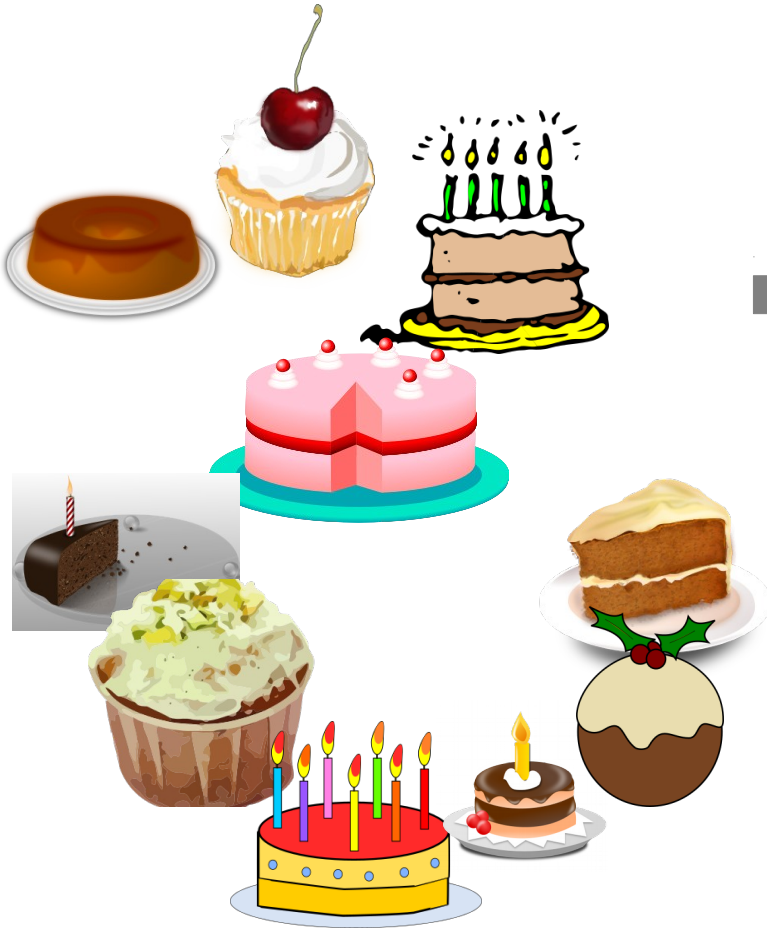


Remember these? Now let's see the menu!

Prototype



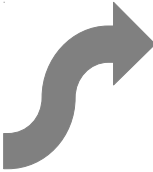
Incubator



Trace Compass master



Users



Users





Trace Types (and some analyses on them)

Json Trace-event traces

MessageLoop::RunTask	2593788.989 000 000	E	INFO	30674	30649	task		
MessageLoop::PostTask	2593788.989 003 000	f	INFO	30674	30649	task	0xaf954be3aa	{queue_duration=1}
ImmediateInputRouter::ProcessInput	2593788.989 003 000	B	INFO	30649	30649	input		{ack=CONSUMED}
MessageLoop::RunTask	2593788.989 004 000	B	INFO	30674	30649	task		{src_file=../../content/public/browser/browser_messa

- UfTrace traces

237206.127 440 475	entry	0	malloc	14819	14819	"/usr/bin/x86_64-linux-gnu-gcc-6"
237206.127 440 542	exit	0	malloc	14819	14819	"/usr/bin/x86_64-linux-gnu-gcc-6"
237206.127 440 952	entry	0	__cxa_atexit	14819	14819	"/usr/bin/x86_64-linux-gnu-gcc-6"

