**OFI WG Bi-Weekly telecom – 08/01/2017**

**Agenda:**

* Update on current release schedule on rev 1.5
* BoF update for SC17
* FI\_FENCE issue
* Persistent Memory - NVML

**Update on current release 1.5 schedule**

* May not need an RC3. Hoping for Rel 1.5 end of this week.

**SC17 BoF**

* Chris Taylor’s proposal for a BoF discussing the integration of libfabric into C++ has been submitted.
* The libfabric proposal will be submitted later this week. Comments on the proposal are still welcome.
* Jeff S. volunteers to sign up as a BoF leader as long as it doesn’t conflict with the OpenMPI BoF (Wednesday at noon).

**FI-FENCE**

* New issue opened. Current definition is not a good match for applications.
* Current semantics – the operation after the fence doesn’t begin until all operations to the same peer have completed. May want to change the semantic so the next operation doesn’t begin until operations to all peers from a given endpoint have completed.
* Sean to create a few slides describing the issue.

**PM over Fabrics – NVML introduction – Chet Douglas (Intel)**

* ‘non-allocating’ means that data is written direct to the persistence domain, bypassing L3 cache
* DDIO means a direct write to L3 (as opposed to non-allocating).
* Reqmt: include persistence as part of memory registration, such that the NIC will know if inbound data is bound for a persistence domain or not. Today, it’s on a per RNIC basis…all writes to the same RNIC are handled the same. Need to discuss/debate the granularity of this.
* CLWB + fence / CLFLUSHOPT + fence / CLFLUSH / NR store + fence / + WBINVD (kernel only) are all CPU specific cache flushing mechanisms and are all abstracted by NVML.
* NVML is user space, open source <http://pmem.io/> librpmem, librmemd are the relevant libraries.
* Use case – data replication: Application opens a file and does R/Ws to the file, then does ‘make persist’ (addr, len) to make a certain range persistent. This same region is then replicated to remote memory
* NVML began as a solution around local NVDIMM. Its main use case is for replication of data between the local and remote nodes.
* **Agreed to continue the discussion during a special off-cadence meeting next week – 8/8/17.**
* **LOGISTICS WILL BE POSTED TO THE OFA CALENDAR:** <https://openfabrics.org/index.php/ofa-calendar.html>
* **Please check the calendar.**

**Recording:**

|  |  |  |
| --- | --- | --- |
| **OFIWG/libfabric meeting-20170801 1608-1** | | |
| Tuesday, August 1, 2017 | | |
| 12:08 pm  |  Eastern Daylight Time (New York, GMT-04:00) | | |
| [**Play recording**](https://cisco.webex.com/ciscosales/lsr.php?RCID=2b980800b3a34440b65b1e232625efcf) (54 min 47 sec) | |
| Recording password: UvzWRwE7 |

|  |
| --- |
|  |

|  |
| --- |
|  |

**Webex link:** See the OFA central calendar for meeting logistics. <https://openfabrics.org/index.php/ofa-calendar.html>

**OFIWG Download Site:** [www.openfabrics.org/downloads/OFIWG](http://www.openfabrics.org/downloads/OFIWG)

**Github:** <https://github.com/ofiwg/libfabric>

**OFI Software Download Site:** [www.openfabrics.org/downloads/OFI](http://www.openfabrics.org/downloads/OFIWG)

**Next regular telecon**

Next meeting: Tuesday, 8/15/17

9am – 10am Pacific daylight time