OpenSHMEM

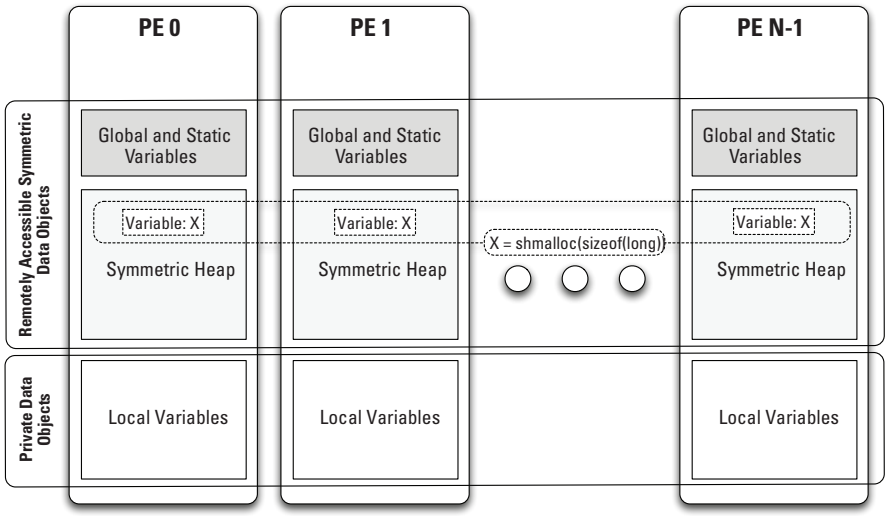
COMMUNICATION LIBRARY, SPECIFICATION AND TOOLS ECOSYSTEM

www.openshmem.org

OAK RIDGE National Laboratory | U.S. Department of Defense | Mellanox TECHNOLOGIES | ICL UT | UNIVERSITY of HOUSTON

OpenSHMEM is a Partitioned Global Address Space (PGAS) library interface specification. OpenSHMEM aims to provide a standard Application Programming Interface (API) for SHMEM libraries to aid portability across multiple vendors including SGI, Cray, IBM, HP, Mellanox, and Intel. OpenSHMEM supports one-sided communication that is a perfect fit for applications with irregular communication patterns with small/medium sized data transfers since it is optimized for low-latency data transfers. OpenSHMEM Library API provides calls for data communication, group synchronization, data collection, data reduction, distributed locking of critical regions, and data and process accessibility to OpenSHMEM PEs. PEs put/get data to/from remotely accessible symmetric data objects on other PEs.

OpenSHMEM IMPLEMENTATION OF PGAS PROGRAMMING MODEL



WHAT ARE SYMMETRIC DATA OBJECTS?

- Fortran data objects in common blocks or with the SAVE attribute. These data objects must not be defined in a dynamic shared object (DSO).
- Global and static C and C++ variables. These data objects must not be defined in a DSO.
- Fortran arrays allocated with shpalloc.
- C and C++ data allocated by shmalloc.

The OpenSHMEM interfaces can be used to implement Single Program Multiple Data (SPMD) style programs. It provides interfaces to start the OpenSHMEM PEs in parallel, and communication and synchronization interfaces to access remotely accessible data objects across PEs. These interfaces can be leveraged to divide a problem into multiple sub-problems that can be solved independently or with coordination using the communication and synchronization interfaces.

IMPORTANT OpenSHMEM EVENTS:

Daily OpenSHMEM/UCCS demo	Mellanox Technologies, Booth #2939
Monday, November 17	TUTORIAL **"Introductory and Advanced OpenSHMEM Programming"**, Room 393 (1:30 pm to 5 pm)
Tuesday, November 18	TALK **"Recent Advances in OpenSHMEM and UCCS"**, UTK Booth 2925 (4:00pm to 4:15pm)
Wednesday, November 19	BIRDS-OF-A-FEATHER SESSIONS
	PGAS – The Partioned Global Address Space Programming Model, Room 273 (12:15pm to 1:15pm)
	OpenSHMEM: Further Developing the SHMEM Standard for the HPC Community, Room 294 (5:30 pm to 7pm)
	Application Experiences with Emerging PGAS APIs: MPI-3, OpenSHMEM and GASPI, Room 386-87 (5:30 pm to 7pm)
Thursday, November 20	Featured OpenSHMEM Presentation at DOE booth #1939 (10:15 am to 10:45 am)
	OpenSHMEM/UCCS demo,DOE Booth #1939 (12:00 pm to 2 pm)

