**OFI Data Storage / Data Access Subteam Weekly telecom – 07/14/2015**

**DS/DA Shared Documents:** <http://downloads.openfabrics.org/WorkGroups/ofiwg/>

**Agenda**

* roll call, agenda bashing
* kfabric intro slidedeck, Again
* user mode I/O
* NVM usage models

**News**

HotI paper was submitted last night, Sean planning to present.

**slide deck – kfabric-framework\_2015\_0707a.pptx**

- Resuming the discussion beginning with the last bullet on slide 5:

* Smooth transition path from existing kernel verbs

- future proofs the kernel fabric stack (ibverbs) with a fabric independent framework

- Decided at the last meeting to proactively address the question of ‘why not verbs?’

- Looking at new slide, “why not extend ibverbs?”

- Bernard – may want to address the idea for NVW that you really don’t want the overhead of standard connection management. The requirement for connection management is, again, largely an artifact of the QP based approach taken by IB.

- Added a new slide, “Why kfi for NVM?” as a means of directly addressing NVM as a consumer of kfi.

- a popular mechanism for achieving HA for storage on the midplane is the use of non-transparent bridging over PCIe.

- Worked through some updates to slide 8 (kfi framework).

- Discussed whether or not we should include the canonical OFI architecture diagram, or an abstracted version of it to support the existing slide 9.

- Agreed that there is a gap in the slides w.r.t. repo directory structure.

- This immediately begs the question as to whether we are talking to the correct kernel maintainer. Raises questions about where kfi belongs in the stack – is it part of the RDMA stack (which lives under drivers, InfiniBand), or is it part of the network stack? The key is that kfi is really more abstract than the classical driver, i.e. kfi isn’t just another driver.

- Because IB is QP-based, if you want to fit into the existing Drivers/IB stack you really end up emulating a QP-based architecture.

- The real objective here is to promote RDMA up a couple of levels of abstraction from the current Drivers/IB stack. In the existing world, IB is an implementation of RDMA.

- This is really the nut of the matter.

- We’re really trying to create a new paradigm, somewhat like sockets.

**Agenda for next meeting**

- final review of the slide deck (current version: kfabric-framework\_2015\_0707a.pptx)

**Webex Recording:** [**Play recording**](https://cisco.webex.com/ciscosales/ldr.php?RCID=b27c4c6853e43fda3e1d51371898f5e3)

**Next regular telecom**

Next meeting: Tuesday, 07/21/15

8am-9am Pacific daylight time

**NOTE:** We have switched over to using Webex (courtesy of Cisco). The URL for joining meetings is:

<https://cisco.webex.com/ciscosales/j.php?MTID=m68f7fe26d65ee019c5870bc424875838>

**Join by phone**

+1-866-432-9903 Call-in toll-free number (US/Canada)

+1-408-525-6800 Call-in toll number (US/Canada)

Access code: 205 894 276