- Perf sampling traces

1504633868.175 610	perf_stream-0-0	0	cycles:ppp	perf_ip=0x7f699f8a3cf0, perf_tid=6825, perf_pid=6801, perf_period=5171732, perf_callchain_size=6, perf_callchain=[0xfffffffffff00, 0x7f699f8a3cf0]		
1504633868.238 302	perf_stream-0-0	0	perf_mmap2	pid=6948, tid=6948, start=0x7f5a0ab0f000, filename=/usr/lib/python3.6/lib-dynload/_json.cpython-36m-x86_64-linux-gnu.so		
1504633868.263 728	perf_stream-0-0	0	perf_exit	pid=6948, ppid=6948, tid=6948, ptid=6948, time=609964377326		
1504633868.264 657	perf_stream-0-0	0	perf_fork	pid=6950, ppid=6801, tid=6950, ptid=6875, time=609965306757		
1504633868.264 701	perf_stream-0-0	0	cycles:ppp	perf_ip=0xffffffff89064504, perf_tid=6950, perf_pid=6950, perf_period=1, perf_callchain_size=34, perf_callchain=[0xfffffffffff80, 0xffffffff89064504,		
1504633868.264 710	perf_stream-0-0	0	cycles:ppp	perf_ip=0xffffffff89064504, perf_tid=6950, perf_pid=6950, perf_period=1, perf_callchain_size=34, perf_callchain=[0xfffffffffff80, 0xffffffff89064504,		

- Trace Compass JUL Log traces

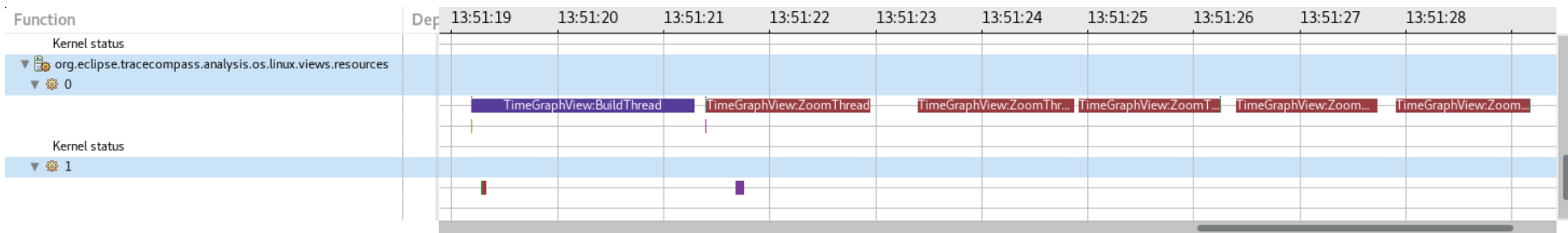
1504633863.250 859	ltnng_jul_channel_3	3	ltnng_jul:event	msg={"ts":604809333817,"ph":"i","tid":1,"name":"TimeGraphView.LoadingTrace","args":{"trace":"many-threads","viewId":"org.eclipse.tracecompass.anal		
1504633863.251 058	ltnng_jul_channel_3	3	ltnng_jul:event	msg={"ts":604951620732,"ph":"s","tid":1,"name":"TimeGraphView.Rebuilding","cat":"org.eclipse.tracecompass.analysis.os.linux.views.controlflow","id":		
1504633863.251 105	ltnng_jul_channel_3	3	ltnng_jul:event	msg={"ts":604951734417,"ph":"s","tid":1,"name":"RefreshRequested","cat":"org.eclipse.tracecompass.analysis.os.linux.views.controlflow","id":"0x10"},		



