Hewlett Packard
Enterprise

Math in Your Network: Slingshot Hardware Accelerated Reductions

Forest Godfrey and Duncan Roweth

CUG 2025
May 6, 2025

About the Authors

Forest Godfrey

Distinguished Technologist

Slingshot Architecture

- 26 years of experience in High Performance Computing at SGI, Cray Inc., and HPE
- Have worked on various Cray architectures including the X1/X1E vector architecture and the Gemini and Aries networks
- Working on Slingshot since 2017 in various capacities
- Current tech lead for the software side of Slingshot Collectives

Duncan Roweth

HPE Fellow

Slingshot Architecture

- 40 years experience in HPC at Inmos, Meiko, Quadrics, Cray, and HPE
- Working on Slingshot since it started in 2015
- Prior to that, worked on QsNet at Quadrics and Aries at Cray
- Currently Chief Architect for Slingshot, defining our next generation products

Agenda

- Brief Introduction to Collectives and Reductions
- Software Implementation of Collectives in Slingshot
- Hardware Implementation of Collectives in Slingshot
- Interactions With Fabric Manager and Job Scheduler
- Schedule and Preliminary Results
- Questions

Agenda

Brief Introduction to Collectives and Reductions

Software Implementation of Collectives in Slingshot

Hardware Implementation of Collectives in Slingshot

Interactions With Fabric Manager and Job Scheduler

Schedule and Preliminary Results

Questions

What are Reductions and Collectives

- Reductions and collectives are common elements of the Bulk Synchronous Parallel (BSP) programming paradigm which is common in HPC
- Reduction and Collective Operations
  - Reductions take a large set of data and ”reduce” it by performing an operation
    - Example: Global sum, Global Min/Max
  - Collectives are operations where all nodes (“the collective”) participate effectively as peers (though some nodes may function as aggregators)
    - Example: Barrier, All-to-All, Gather/Scatter or Broadcast
- Are useful in many programming models, but this presentation focusses on MPI
- Collectives and Reductions can be performed on all ranks within a job or on a subset
  - Subsets are implemented with sub-communicators in MPI
- Typically implemented in software
  - Variety of algorithms for doing so
  - (Arguably) Simplest is a tree where all ranks contribute up a tree to a root, which then distributes the result
    - Distribution is called “Broadcasting”

- This presentation is not meant as an in-depth exploration of collectives/reductions
  - Useful to have an overview

Performance Requirements in Collectives and Reductions

- Operations are synchronization points in applications
- In many operations, each rank must complete the operation before any rank may advance past it
  - The application is only as fast as the slowest node
  - All time spent waiting for collectives to complete is time not spent doing computation
- Collective performance requirements are application dependent
  - Prototypical example - strong scaling in HPC applications
    - A fixed task is distributed over more and more nodes
    - Work is divided up between more and more CPUs/GPUs
    - Collective (e.g. a convergence check) takes more and more time
  - Example applications that are sensitive to collective performance
    - Pennant, LANL hydrodynamics app

Agenda

Brief Introduction to Collectives and Reductions

Software Implementation of Collectives in Slingshot

Hardware Implementation of Collectives in Slingshot

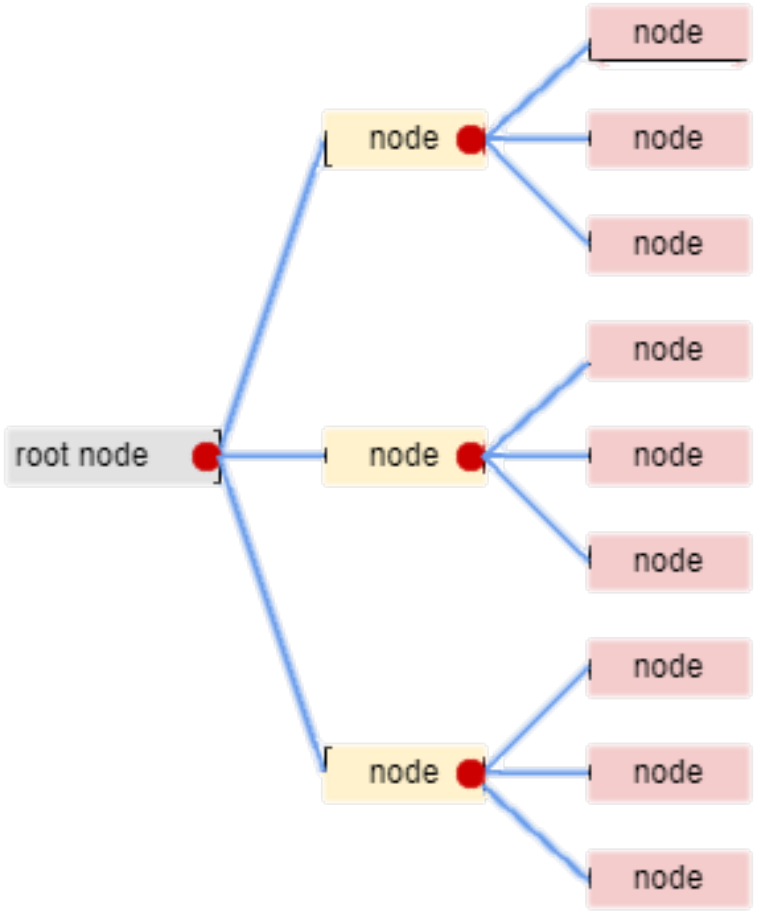
Interactions With Fabric Manager and Job Scheduler

