**OFI WG telecon – 04/23/2019**

**Agenda:**

1. Opens, Agenda bashing
2. Remote Persistent Memory
3. Begin discussion of Traffic Classification
4. MR cache - Raghu

**Opens**

**Discussion – Remote Persistent Memory for HA Whitepaper – Doug Voigt**

Comment period on PM for HA document v1.8 is now closed.

Next will be any required edits, followed by a SNIA approval process.

**MR cache – Raghu**

If a core provider uses a memory cache, but does not use bounce buffers, how does a utility provider know that there is a cache below it? Does it make sense to add a set\_opt or get\_opt flag to allow the consumer to find this?

Q: Why are mode bits not sufficient?

A: There doesn’t appear to be a specific combination of bits that covers that.

Q: Why not MR\_LOCAL?

A: It’s possible.

RXD will almost always use bounce buffers, but RXM using rendezevous protocol, may be a different story. As of today, RXM uses bounce buffers.

**Discussion – Traffic Classification – Paul Grun**

High level proposal overview - A scheme that:

* Includes the usual hints mechanisms to allow a consumer to request specific services and the provider to respond with its capabilities
* Allows HPC-specific labels to be transmitted across the API, or
* Allows specific DSCP values to be transmitted across the API (but not both) to meet the meets of classical IP-based applications
* Proposal includes five labels with accompanying text describing the usage for the five labels.

Q: Do we need a label dedicated to fabric management? We should look at adding one.

Q: Is a label needed specifically for checkpoints?

Could use the existing Bulk Data class or Dedicated Access class, but there might be value in specifically calling out checkpoint, even if it aliases directly to one of the above.

Q: In the absence of a mechanism for rapidly changing the traffic class for a given endpoint, this scheme implies that an application that requires multiple traffic classes will require more resources to implement each needed traffic class. The alternative is to build a more complex mechanism that anticipates rapid re-programming the traffic class for an endpoint.

A: True, but it seems likely that different traffic will be targeting different endpoints anyway. E.g. checkpoints using Bulk Data are going to a different set of endpoints than standard MPI inter-rank traffic. The sense of the body is that the current approach is correct, while acknowledging that more complex schemes are possible.

Q: Should the calls used to control traffic class (presumably FI\_SETOPT/FI\_GETOPT) be synchronous or asynchronous?

**Next meeting**

Tuesday, May 4, 2019

9:00 – 10:00AM PST

**Recording:**

**-none-**

**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)