MPI-3 Collective WG

Torsten Hoefler and Andrew Lumsdaine

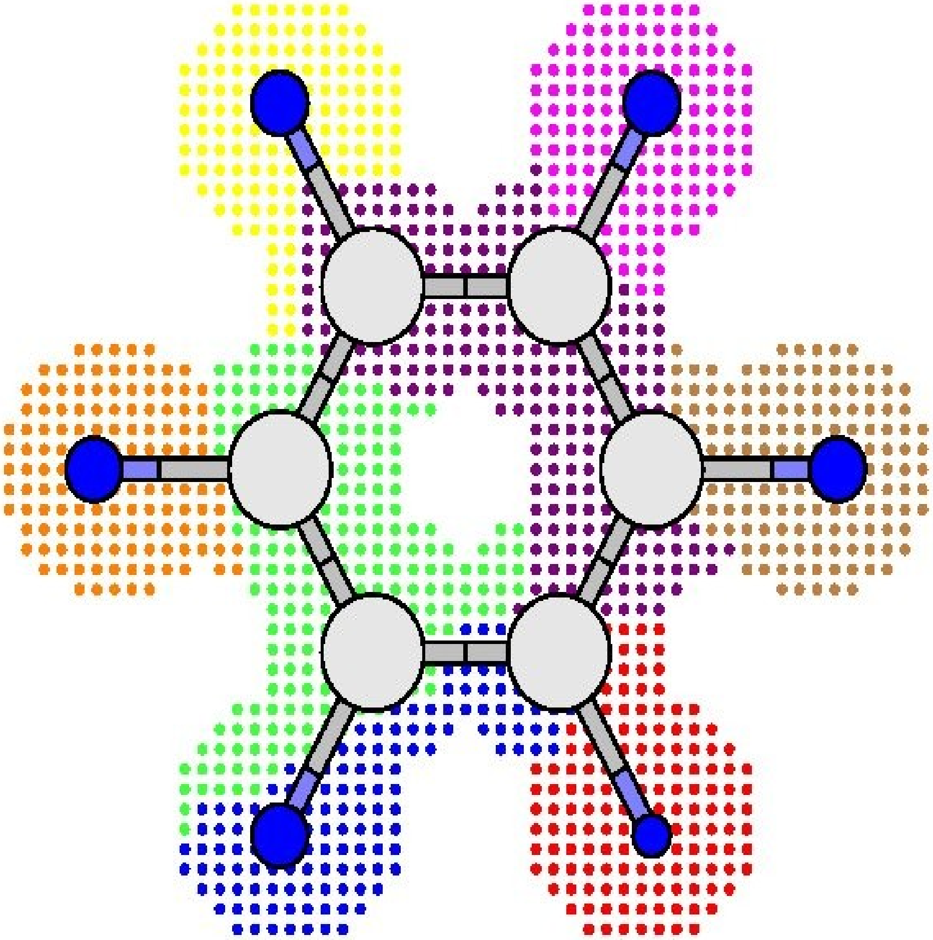
MPI Forum
March 12th 2008
Chicago, IL, USA

