# HP-MPI on Infiniband

www.hp.com/go/mpi

Changqing Tang

May 24, 2009
Sonoma, CA

# Agenda

- 1. HP-MPI on HP-UX
- 2. HP-MPI on Linux.
  - HP-MPI for HPC Application
  - HP-MPI for Commercial Application
- 3. HP-MPI on Windows
- 4. HP-MPI Requirements for OFED
- 5. Q/A and discussion

# HP-MPI on HP-UX

- Proprietary stack developed by HP
- IT-API interface
- Same IB HCA, Cable, Switch as Linux
- Kernel memory registration caching

# HP-MPI on Linux

- Support OFED either via IB verbs interface, or uDAPL interface, PSM if on Qlogic system.
- Either RDMA, send/recv or SRQ mode, but not mixed
- Memory registration caching via ptmalloc3 library
- Only using RC, XRC

# HP-MPI Linux IB Features

- Support port failover, card failover
- Support APM
- Support detecting broken connection
- Support dynamic processes
  spawn/connect/accept
- Support singleton MPI
- Support mixed rdma-write and rdma-read
- Support atomic in one-sided sync operation

# HP-MPI for HPC Application

- Scale as large as 14000 ranks using XRC
- iWARP using uDAPL protocol
- Dynamic rdma buffer management for SRQ
- Using multi-rail for improved bandwidth
- Using port failover for HA
- Using rdma-read for async MPI communication (ENZO)
- Using dynamic processes for dynamic apps.
- Using one-sided operation as needed.
- Using large data transfer (>2G) support in apps.

# HP-MPI for Commercial Application

- Internal project
- Using singleton processes startup
- Using broken connection detection
- Need port failover for interconnect HA
- MPI HA – recover from connection failure – drop the connection
- MPI HA – recover from rank failure – isolate the rank
- Low MPI CPU overhead on heavily oversubscribed system, 1000 processes per node (8 cores)
- Support multi-thread MPI library
- Welcome to contact HP-MPI if you want to have MPI based commercial application.

# HP-MPI on Windows

- Using IBAL interface
- HPC 2008
- Support the same functionalities as Linux
  - Dynamic process, multi-cards, port failover, rdma/send-recv/srq, rdma-write/read, …

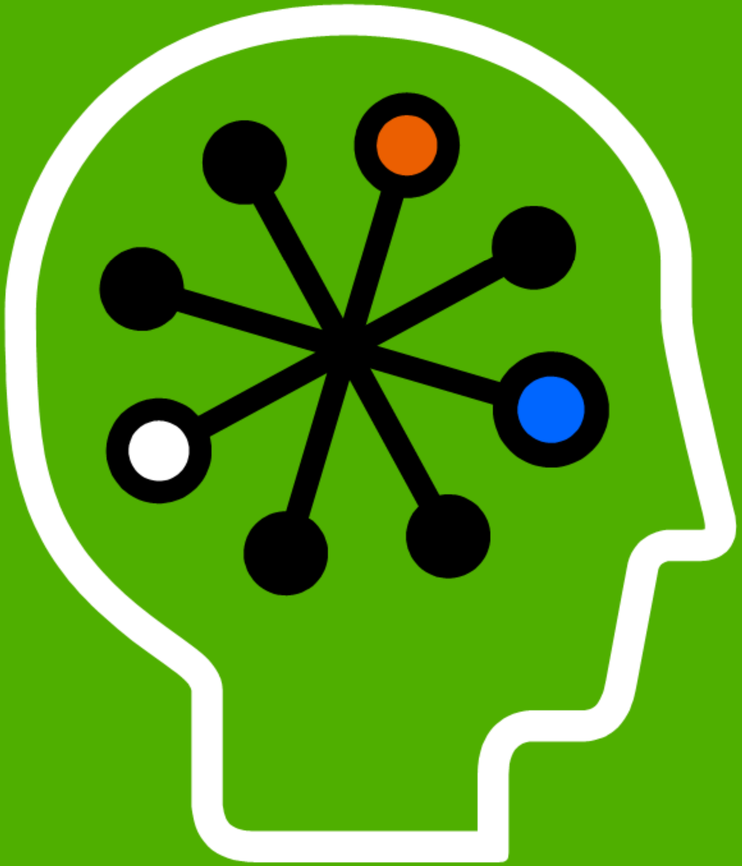
# HP-MPI Suggestions for OFED

- 1. Fork()/Exec() support:
  - Child does COW, as if no registration happened.
  - Parent copies the page, and assign to child. Parent keeps the old page
  - Detecting broken connection: after fork()/exec(), close device fd.
- 2. SRQ peer identification
  - Create SRQ
  - Create QP with (srq, context)
  - Completion event returns context if message is from that QP
- 3. Memory register: slow and complicate code.
  - hp-mpi uses ptmalloc3 for caching, still problem.
  - Either remove memory registration, or improve performance
- 4. Simplify LMC: let driver manage it.
- 5. APM – waste resource,
  - Can driver use active-active mode ?
- 6. RD ?
  - It is in specification.

# More Information

- www.hp.com/go/mpi
- Questions ?

# Back-up



Technology for better business outcomes