UPC and GASNet Collectives

Paul H. Hargrove
LBNL
UPC Community Background

• UPC Collectives Spec v1.0 completed
  - Broadcast, Scatter & Gather
  - Gather All & Exchange
  - Permute
  - Reduce & Prefix Reduce
  - Sort

• Semantics are very unlike MPI collectives

• UPC-level reference implementation (MTU)
  - Not designed for performance
  - No portable use of hardware support
Overview of the work

• Overall goal
  - Tuned implementation of UPC Collectives in GASNet
  - Platform for collectives research

• The steps
  - Extensions to GASNet specification
  - Reference implementation in GASNet
    - Extensible, customizable, & tunable
  - Tuned implementations for specific networks
GASNet Extensions

- Forward-looking design
  - Split-phase (a.k.a. non-blocking)
  - Teams (subsets of UPC threads)
  - Aggregation hints (for optimizations)
- Inclusive Design
  - Titanium, CAF and even MPI
- Status: Interface design nearly complete
  - Now implementing to validate
  - Target is Summer
Reference Implementation

- General applicability
  - A portable default implementation
  - A layer over the remainder of GASNet
- Tunable
  - Compile-time and/or run-time selection of algorithms and parameters
- Customizable
  - Easy to override with network-specific implementations
- Status: Early stages (framework + bcast)
  - Target is end of FY04
Optimization Opportunities

• Network-specific support
  - Choice of algorithms and parameters
  - Use direct hardware support
    - e.g. Quadrics barrier & broadcast

• Aggregation
  - Amortizes synchronization (barriers)

• Automatic tuning
  - Build “optimal” schedule
    - LogP or LogGP model
  - Lottery Scheduler (Rajesh’s talk)
Preliminary Results

Latency of 8-byte Broadcast (elan-conduit)

- UPC (blocking)
- UPC (non-blocking)
- GASNet
- - - Hardware (estimated)

Latency (us)

Number of UPC Threads

10000
1000
100
10
1

7.4us
13.8us
296us
936us
1009us