Schedule and Preliminary Results

Questions

Software Implementation of Collectives in Slingshot

- Implemented in libfabric
  - MPI calls libfabric for cases supported in hardware
- Tree algorithm
  - Upstream traffic partially-collected at intermediate compute nodes which contribute data, perform any necessary computation, and send upstream
  - Traffic is collected upstream at a root node, which is one of the job ranks
  - Downstream flows through intermediate NIC that "fan out" the traffic.



Agenda

Brief Introduction to Collectives and Reductions

Software Implementation of Collectives in Slingshot

Hardware Implementation of Collectives in Slingshot

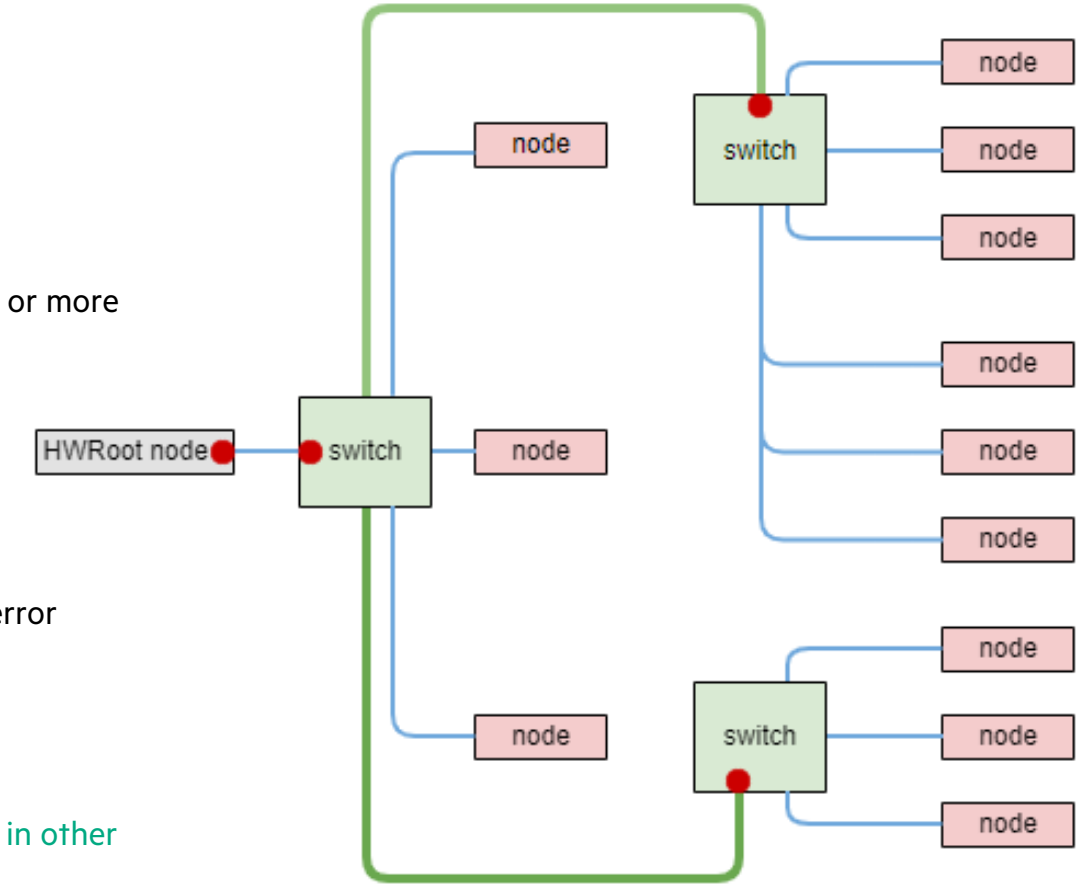
Interactions With Fabric Manager and Job Scheduler

