**OFI WG Bi-Weekly telecom – 01/26/2016**

**Agenda:**

* Roll call, agenda bashing
* SC16 Workshop
* Support for other OS’ – see Sean’s email “ofiwg item: supporting other OS’s” (<http://lists.openfabrics.org/pipermail/ofiwg/2016-January/000996.html>)
* Completion optimizations – see Sean’s email “completion optimizations” (<http://lists.openfabrics.org/pipermail/ofiwg/2016-January/000997.html>)
* Tentative release dates
* Wait set /poll set / cq\_sread discussion

**SC16 Call for Workshops**

* Take a pass

**Tentative Release Dates**

* Only thing being suggested is to forego a Q3 quarterly release, and bring the Q4 release in a little bit. This is because we are driving toward a larger scope of changes in Q4, hence eliminate the Q3 release in favor of giving ourselves more time for testing and so on.
* The API extensions would be included in a 1/5 release, since we are speculating that we will be able to maintain ABI compatibility.

**Support for other OS’**

* A request for alternate OS support had come from the national labs
* Primarily Solaris (x86 & sparc) for the MPICH project
* Testing on Sparc is difficult because of access to Sparc hardware. The question is whether access to hardware can be achieved. Ken to take this request to ANL to see what happens.
* Sean has contacted Intel MPI team. Supporting Solaris is not too difficult, really just a testing question, but Windows may impact the structure of the code, and the build process, packaging, etc. Waiting for further input as to what changes to the code base may be required.
* Oracle makes Solaris available (on x86) for testing purposes. It still boils down to testing on Sparc hardware.
* Conclusion – best effort support for Solaris for now; ANL to try to help with Sparc hardware for testing. Re Windows, waiting for further input from the Intel MPI team.

**(from two weeks ago, left here for context)**

* Questions have arisen in the past about support for other OS’, Solaris, Windows…
* There is a specific request for windows, the Solaris request may be coming from MPI
	+ Not clear if the Solaris request is for x86, or Power, or…?
* Provider ports to another platform would be a function of the vendor, except for sockets.
	+ Not clear if e.g. the sockets provider would be a combined Linux/xxx provider, or two separate providers
* Tool chain issues – do you force the use of Cygwin, or maintain parallel tool trees? Mainly an issue for Windows, Solaris should be straightforward.
* Think of networkdirect as a provider underneath libfabric.
* May need some changes to libfabric for e.g. wait objects, loader sequence, etc.
* An alternative would be to create a separate fork.
* Sean to discuss with Intel MPI term to better understand their perspective and plans for long term support. If we pursue it, there needs to be somebody willing to own it.
	+ Same questions for Solaris – would this be for x86? X86-64? Sparc?

**(unfortunately, we failed to turn on the webex recording until this point)**

**Wait set / poll set / cq\_sread – Dave Goodell**

* Cisco has an internal customer, but the requirements aren’t yet well understood.
* Verbs is reasonably well understood, but we don’t seem to have the same level of clarity for libfabric.
* But libfabric is intended to be more abstract since the underlying implementation isn’t well understood.
* GitHub issue #1645 –
* Once a wait has been signaled, when does it go away?
* When you create a wait set, and call wait, and it’s signaled, how does the application go back to wait again?
* How to handle multiple CQs on the same wait object?
	+ Possibly use a poll set to see which CQs have a completion on them.
* What is the use case for a wait with a single CQ? You would instead block on cq\_sread.
* What is the use case for wait sets? Not especially useful for a single CQ, and perhaps better covered by poll sets for multiple CQs. But a wait set may make it easier to combine CQs and EQs. Is it worth it? Consider eliminating wait sets…? Merge poll sets and wait sets somehow?
* Who is using wait sets today?
	+ MPICH CH4 possibly
* No real conclusions today, Cisco to continue thinking about ways to improve it.
* Find/document a fix without (or minimal) changes to the API, with an eye toward an ideal fix toward the end of the year.

**(from two weeks ago – no updates today, awaiting a formal proposal from Sean)**

**Completion Optimizations – Sean’s slide fi-v2.pptx**

* Four different completion topics
* Relaxing support for completion flags.
	+ Currently, a lot of information is given (is it RMA? Is it Read, or Write). For some providers, particularly on error, it is difficult to provide this level of detail.
	+ Proposal: FI\_NOTIFY\_FLAGS\_ONLY mode bit. Only the following flags are used: FI\_REMOTE\_READ | FI\_RMA, FI\_REMOTE\_WRITE | FI\_RMA, FI\_REMOTE\_CQ\_DATA, FI\_MUTLI\_RCV…
	+ This bits would be sourced by the provider to indicate to the consumer that it can provide only a certain amount of completion information.
* Report Optimal CQ Attributes
	+ For example, what is the optimal size for the CQ? No way today for an application to know what kind of CQ to use, no way to know what kind of structures the provider can deliver optimally.
	+ Proposal: provide a linkage into the fi\_getinfo structure
	+ Add fi\_cq\_attr to fi\_info (or fi\_tx/rx\_attr
	+ This would require a change to the API which would require us to use a library versioning mechanism.
* Restrict EP and CQ Bindings (also about optimizing CQs)
	+ Currently no way to restrict which Endpoints use which CQs.
	+ Proposal: Add capability and mode bit fields into fi\_cq\_attr, saying that this CQ will only support a particular set of capabilities.
* Do we start moving on these types of changes, or wait to see if other issues appear that need to be addressed? In other words, the frequency of changes to the API. The issue really boils down to how pressing these issues are to the users of the API.
* Sean to try to work down into a bit more detail. Put on next agenda to check-in, but will probably take a little longer to get somewhere. The provider developers should also try to think through some of the implications.

**Next Agenda:**

* Revisit the possibility of an SC16 workshop
* Continue discussion of support for other OS’
* Check in on Completion Optimizations thinking

**Webex link:** <https://cisco.webex.com/ciscosales/j.php?MTID=m9389b0513c9ae643d57e2381e254dcf5>
Webex password: ofi

**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)

**Link to WebEx Recording** - [**Play recording**](https://cisco.webex.com/ciscosales/lsr.php?RCID=0f0fc92c4d944d0491929d75e3cb5b2c)

**Next regular telecon**

Next meeting: Tuesday, 1/26/16

9am-10am Pacific daylight time