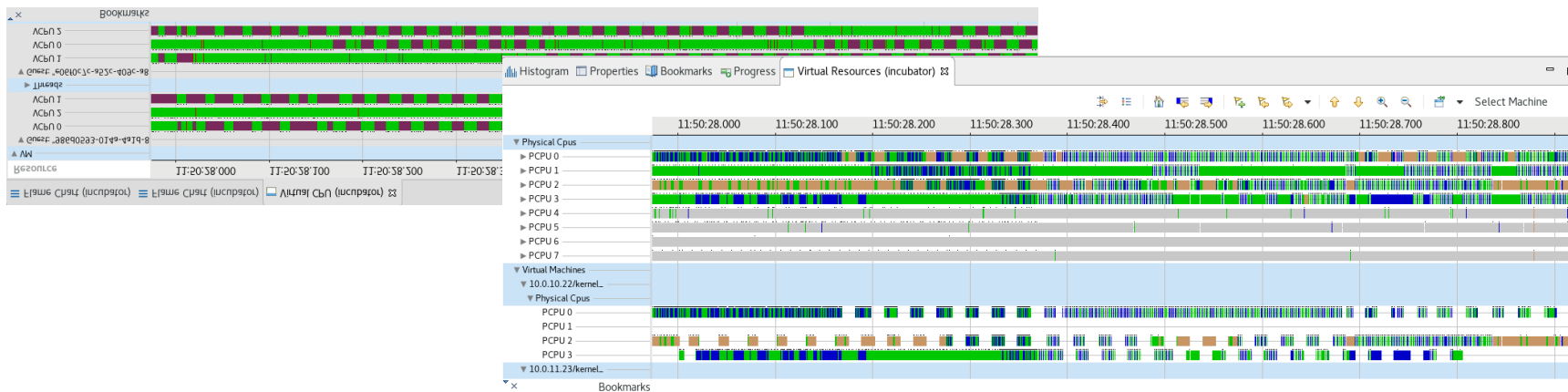


Analyses (frameworks of)

- Generic callstack: data-driven, instrumented or profiled applications, grouped in various ways (not just pid/tid), all kinds of statistics on this data type



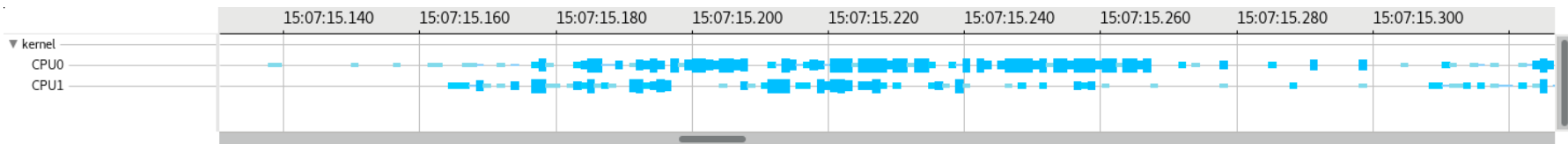
- Virtual machine analysis: Qemu/KVM and containers, tracing host only or host + VMs, various analyses: usage of physical resources, overhead, critical path



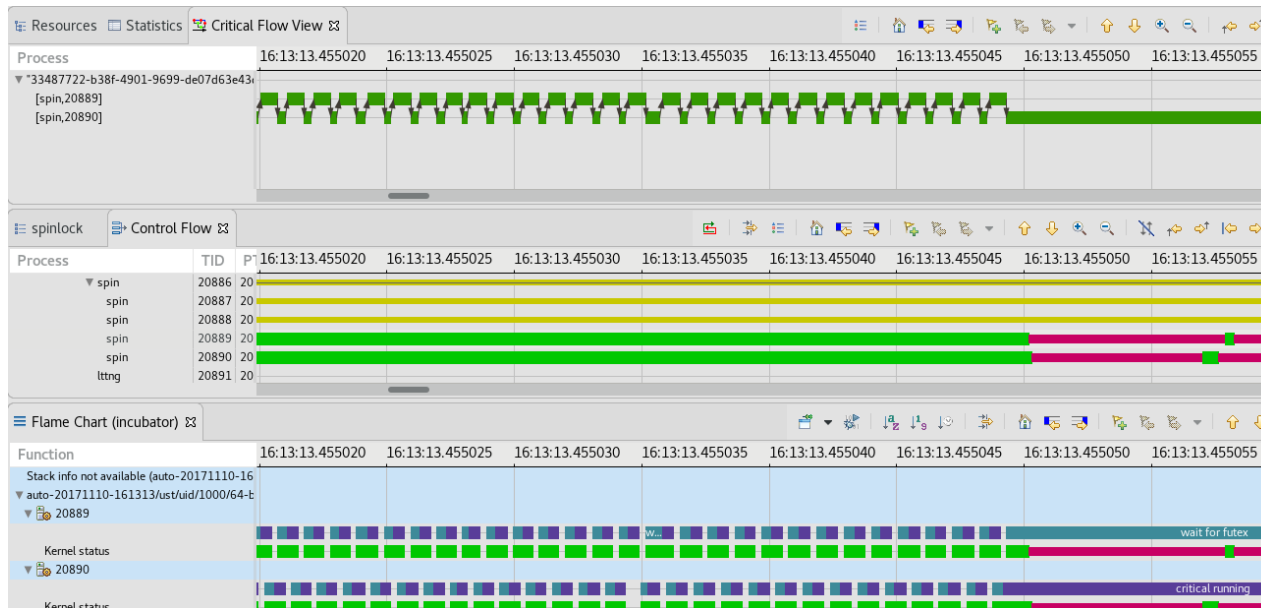


Analyses (frameworks of)