Schedule and Preliminary Results

Questions

Slingshot Hardware Collectives Acceleration

- Collective/Reduction engine hardware exists inside Rosetta
  - Includes IEEE floating point pipeline for reduction operations
- Uses the same area of hardware as Ethernet multicast/broadcast
  - Requires allocation of special multicast address to function
    - More details in the next section
- Limited resource but sufficient for most practical uses
  - Likely to see a benefit to hardware acceleration with larger jobs, e.g. 64 or more nodes
- Collectives performed on tree of switches
  - Eliminates traffic in and out of intermediate compute nodes
  - Eliminates compute nodes performing intermediate reductions and minimizes compute on root node
  - Handles late arrival and resource exhaustion, stateless in the event of error
  - Handles up to 8×4×8-byte payload sizes
- Hardware root node is one of the normal job ranks
- Software interface implemented in libfabric
  - MPI calls libfabric for these cases – uses optimized software algorithms in other cases



Collective & Reduction Operations Accelerated by Hardware

Libfabric opcode	Operation
COLL_OPCODE_BARRIER	Barrier
COLL_OPCODE_BIT_{AND,OR,XOR}	Bitwise operations
COLL_OPCODE_LOG_{AND,OR,XOR}	Logical operations
COLL_OPCODE_INT_{MIN,MAX}	Integer min /max
COLL_OPCODE_INT_MINMAXLOC	Integer min /max with location
COLL_OPCODE_INT_SUM	Integer sum
COLL_OPCODE_FLT_{MINNUM,MAXNUM}	Floating point min/max
COLL_OPCODE_FLT_MINMAXNUMLOC	Floating point min/max with location
COLL_OPCODE_FLT_SUM_NOFTZ_RND{0,1,2,3}	Floating point sum with
COLL_OPCODE_FLT_SUM_FTZ_RND{0,1,2,3}	
COLL_OPCODE_FLT_REPSUM	Reproducible floating-point sum
COLL_OPCODE_MAX	

Software Use/Management of Hardware Accelerated Collectives

- **Translating Hardware Collectives To MPI**
  - MPI collectives are called using communicators (MPI_COMM_WORLD, or subset of MPI ranks)
  - Each communicator requires a hardware collective tree to be created on the fabric
  - Each collective tree requires a single multicast Ethernet address since the collective hardware is implemented using the Ethernet multicast hardware in Rosetta
  - Max of **8192/16384** multicast addresses available on any system (regardless of size) in Rosetta-1/Rosetta-2
    - Multicast address space is shared by other network features (like VLANs)
    - Today use for HW collectives is liited to half of the addresses
    - This is not really a limit to job size
- **Shared system multicast addresses must be managed**
  - WLMs may be best suited to dynamically manage collective and multicast resources
    - WLMs already manage specialized compute-node resources
  - Multicast addresses are an abstract resource; would require significant development
  - System specific policies, such as queues, may be used to define which jobs are allowed to allocate collectives

