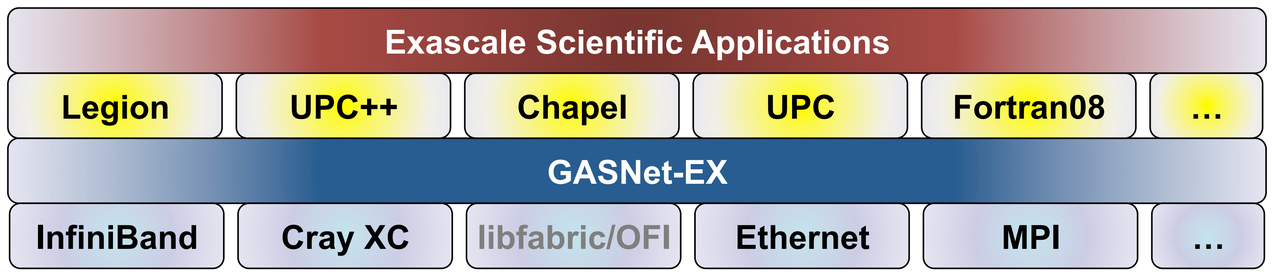
GASNet-EX: RMA and Active Message Communication for Exascale Programming Models

Paul H. Hargrove and Dan Bonachea

WBS 2.3.1.14

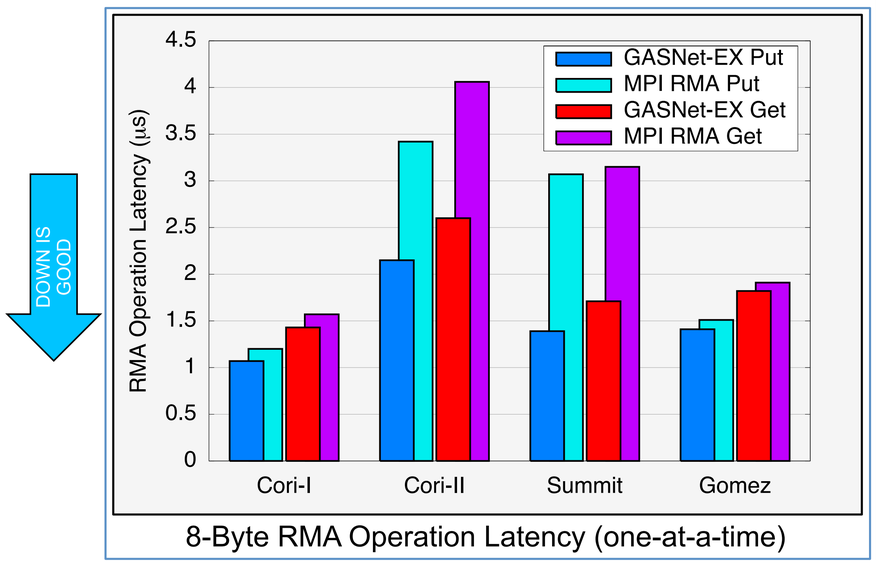
GASNet-EX at Lawrence Berkeley National Lab (gasnet.lbl.gov)

- GASNet-EX: communications middleware to support exascale clients
  - One-sided communication – Remote Memory Access (RMA)
  - Active Messages - remote procedure call
  - Implemented over the native APIs for all networks of interest to DOE
- GASNet-EX is an evolution of GASNet-1 for exascale
  - Retains GASNet-1's wide portability (laptops to supercomputers)
  - Provides backwards compatibility for the dozens of GASNet-1 clients, including multiple UPC and CAF/Fortran08 compilers
  - Focus remains on one-sided RMA and Active Messages
  - Reduces CPU and memory overheads
  - Improves many-core and multi-threading support
- Current enhancements:
  - "Immediate mode" injection to avoid stalls due to back-pressure
  - Explicit handling of local-completion (source buffer lifetime)
  - New AM interfaces, e.g. to reduce buffer copies between layers
  - Vector-Index-Strided for non-contiguous point-to-point RMA
  - Remote Atomics, implemented with NIC offload where available
  - Subset teams and non-blocking collectives
- Future enhancements include:
  - Communication directly to/from device memory (e.g. GPUDirect)