- Context Switch HeatMap: Violon plot for number of context switches in kernel trace



- LTTng-UST extras: Critical path including userspace spin locks





Experimentations

- XaF analysis: Analysis of patterns using real time state machines
- Jersey Rest Server: A rest server for Trace Compass



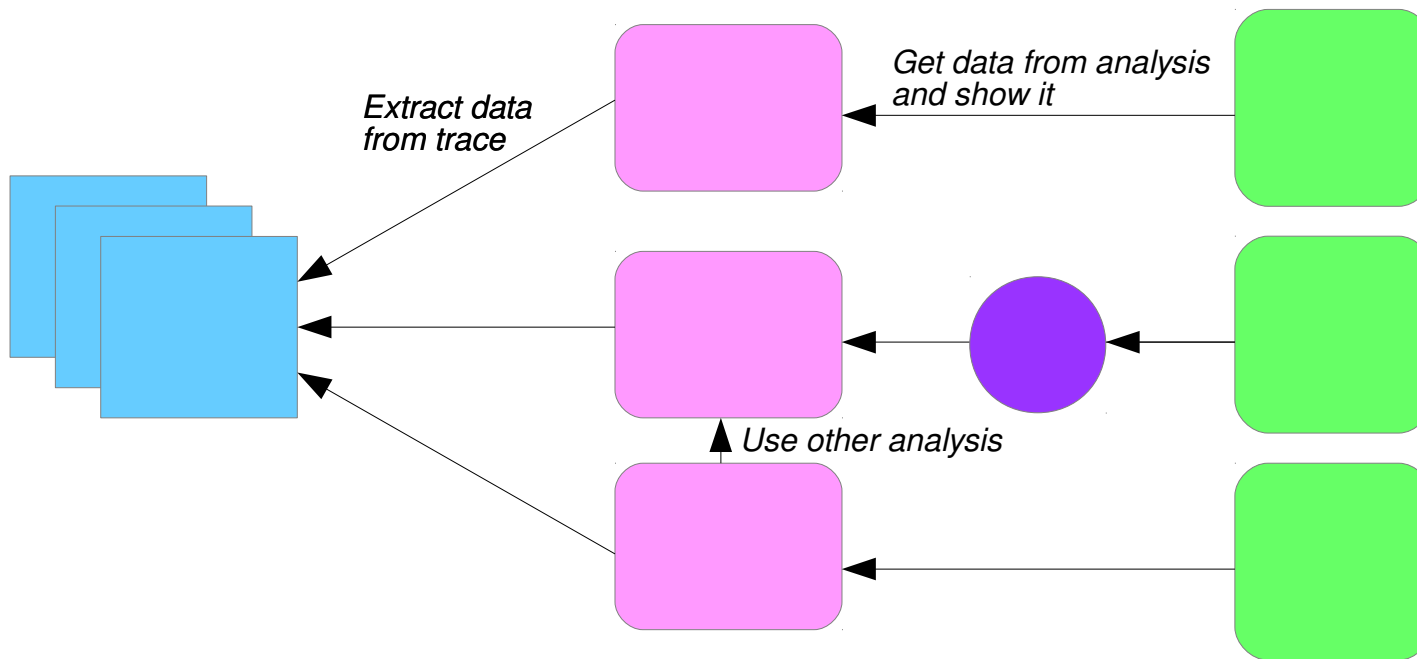
Trace Compass habitual approach

