**OFI Data Storage / Data Access Subteam Weekly telecom – 09/29/2015**

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

**Agenda**

* roll call, agenda bashing
* NVM usage models – Chet Douglas

**NVDIMM usage models – Chet Douglas part 2, see Chet’s slides**

**“RDMA with PMEM, Software Mechanisms for Enabling Access to Remote Persistent Memory”**

- Problem statement – current RDMA architectures don’t take persistent memory into account.

- This slide deck mostly focused on byte addressable memory, mainly focused on user mode.

- may not be relevant to our immediate interest, which is focused on kernel mode NVM.

- Visibility vs Durability: In the Intel case, data is normally ‘visible’ when it hits the L3 cache, but is not durable until it gets to NVMedia.

- Flushing data to local NVM: uses two new Intel instructions, CLFLUSH (flush w/ invalidate), CLFLUSHOPT (flushes per cache line with invalidate), CLWB (flushes but doesn’t cache invalidate). Requires an explicit PCOMMIT to force the data into NVDIMM.

- Remote data access – much the same. A remote initiator transfers data to the local (target) node; the local node still needs to execute one of the new instructions to flush the data from cache into the memory domain.

- “Appliance Method” - Remote (initiator) forces a flush to memory by executing a write/write/write…followed by a read. The read forces a PCIe flush to the memory controller. Data does not transit through L3 cache.

- “General purpose Server Method” - consists of a series of RDMA writes to the target, which caches them in L3. Eventually the initiator executes a SEND to the target, which wakes it up and causes it to flush the cache using one of the new flush commands. Requires a second handshake from the target back to the initiator, signaling completion.

- All the above mechanisms are acknowledged as being non-optimal; they are designed to be implementable using today’s hardware through s/w changes only.

- There are hardware mechanisms coming down the pike to support the forcing of commits to NVDIMM without requiring any sort of explicit additional handshake between initiator and target.

Chet also some further thoughts that are specific to kernel mode. We can discuss these next week. He will re-send an email listing these topics.

**Agenda for next meeting**

- Further thoughts specific to kernel mode from Chet.

- Focus on kernel use cases for NVM for the Linux Kernel Maintainers Slide Deck

**Webex Recording:** [**Play recording**](https://cisco.webex.com/ciscosales/ldr.php?RCID=342cca83199bbc7b37a1e42e0382d8f8)

**Next regular telecom:**

Next meeting: Tuesday, 10/6/15

8am-9am Pacific daylight time

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

[Join WebEx meeting](https://cisco.webex.com/ciscosales/j.php?MTID=m221d8a20185d84b30daa0096aca0f182)

**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: 201 212 241