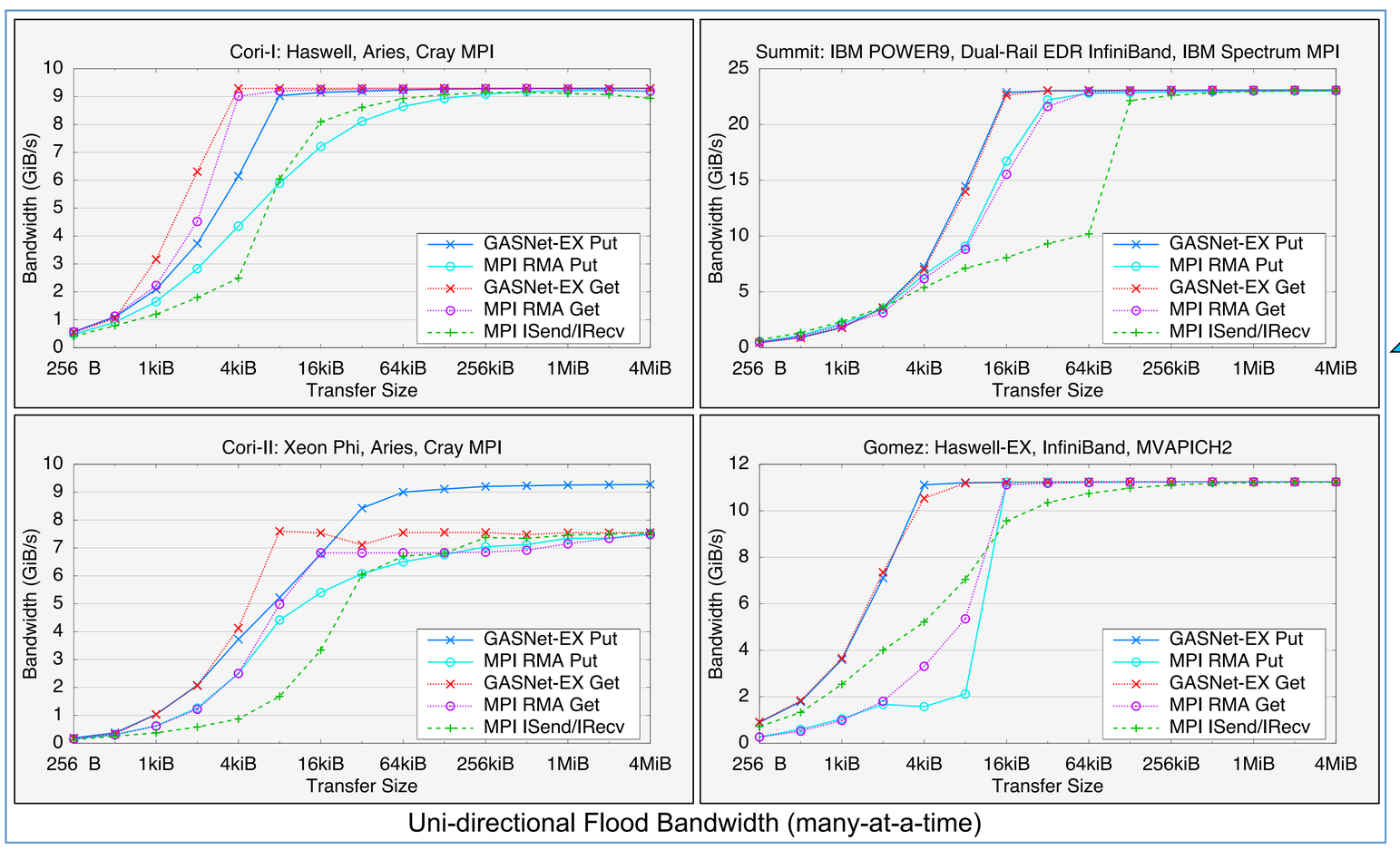


GASNet-EX RMA Performance versus MPI RMA and Isend/Irecv

- Three different MPI implementations
- Two distinct network hardware types
- On four systems the performance of GASNet-EX matches or exceeds that of MPI RMA and message-passing:
  - 8-byte Put latency 6% to 55% better
  - 8-byte Get latency 5% to 45% better
  - Better flood bandwidth efficiency, typically saturating at ½ or ¼ the transfer size



8-Byte RMA Operation Latency (one-at-a-time)



