Subject: Re: [PATCH 7/7] event: add tracepoint for accounting block time
Posted by Arnaldo Carvalho de M[2] on Mon, 28 Nov 2011 14:31:40 GMT
View Forum Message <> Reply to Message

Em Mon, Nov 28, 2011 at 06:02:27AM -0800, Arjan van de Ven escreveu:
> On 11/28/2011 3:42 AM, Peter Zijlstra wrote:
> > On Mon, 2011-11-28 at 12:03 +0300, Andrew Vagin wrote:
> >> This tracepoint shows how long a task is sleeping in uninterruptible state.

> >> E.g.
> >> It may show how long and where a mutex is waited.

> > Fair enough, makes one wonder how much it would take to make
> > account_scheduler_latency() go away..

> I would *love* to switch latencytop to using trace points / perf events.
> But as long as this just means I get yelled at more for using "internal
> ABIs" and the like at various occasions, I'm rather hesitant to turn
> more tools into using perf.

Have you read the the discussion with Robert Richter about that?

https://lkml.org/lkml/2011/11/24/237

perf_evlist is what you call perf_bundle and perf_evsel is what you call
perf_event in powertop.

That part of the API should be ok for wider use and is in fact exported
in the python binding.

I'm rearranging my tmp.perf/trace4 branch into perf/core to ask for
merging, that is a step in the direction of having the 'perf tool' class
added to what will become libperf.so.

I almost embarked into an attempt to make powertop use it, but there are
other stuff to do first, like making the tracepoint based tools already
in perf stop using long if-strcmp(evname, "foo")-call-process_foo_event
(we have IDs, a hash, better use that, etc).

The 'strace' will be Ingo & tglx's 'trace' tool using these changes,
should be done in a way that shows how to use the resulting abstractions
in libperf.so, that together with the other tools already in the kernel
(kmem, lock, etc).

> (and we all know what the next steps to resolve this are, they just have
> not happened yet; not all hope is lost)

Sure its not :-)

- Arnaldo