**OFI WG Data Storage / Data Access Subteam Weekly telecom – 11/04/2014**

**OFIWG Download Site:** [www.openfabrics.org](http://www.openfabrics.org) 🡪OFED/OFA Resources 🡪 OpenFabrics Interfaces WG

**Agenda**

* role call, agenda bashing
* Chet Douglas – some challenges being faced by Intel (and other processor vendors) in making data persistent.
* Next steps

**Processor challenges in making data persistent – Chet Douglas**

**see the slide deck “RDMA with byte-addressable PM”**

**-** three fundamental elements of current processor I/O architecture:

 - ADR (Asynchronous DRAM Refresh): the ADR domain includes main memory and an integrated memory controller. Once inside the ‘adr domain’ data is headed toward NVM. ADR is the set of super caps and other hardware to assure that data, once inside the ADR domain, is considered persistent.

 - IIO – Integrated IO Controller – controls IO flow between PCIe devices and main memory (e.g. ADR domain). Used to be in server southbridge. Important point: it contains internal buffers; does some amount of ‘caching’ but separate from memory cache.

 -DDIO – Data Direct IO – Can move data from the IIO directly into processor cache completely avoiding main memory.

- ‘Short term’ NVM considerations: using ADR, but with DDIO turned off

 - assume that ADR is being used and contains NVM, (and DDIO is turned off), thus the IIO pushes data directly to the integrated memory controller (iMC). Once inside the ‘ADR domain’, the data is in the control of the iMC (memory controller) and is considered non-volatile. But data may be stuck in the IIO’s internal buffers, therefore must use some sort of operation (e.g. SEND/RECEIVE, RDMA READ) to synchronize data between the IIO and ADR domains. For now, the requester has to do a SEND/RECEIVE or RDMA READ operation to force PCIe synchronization, but in the future this could be done automatically by the responder side RNIC.

- ‘Short term’ NVM considerations: without using ADR, without DDIO

 - same sort of problem as previously, but in this case pushing data into the ADR domain (the memory controller) isn’t sufficient to ensure the data is non-volatile, you have to force the write to NVM to occur. Must use either a SEND/RECEIVE (or potentially an RDMA WRITE w/ immed) to force the write to memory to occur. One issue with using RDMA WRITE w/immed is that iWARP doesn’t currently support RDMA WRITE w/immed

 - new Intel ISA command: PCOMMIT/SFENCE flushes iMC and makes data persistent in NVM.

- ‘Short term’ NVM considerations: without ADR, but with DDIO (thus bypassing non-volatile memory)

 - using DDIO pushes data directly into cache, but it is not in NVM and thus not non-volatile.

 - new Intel instructions (CLFLUSHOPT/SFENCE) flushes processor caches to NVM, in connection with new PCOMMIT/SFENCE operation.

- Long Term NVM Considerations

 - ADR HW – increase the diameter of ADR Domain to include both LLC and IIO internal buffers, thus all I/O data would be in the power-safe domain.

 - IIO HW – Make HW aware of persistent memory ranges by making the synchronization operation automatic to force synchronization between the IIO and the ADR domain without requiring a following SEND/RECEIVE to force synchronization.

 - DDIO HW – same sort of approach, enable DDIO for DRAM forcing persistent memory transactions on the fly.

**Next Steps**

How do we integrate this into the libfabrics API?

- Chet’s presentation sensitizes us to the fact that the data pipeline may not be included in the NVM domain; it’s up to us to decide how this should be reflected as a requirement. Is this some sort of a new operation type?

- Who needs to be aware of NVM? The responder side? The requester side at the provider layer? The application at the requester side?

For next meeting we need a summary of requirements gathered to date. We will plan to have a meeting next week either to discuss the summary if its available, or to address some other topic that may arise during the week. If not, we may end up cancelling next week’s meeting.

**Agenda for next meeting**

Summary of Requirements gathered to date

**Next regular telecom**

Next meeting: Tuesday, 12/9/14, or possibly 12/16/14.

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/cisco/j.php?J=200935598&PW=67935ad6df07030d5f05044a5b0f>