Subject: Re: [PATCH v3 2/2] decrement static keys on real destroy time
Posted by Glauber Costa on Thu, 26 Apr 2012 22:28:39 GMT
View Forum Message <> Reply to Message

On 04/26/2012 07:22 PM, Tejun Heo wrote:
> Hello,
>
> On Thu, Apr 26, 2012 at 3:17 PM, Glauber Costa<glommer@parallels.com> wrote:
>>
>>> No, what I mean is that why can't you do about the same mutexed
>>> activated inside static_key API function instead of requiring every
>>> user to worry about the function returning asynchronously.
>>> ie. synchronize inside static_key API instead of in the callers.
>>>
>>
>> Like this?
>
> Yeah, something like that. If keeping the inc operation a single
> atomic op is important for performance or whatever reasons, you can
> play some trick with large negative bias value while activation is
> going on and use atomic_add_return() to determine both whether it's
> the first incrementer and someone else is in the process of
> activating.
>
> Thanks.
>
We need a broader audience for this, but if I understand the interface
right, those functions should not be called in fast paths at all
(contrary to the static_branch tests)

The static_branch tests can be called from irq context, so we can't just
get rid of the atomic op and use the mutex everywhere, we'd have
to live with both.

I will repost this series, with some more people in the CC list.