Traces

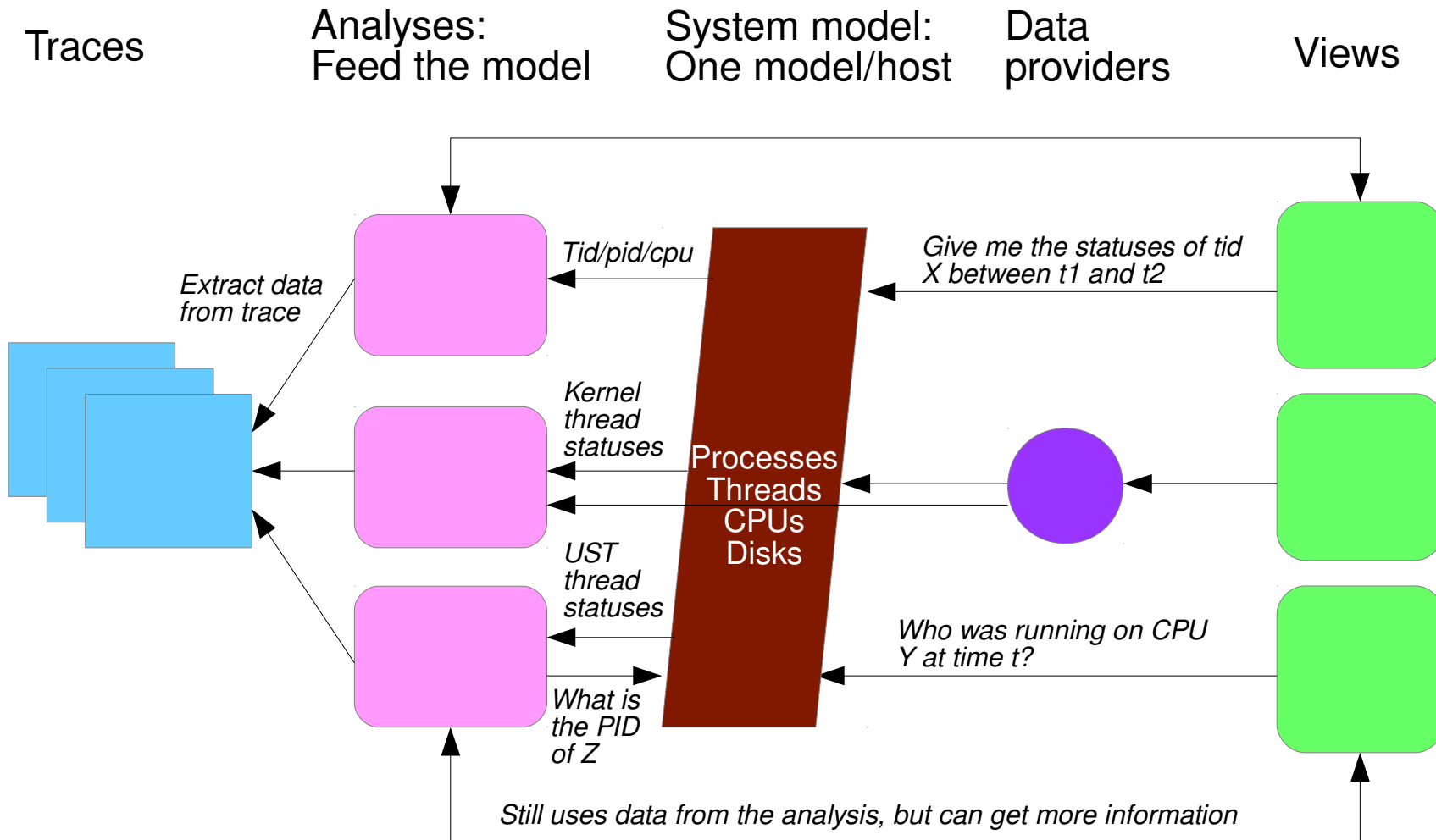
Analysis

Data providers

Views



Incubator approach to trace analysis



Example of the approach:
Generic Callstack



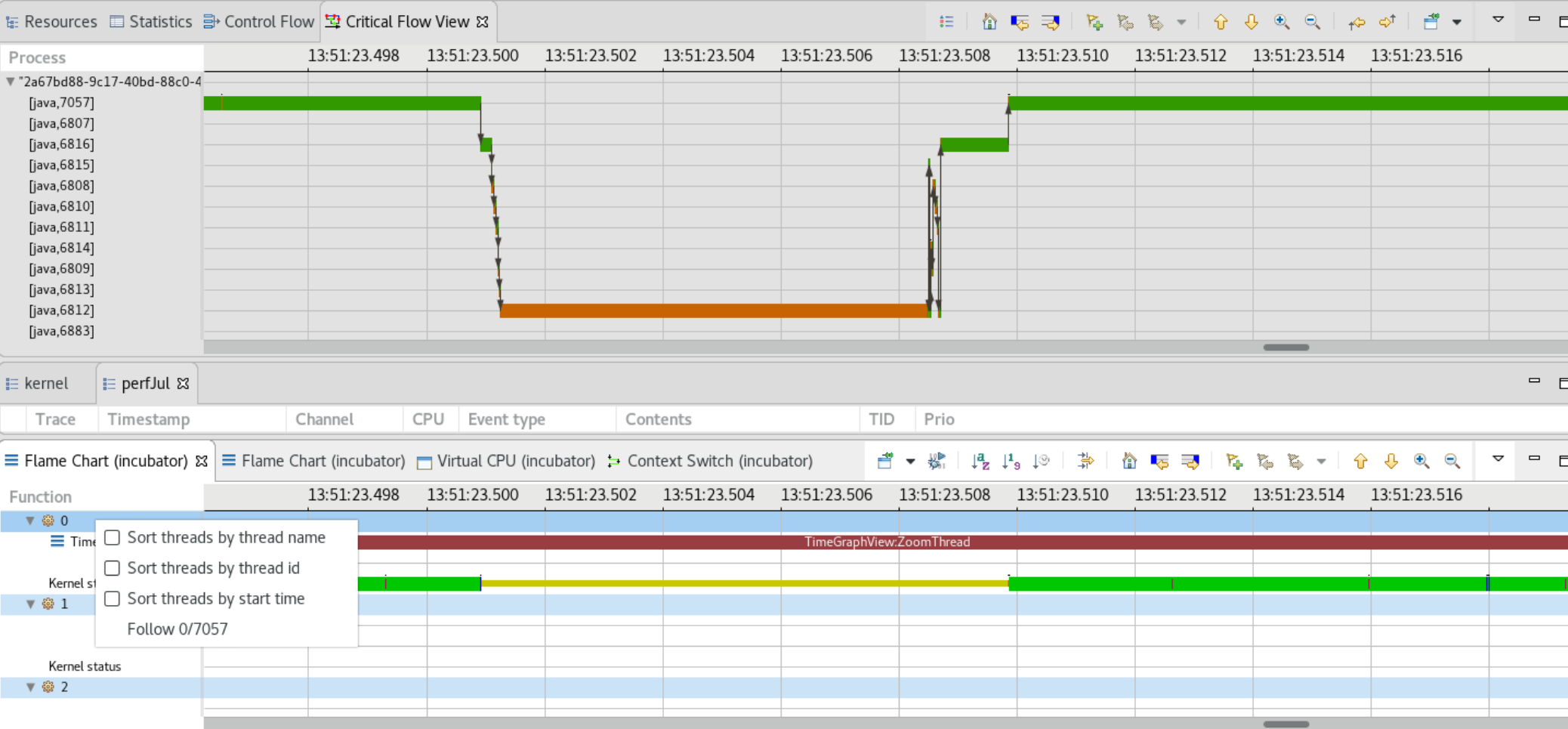
Questions ?

Resources

- Sources: <http://git.eclipse.org/c/tracecompass.incubator/org.eclipse.tracecompass.incubator.git>
- Traces used in this demo: https://secretaire.dorsal.polymtl.ca/~gbastien/traces/jul_perf.tgz
https://secretaire.dorsal.polymtl.ca/~gbastien/traces/vm_ls.tgz
- My personal blog on new tracing features: <http://versatic.net>
- Twitter: @genbastien (Geneviève Bastien) @DavisTurlis (Matthew Khouzam as the name implies...)



Supporting screenshots (Java critical path)



Supporting screenshots (VM overhead)