Uni-directional Flood Bandwidth (many-at-a-time)

GASNet-EX results from v2018.9.0
MPI results from Intel MPI Benchmarks v2018.1

For more details see Languages and Compilers for Parallel Computing (LCPC'18).
https://doi.org/10.25344/S4QP4W

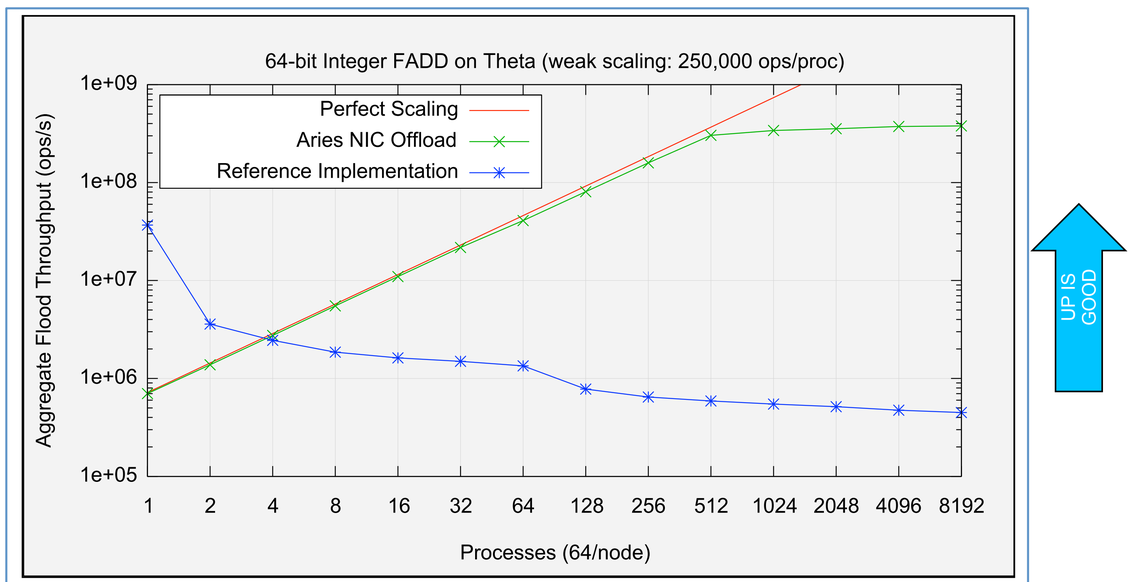
Remote Atomics with Cray Aries NIC Offload

- Implements the Atomic Domains concept (first introduced by UPC 1.3)
  - Domains permit use of NIC offload even when not coherent with CPU
  - Domains are created collectively outside the critical path
  - A Domain has an associated data type and set of allowed operations
    - Domains select the best implementation for the data type and ops
    - e.g. use offload if and only if NIC implements **all** the requested ops
- Example: non-blocking atomic fetch-and-add (FADD) on unsigned 64-bit integer

```
gex_Event_t ev = // *result = ATOMICALLY( *target += addend )
gex_AD_OpNB_U64(domain, &result, target_rank, target_address,
                GEX_OP_FADD, addend, 0 /*unused op2*/, flags);
```

- **`flags`** includes optional behaviors and assertions, such as memory fences
- GASNet-EX provides a network-independent "reference implementation"
  - Uses Active Messages to perform operations using the target CPU
  - Uses GASNet-Tools for atomicity (inline assembly for numerous CPUs)
- Specialization for Cray Aries improves performance vs. reference implementation
  - Reduces latency of inter-node FADD from 4.9us to 2.8us
  - Greatly increases throughput under contention

For more details see PAW-ATM'18 **https://doi.org/10.1109/PAW-ATM.2018.00008**



This figure shows throughput of 1 to 8192 processes (64 per node) performing pipelined FADD of a central counter (measured on ALCF's Theta).

This research was supported in part by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of the U.S. Department of Energy Office of Science and the National Nuclear Security Administration.

This research used resources of the National Energy Research Scientific Computing Center, a DOE Office of Science User Facility supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231.

This research used resources of the Argonne Leadership Computing Facility, which is a DOE Office of Science User Facility supported under Contract DE-AC02-06CH11357.

This research used resources of the Oak Ridge Leadership Computing Facility at the Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC05-00OR22725.