SC18 OpenSHMEM BOF: OSSS OpenSHMEM-on-UCX

Tony Curtis <anthony.curtis@stonybrook.edu>

Abdullah Shahneous Bari Sayket

Wenbin Lü

Dr. Barbara Chapman

SC18 OpenSHMEM BOF: OSSS OpenSHMEM-on-UCX

- Reference Implementation
  - Communications: UCX
  - Wireup: PMIx
  - Collectives: SHCOLL (Rice)
- OpenSHMEM Spec. 1.4
- Will be 1.5, … as specifications released
- Optionally configured extensions

OSSS OpenSHMEM-on-UCX

Tony Curtis (SBU), Howard Pritchard (LANL)

http://www.openshmem.org/

- http://www.openucx.org/
- https://github.com/openshmem-org/osss-ucx
- https://pmix.github.io/pmix/
- http://www.open-mpi.org/
- https://github.com/pmix/prrte

Job scheduler ⟷ $ oshrun -np N program

PMIx server

- Reference OpenSHMEM 1.4 ++ Implementation
  - Open Source Software Solutions
  - LANL
  - Stony Brook U
  - Rice U / Georgia Tech
- UCX for communications
  - User and contributor

PMIx client

PE #0 ⟷ PE #N-1

OpenSHMEM

- OpenSHMEM API
- Collectives API
- Abstr. Comms API
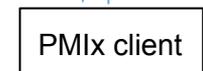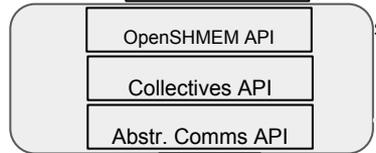
shmem_long_put   shmem_quiet   shmem_long_atomic_add ...

shmemc_put   shmemc_quiet   shmemc_add64   ...

- PMIx for startup, resilience
- Program launch via
  - mpiexec: Open-MPI
  - PMIx Reference RunTime Environment
    - PRRTE

UCX

UCP

UCT
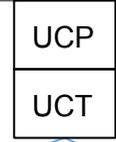
ucp_put_nb   ucp_worker_flush   ucp_atomic_post   ...

cma knem xpmem

IBV

· · · · ·

GNI

3
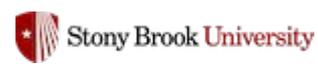
SC18 OpenSHMEM BOF: OSSS OpenSHMEM-on-UCX

- Eureka! Method
  - Early termination of distributed computation when 1 solution found anywhere
  - Uses
    - Active Messages
      - GASNet -> UCX: minimize spin waits
 - In coordination with LANL
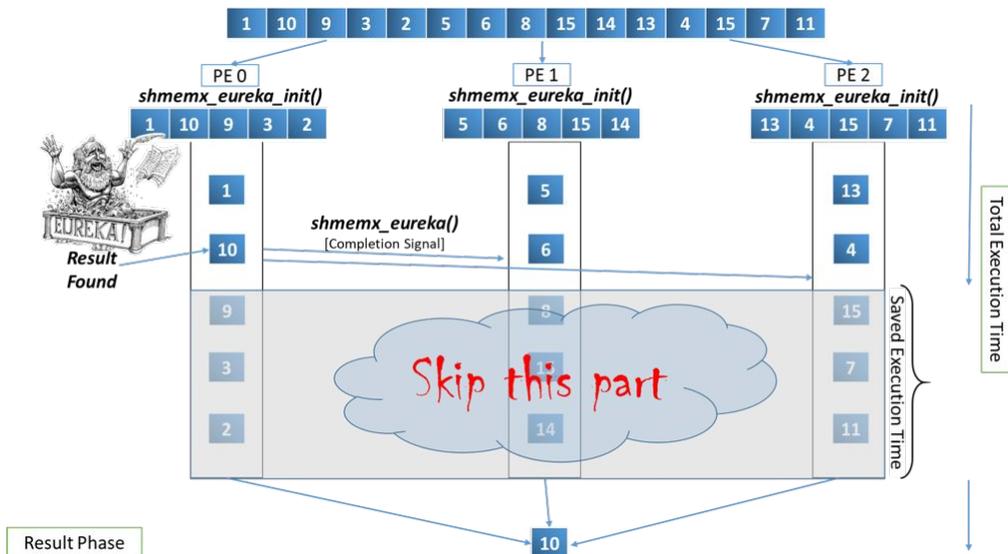
SC18 OpenSHMEM BOF: OSSS OpenSHMEM-on-UCX