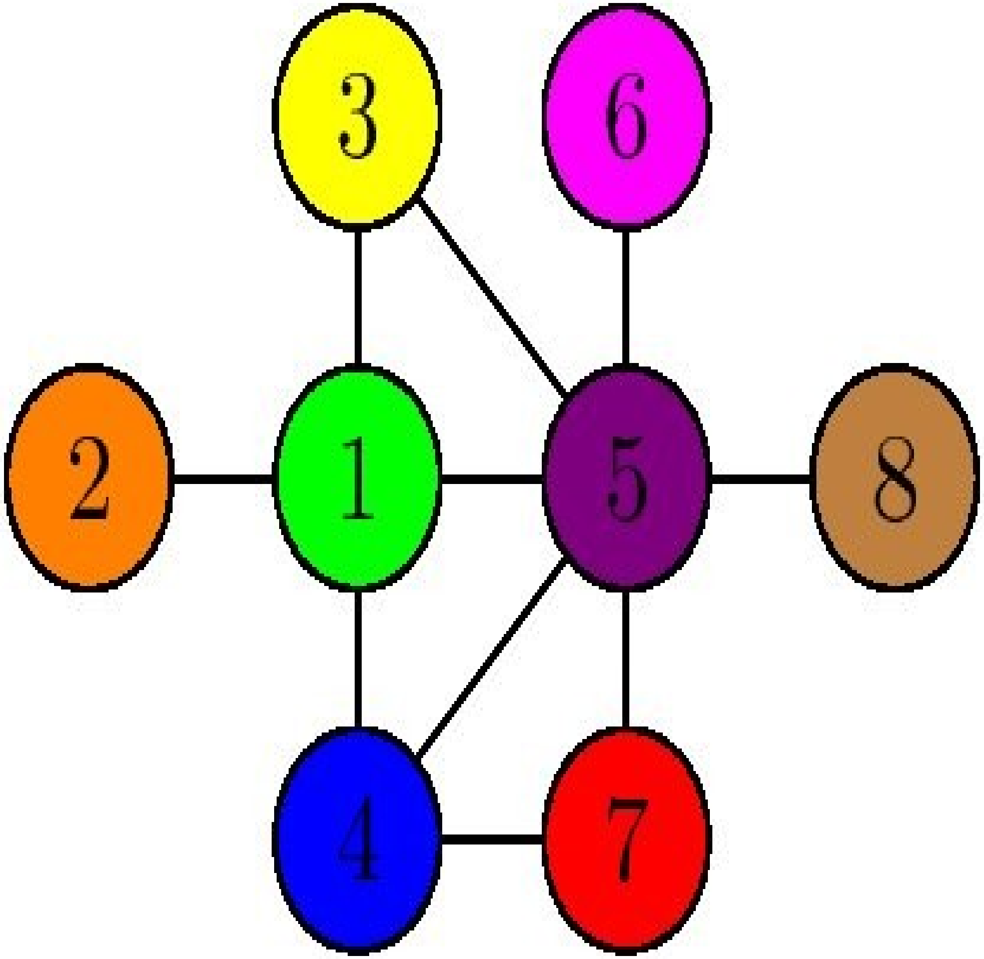
1st Working Group Meeting

MPI-3 Collective WG

1. New collective Ops (Jesper)
2. New collective Ops (Alexander)
3. Topological/sparse colls (Torsten)
4. Non-blocking colls (Torsten)
5. Persistent collectives (Tony?)
6. Subsetting (?)

Topological/Sparse Collectives





Topological/Sparse Collectives

- MPI_Alltoallv() is not scalable
- Topologies are not really “used”
- Collectives could use knowledge of Topologies (vs. users doing p2p)

Topological/Sparse Collectives

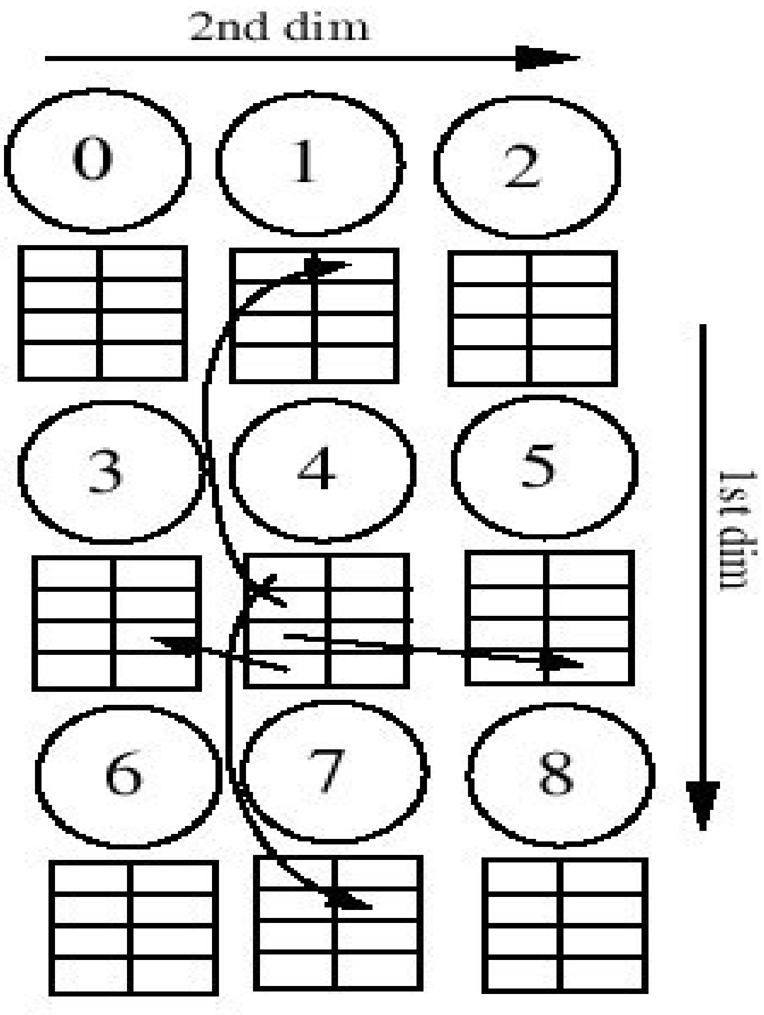
- MPI_Neighbor_xchg[v]()