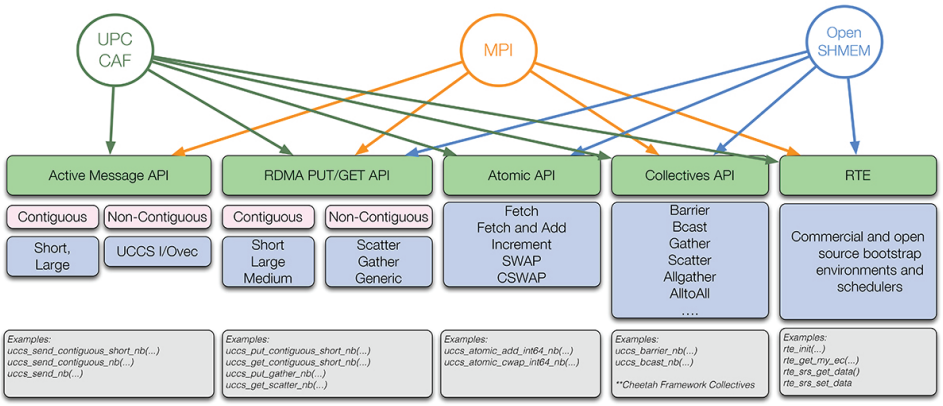
UCCS Universal Common Communication Substrate

WEB http://uccs.github.io/uccs E-MAIL uccs-info@ornl.gov

UCCS is a new universal, portable interface to expose low level high performance network capabilities. It provides commonly required services to implement high level parallel programming models, like lightweight remote memory access operations, lightweight synchronizations, active messages, atomic operations, and a portable runtime environment interface.

GOALS

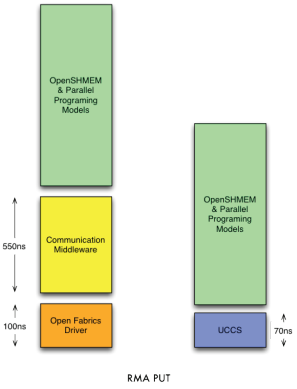
- Provide a common low-level scalable, robust, portable, simple and performance driven communication API for multiple parallel programming models over modern network interfaces
- Provide common network code for implementing programming models, increasing code reusability and reducing development effort
- Support hybrid programming environments efficiently
- Provide flexible API to accommodate requirements of I/O systems, Big Data applications, and Languages



API:

- All communications interfaces are non-blocking
- All communications interfaces are thread safe
- API independent of a bootstrap environment

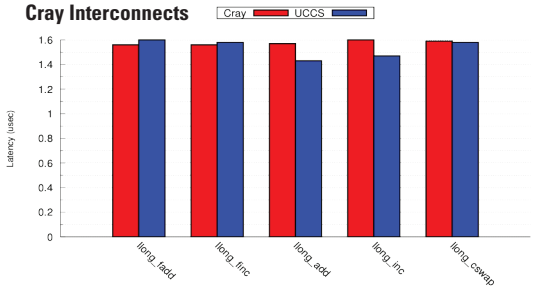
FASTER THAN DRIVER



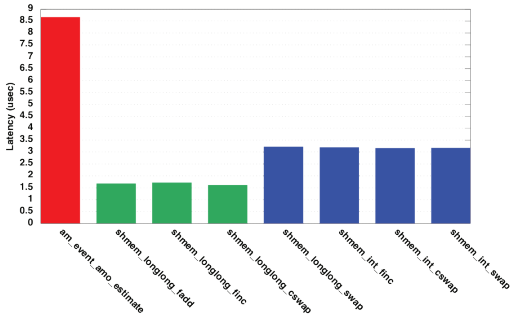
Very short communication paths for latency sensitive operations on Mellanox's MLX4 hardware

SUPPORTS MULTIPLE NETWORK TECHNOLOGIES

Cray Interconnects

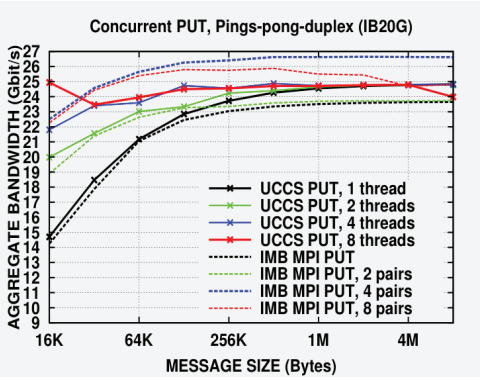


Mellanox Technologies



THREAD SAFE DESIGN

- All operations are thread safe: helps mixed model programming
- Specified to encourage fine grain locking: excelling performance when employing multiple threads
- Performance increases when using multiple threads to submit/progress communications



Even though MPI IMB benchmark benefits from full process memory separation, multi-threaded UCCS obtains comparable performance

Open SHMEM · OAK RIDGE National Laboratory · U.S. Department of Defense · Mellanox TECHNOLOGIES · ICL UT · UNIVERSITY of HOUSTON