- OpenSHMEM + threads/tasks
  - How OpenSHMEM interacts with OpenMP, tasking models
    - E.g. Contexts with Charm++, Legion
  - Resource partitioning
  - Thread/task allocation conflicts
  - Explore math methods that could work well
    - E.g. AMR

SC18 OpenSHMEM BOF: OSSS OpenSHMEM-on-UCX

- OpenSHMEM + graphs
  - Use of Rice's HOOVER to model mosquito-borne disease
    - More about HOOVER later...
  - Patch agent hybrid model
  - (Wenbin internship summer 2018 @ LANL)

● Fault Tolerance (NSF) #1

- ○ Project with UTK and Rutgers
- ○ Current OpenSHMEM specification lacks general fault tolerance (FT) features
- ○ PMIx has basic FT building blocks already
  - ■ Event handling, process monitoring, job control
- ○ Using these features to build FT API for OpenSHMEM
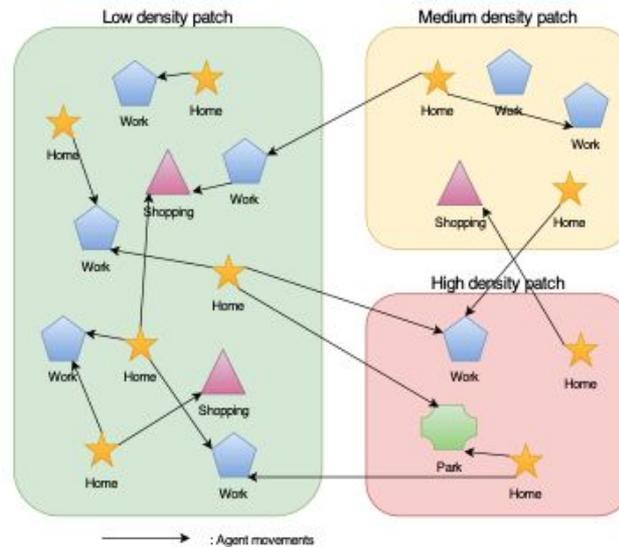
SC18 OpenSHMEM BOF: OSSS OpenSHMEM-on-UCX

- Fault Tolerance (NSF) #2
  - User specifies
    - Desired process monitoring scheme
    - Application-specific fault mitigation procedure
  - FT API takes care of:
    - Initiating specified process monitoring
    - Registering fault mitigation routine with PMIx server

- Fault Tolerance (NSF) #3
  - Longer-term goal: automated selection and implementation of FT techniques
  - Compiler chooses from a small set of pre-packaged FT schemes
  - Appropriate technique selected based on application structure and data access patterns
  - Compiler implements the scheme at compile-time

SC18 OpenSHMEM BOF: OSSS OpenSHMEM-on-UCX

● OSSS-UCX Reference Implementation

- ○ https://github.com/openshmem-org/osss-ucx/releases

SC18 OpenSHMEM BOF: OSSS OpenSHMEM-on-UCX

Thanks to sponsors and project partners

Tony Curtis <anthony.curtis@stonybrook.edu>

Abdullah Shahneous Bari Sayket

Wenbin Lü

Dr. Barbara Chapman

http://www.openshmem.org/

https://www.iacs.stonybrook.edu/