Agenda

Brief Introduction to Collectives and Reductions

Software Implementation of Collectives in Slingshot

Hardware Implementation of Collectives in Slingshot

Interactions With Fabric Manager and Job Scheduler

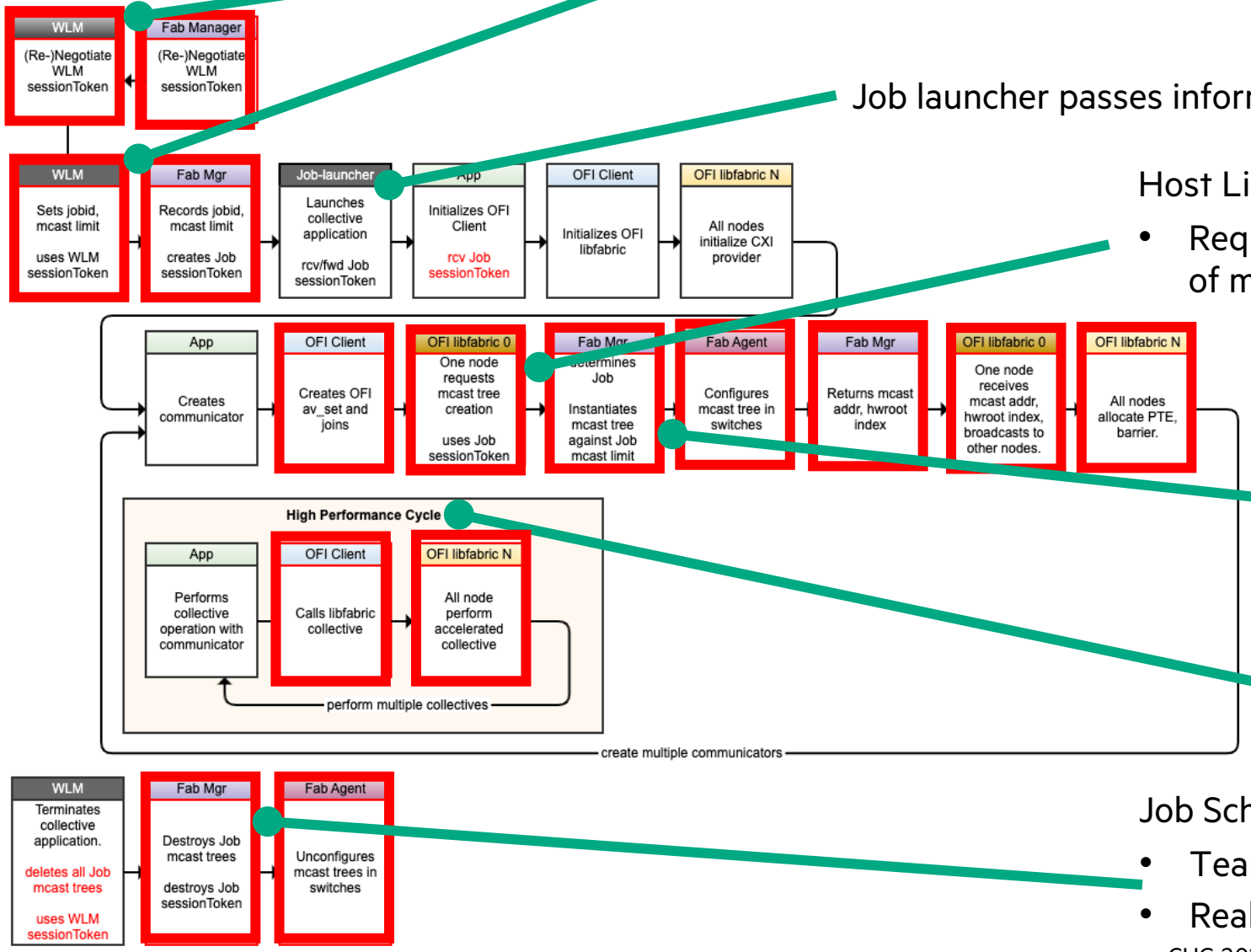
Schedule and Preliminary Results

Questions

Interaction With Workload Managers

- Need to reserve multicast address(es) at job creation time
  - Done via an authenticated REST call to the fabric manager
    - Authentication by “OAUTH” style token in first release
      - Future support for mTLS
  - Result is a job token
- Job token communicated to job at launch time via environment variables
  - Will be documented
  - These variables are used by libfabric’s Slingshot provider when told by MPI it should attempt to accelerate collectives
  - Job ranks communicate with fabric manager on first use of collectives to cause the tree to be formed in hardware
    - Authenticated by the job token
- Multicast addresses must be released at job termination
  - Causes underlying hardware resources to be torn down and made available for future use
- Workload Manager Integration
  - HPE working with SchedMD and Altair for native support in SLURM and PBS Pro
  - Additionally, using HPE provided documentation, LLNL is working to enable native support in Flux
  - Will provide documentation for integration into other WLMs
    - Theoretically can function without WLM integration through the use of scripting

The Flow



Fabric Manager-to-WLM Negotiation

- Session and job tokens
- Number of tree identifiers

Job launcher passes information to the application

Host Libfabric-to-FM

- Requests access and arming of multicast tree

FM-to-Switches

- Associate multicast trees with logical operation
- Program the switches

Host software/MPI

- Use the acceleration
- Reuse the tree

Job Scheduler to FM/FM to Switch

- Tear-down switch multicast tree
- Reallocate tree identifiers

New code/APIs/Env Variables/Etc.

