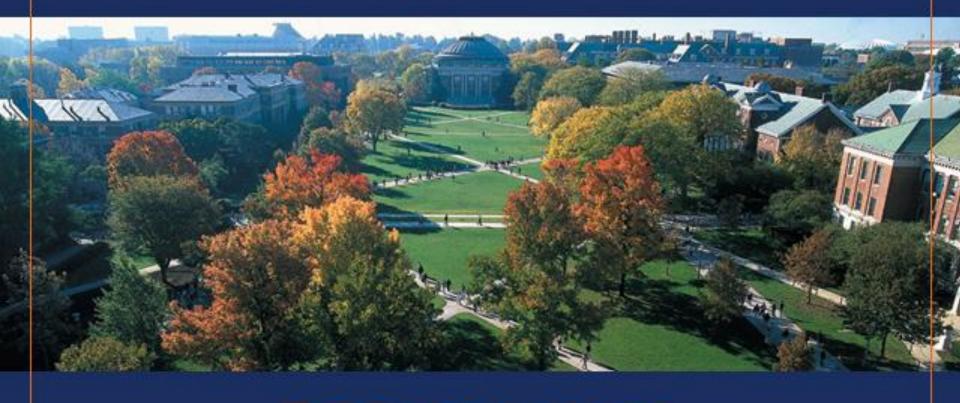
A Second Note on Assertions

Torsten Hoefler and Adam Moody



FILLINOIS

AT URBANA-CHAMPAIGN

MPI Assertions

- Previous proposal by Dick Treuman
 - Seems to be stuck somehow
 - Very inflexible (basically a fixed set of assertions at the beginning)
 - Puts the burden on us to decide which assertions to standardize
 - + weird discussions about bit vectors ©



Another Viewpoint

- We have MPI_Info objects
 - They're very flexible and allow for vendorspecific extensions
 - Test before standardizing
 - Good example: graph topologies
- Usage of MPI_Info is inflexible
 - Some functions have info arguments,
 some don't, seems rather random ⁽³⁾



Info vs. Assertions

- Info arguments can be used to assert usage of MPI as long as:
 - They can safely be ignored!
 - i.e., they don't affect correctness if ignored
 - BUT: they may affect correctness if used wrongly (e.g., "no_locks")
- Usually used for performance
 - Similar to main motivation for assertions



Assertions through Info?

- Info objects are used for two purposes:
 - assertions (e.g., no_lock)
 - hints (e.g., access_style)
- We could reflect this in the syntax (e.g., add explicit MPI_Assert)
 - remember the rule: naming comes last
- In the following, we'll use Info as in MPI-2.2



Proposal

- Allow to query info values (important for modularity if info is an assert! Bug in 2.2?)
 - MPI_Comm_get_info(MPI_Comm comm, MPI_Info *info)
 - Returned info can be decomposed with info functions
- Allow to attach new info values to MPI objects (comm, win, ddt, ...)
 - MPI_Comm_add_info(MPI_Comm comm, MPI_Info info)
 - Key/value pairs are merged into the existing info!
 - Can easily be extended!



Possible Assertions

Per Communicator

- no_any_source simplifies matching
- no_any_tag simplifies matching
- free_strict no reference counting
- no_reduction_determinism optimized colls
- no_heterogeneity simpler (D)DT handling
- no_send_cancel simpler send handling
- match_packed packed messages will only match packed messages



Possible Assertions

- Topological Communicator
 - strict_topology communicate *only* along topology (saves resources)
- Datatype
 - free_strict no reference counting
- Windows
 - no_locks exists already ©
 - no_ordering proposed in MPI-3 RMA WG



Discussion

- Are those useful?
 - Does it matter that the assertions come "late"?
 - Infos can never be removed (only added or queried) – modularity
- More/other assertions or objects?
 - Should see a particular benefit!
 - Proposals?



Discussion

- Are attached info objects copied in MPI_Comm_dup?
 - I assume not but we might want to handle it similarly to attributes (?)
 - Infos can never be "needed", they only enhance performance (by constraining semantics if used as assertions)

