

Re: [openib-general] Re: [PATCH 0 of 18] ipath driver – for inclusion in 2.6.17

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2006-03/msg08691.html>

- *From:* Roland Dreier <rdreier@xxxxxxxxx>
 - *Date:* Fri, 24 Mar 2006 15:21:09 -0800
-

Roland> I would just get rid of your atomic_clear_mask() and
Roland> atomic_set_mask() calls. They're bogus because you're not
Roland> even operating on an atomic_t, and not many architectures
Roland> implement them.

Bryan> They're not obviously defined to operate on atomic_t
Bryan> objects, but what you say makes sense. I guess that's a
Bryan> peril of using macros for that stuff.

Not on x86_64, but:

```
./asm-s390/atomic.h:static __inline__ void atomic_clear_mask(unsigned long mask, atomic_t *v)  
./asm-sh/atomic.h:static __inline__ void atomic_clear_mask(unsigned int mask, atomic_t *v)  
./asm-sh64/atomic.h:static __inline__ void atomic_clear_mask(unsigned int mask, atomic_t *v)
```

etc. Some other architectures do use unsigned long instead. It all looks rather non-portable, and the only drivers that use these functions are s390-specific. Clearly the simplest solution for your situation is just to kill it.

Bryan> There is no duplicated SMA code. There are two routines in
Bryan> ipathfs that handle nodeinfo and portinfo structures, but
Bryan> they're for passing them to userspace; they don't even
Bryan> really resemble the code in ipath_mad.c.

We seem to be going around and around on this. There definitely is duplicated code; you just hide some of it in userspace. You clearly have two copies of the function to generate a reply to a GET of NodeInfo, for example.

What if you moved the MAD query handling code into your core driver? You could use your current method of sending and receiving replies directly when `ib_ipath` isn't loaded, but just process the queries in the kernel without proxying to userspace. Then if/when QP0 is created, switch to letting the MAD layer handle sending and receiving queries and let it call the same query handling code via your `process_mad` method.

Re: [openib-general] Re: [PATCH 0 of 18] ipath driver – for inclusion in 2.6.17

But it would (I think) solve all the issues of needing `ib_mad` loaded for things to work. Users could even load `ib_ipath` without `ib_mad` and have IB verbs work — anything that actually needed MADs would pull in `ib_mad` as a dependency, and everything else should work fine with the SMA stuff handled by your driver.

Of course this would expose the duplication of the SMI directed route handling logic between your driver and the IB midlayer. Perhaps it makes sense to move this directed route logic into its own module to fix this.

I just really don't like the fact that you apparently have two different MAD handling methods, with different features. Diluting your development by a factor of 2 isn't good for the code. Also, the way you are working around the IB midlayer just smells wrong — either the midlayer or your driver should be fixed so that they work well together.

Bryan> That's true; sorry about that. Do you want me to send you
Bryan> a patch that drops it and pulls it out of headers and
Bryan> kbuild stuff?

Let's just try to get closure on everything and then you can send a real final patchset for merging.

I understand that you really, really want your driver in 2.6.17. But let's do things right. Also, in my opinion, we can still merge `ipath` even after 2.6.17-rc1, since it doesn't touch anything (except trivial kbuild stuff) outside of `drivers/infiniband/hw/ipath`.

– R.

–

To unsubscribe from this list: send the line "unsubscribe linux-kernel" in the body of a message to `majordomo@xxxxxxxxxxxxxxxxx`
More majordomo info at <http://vger.kernel.org/majordomo-info.html>
Please read the FAQ at <http://www.tux.org/lkml/>