Agenda

Brief Introduction to Collectives and Reductions

Software Implementation of Collectives in Slingshot

Hardware Implementation of Collectives in Slingshot

Interactions With Fabric Manager and Job Scheduler

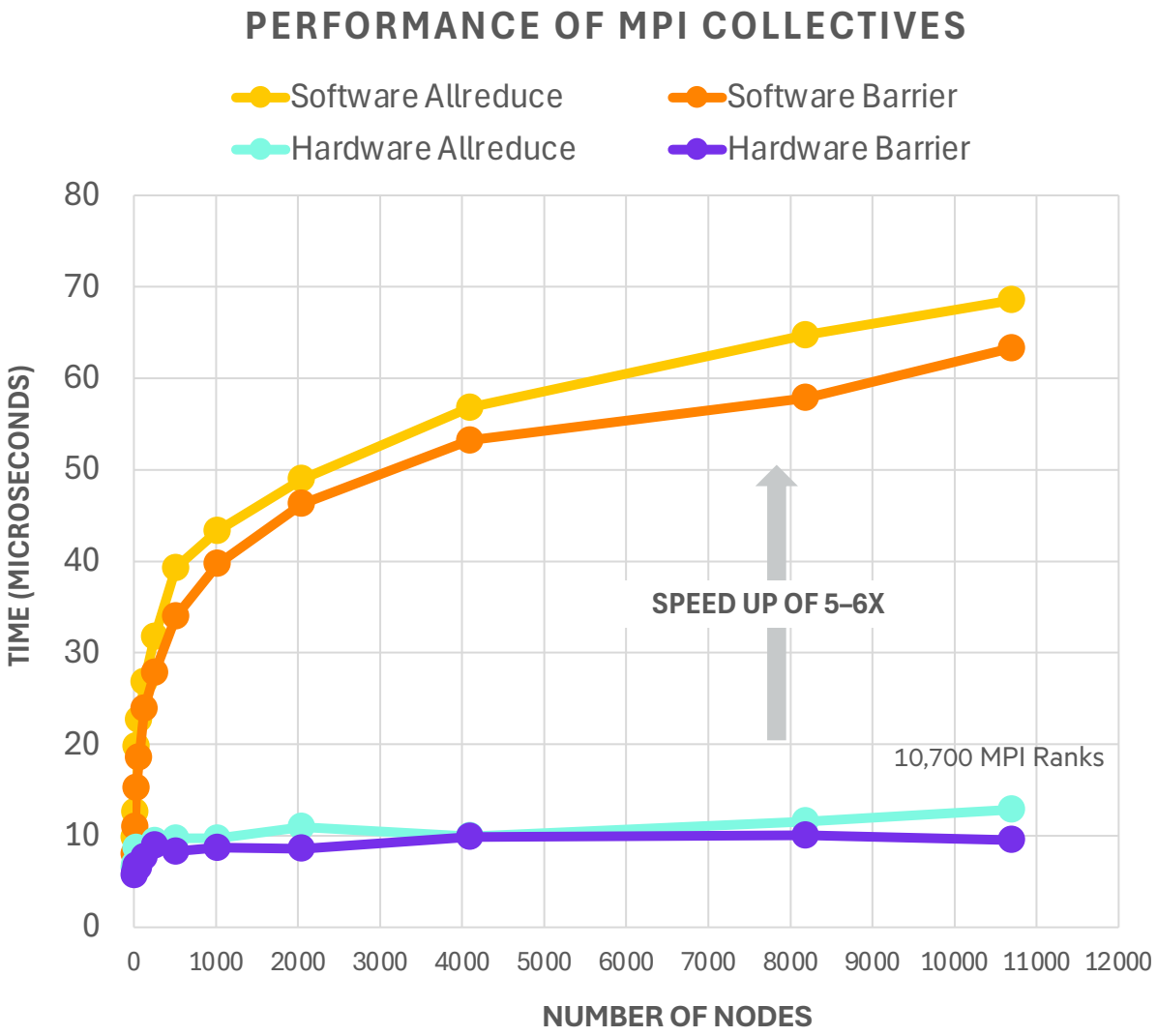
Schedule and Preliminary Results

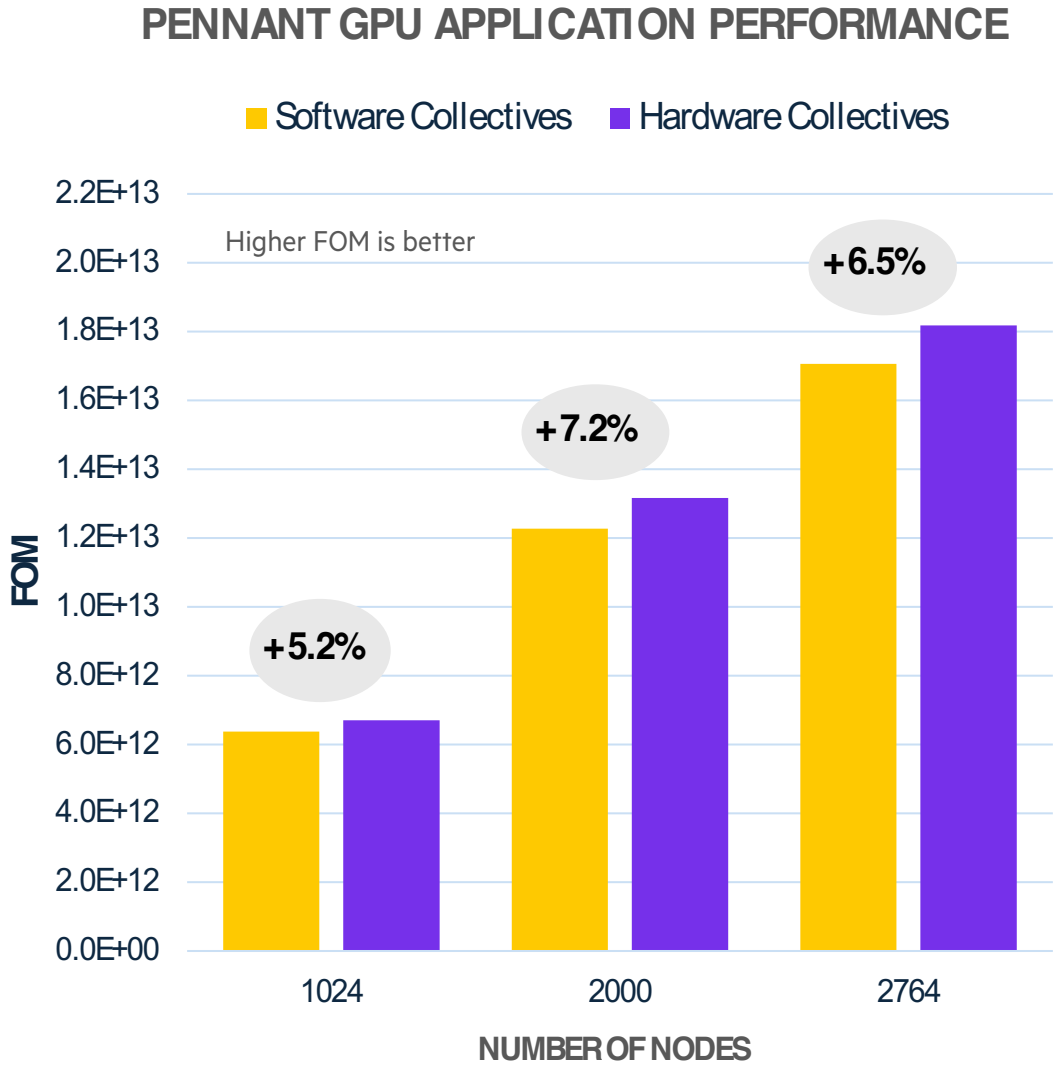
Questions

Schedule and Preliminary Results

- Scheduled to ship with the September “Recipe” releases
- Preliminary results
  - HPE would like to thank Lawrence Livermore National Laboratory for system access and system administration support in testing and benchmarking the prerelease of hardware accelerated collectives at scale on the El Capitan system
  - Results are, again, preliminary and may not reflect final software nor are they intended as an indication of system performance on El Capitan
  - Results Slide Credit to Kim McMahon at HPE who performed the benchmarking and assembled the slide

Hardware Collective Support in HPE Cray MPI





HPE would like to thank LLNL for system access to support development and demonstration of this capability at scale

Agenda

Brief Introduction to Collectives and Reductions

Software Implementation of Collectives in Slingshot

Hardware Implementation of Collectives in Slingshot

Interactions With Fabric Manager and Job Scheduler

Schedule and Preliminary Results

Questions

Thank you

Forest Godfrey
Distinguished Technologist, Slingshot
fgodfrey@hpe.com