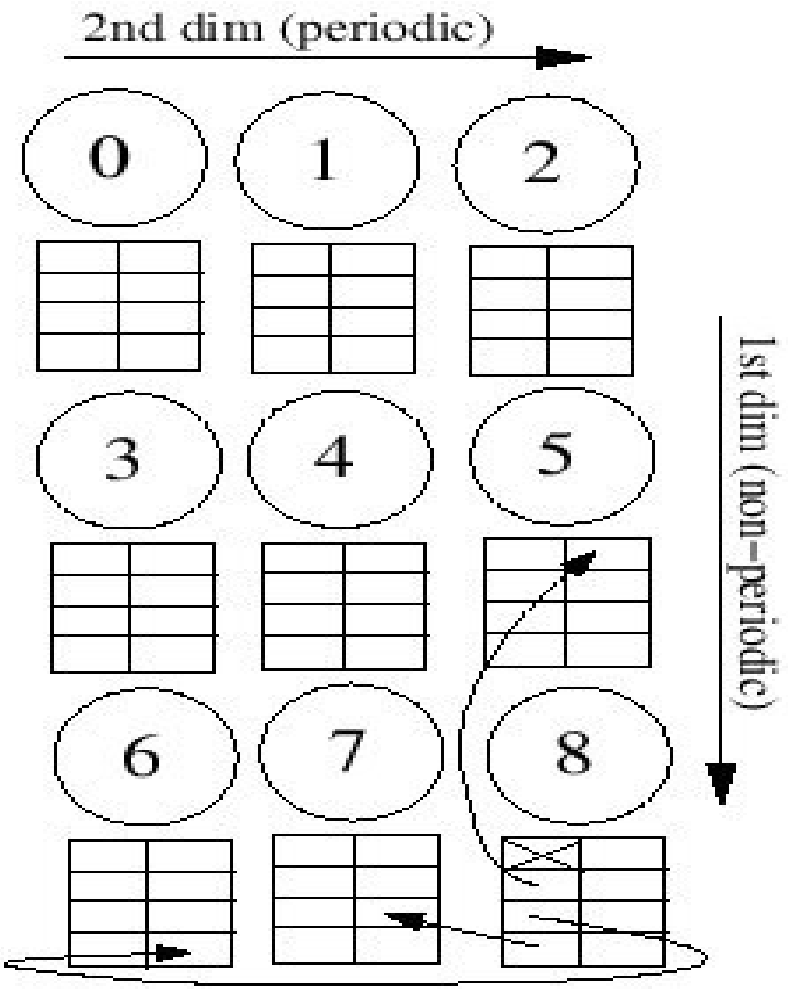
- MPI_Comm_neighbors_count()

- MPI_Comm_neighbors()

- MPI_Cart_shift_xchg()

Topological/Sparse Collectives





Issues

- Graph communicator definition is not scalable (full topology on every node)
- trivial change:
  - only have neighbor information at every host
  - remove rank argument from query functions (MPI_Graph_neighbors[_count]())

MPI-3 Collective WG

1. New collective Ops (Jesper)
2. New collective Ops (Alexander)
3. Topological/sparse colls (Torsten)
4. Non-blocking colls (Torsten)
5. Persistent collectives (Tony?)
6. Subsetting (?)

Non-blocking Collectives (NBC)

- new semantics
  - non-blocking barrier (cf. two-phase barrier)
  - runtime user-error checking (Mathworks' use-case)
- communication/computation overlap
  - hide latency
  - new programming principles

NBC - Interface

```
/* generate data */
MPI_Ibcast(..., MPI_Request &req);
/* do computation */
MPI_Test(&req, &flag, MPI_STATUS_IGNORE);
/* do computation */
MPI_Wait(&req, MPI_STATUS_IGNORE);
/* access communicated data */
```

NBC – Colls in Thread

- spawn thread and do blocking collective
- implemented and demonstrated at EuroPVM'07
- requires MPI_THREAD_MULTIPLE ;-)
- MPI does not define how to implement blocking colls (polling vs. interrupt)
- very likely to “loose” a core

NBC - Tags

- currently no tags in LibNBC
- close to original collective interface (cf. collective matching in threads)
- can easily be added (do we want a reference implementation)
- could be useful for debugging

NBC - Matching

- oringinal proposal defines matching between blocking and non-blocking collectives
- we do not want to impose this restriction
- algorithms for non-blocking colls could optimize for overlap, not for latency
- results in different algorithms that can not match

NBC - Free/Cancel

- ugh, complicated
- not even send/receive case is clearly defined (may fail)
- much more complicated protocols
- much more complicated implementation than for send/recv
- might have performance implications

NBC - Progression

- MPI does not define asynchronous progress
- high-quality implementation ;-) is free to implement is
- we propose not to change this
- might be a barrier for adoption of comm/comp overlap (programmers can not be sure if it works) ?

MPI-3 Collective WG

1. New collective Ops (Jesper)
2. New collective Ops (Alexander)
3. Topological/sparse colls (Torsten)
4. Non-blocking colls (Torsten)
5. Persistent collectives (Tony?)
6. Subsetting (?)