IAA Interconnect workshop: Topologies/routing work group

Craig Stunkel
Chita Das

Outline of work

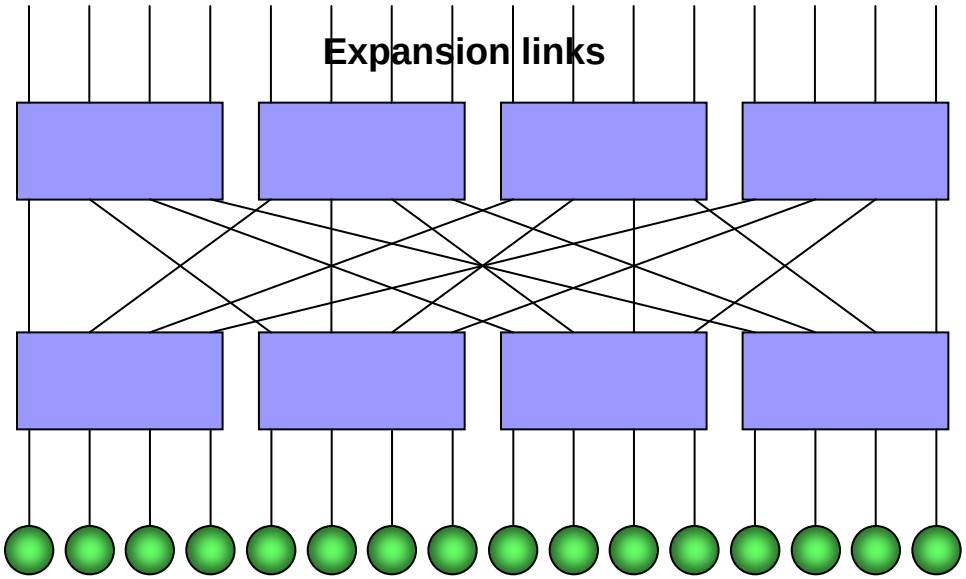
- Background:
  - Today's high-performance networks
  - Exascale game changers
    - Applications
    - Technology
    - Architecture
- Main discussion:
  - Challenges for exascale switches and routing
  - Trends

Today

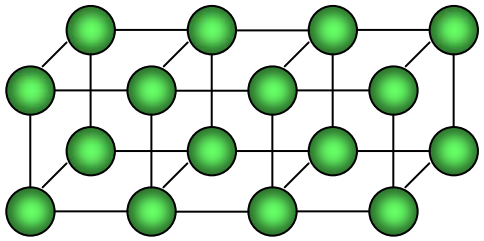
- A multitude of networks have been built, and many more have been proposed
- Two main topology families currently dominate
  - Fat-trees/Clos
  - k-ary n-cubes
- Why
  - High global bandwidth
  - Acceptable performance for important traffic patterns
  - Simple routing, few virtual channels required
  - Fault-tolerance, adaptivity
  - Incrementally scalable

Topologies

- Only two families are of major significance today:
  - Fat-trees/Clos and similar “indirect” networks
  - k-ary n-cubes (meshes, tori, hypercubes)
    - Used in several tightly-packaged MPPs/clusters



Two-level 4-ary fat-tree



3D mesh (4x2x2)

Exascale application game changers

- Lessons from DOE applications & Peta/Tera scale systems
- Streaming?
  - Need high bandwidth, but perhaps not low latency or fast switching
  - Unidirectional bandwidth
- Data mining
- Virtual worlds/reality?
- More accurate physical models
  - More modeling complexity
  - Will nearest-neighbor communication be sufficient for many new apps?
- Bottom line: How many applications can utilize such systems?
- …

Application panel messages

- Plea: let us keep the illusion of a flat, uniform system
- Point-to-point typically bandwidth-bound
- Collectives typically latency-bound
- Both bandwidth and latency will become more challenging as technology scales
- A few very important apps need FFTs
  - Bisection BW important
  - But the apps are currently message overhead-bound instead of bisection-bound
- Strong scaling will become important with flat CPU scaling
  - Will favor implicit methods
  - Latency important
- MPI will remain the most important programming model
  - Although hopefully other models will become more important
  - Most people don't program in MPI, but instead to abstractions/libraries
- Hierarchical programming models (on-chip, off-chip)
- How to handle heterogeneity?
- Fault tolerance an important problem for the whole stack (including HW)
- Both bandwidth and latency important, for different apps
- …

Exascale technology game changers

- 10's of billions of transistors per chip
  - Huge performance
  - Off-chip bandwidth will not increase as fast
- Integrated optics / CMOS-compatible photonics
  - Will it be cheap?
- 3D Stacking
- Phase Change Memory
- Proximity Communication for high radix routers
- Power and technology implications
- …

Exascale architecture game changers

- Switch radix growing
- Many-core chips will require on-chip networks
  - Does this create a hierarchy of networks?
  - How do internal and external networks cooperate with & complement each other?
  - Will the on-chip networks themselves be hierarchical?
- Commodity switches don't natively support torus (ring) routing …
  - Although they can sometimes accommodate via VC mapping

Switch element radix is typically growing

- Switch elements are getting larger (more I/O available)
  - This typically translates to less hops
    - Higher performance, given same buffering per port
- However, it is difficult to grow both port bandwidth and radix on a single chip
  - Eventually requires multiple-chip switch elements
    - More costly and complex
    - And then the minimum packet size must grow
    - Proximity communication should help

Switch element radix growing

- Popular solution: *don't* try to achieve both high radix and high port bandwidth
  - Just go for high radix …
  - … and have multiple switch planes/networks if needed to achieve high bandwidth
  - Better aligned with commodity (cheap) switch requirements
- 24-port IB-4x switch silicon is a great example

Routing in Exascale systems

- Necessity of moving from simple oblivious routing to sophisticated routing schemes?
  - Performance
  - Energy
  - QoS/Fault-Tolerance
- Understand Exascale application requirements

Discussion

- Network size
- Performance requirements
- Power consumption
- Reliability
- Performance-Power-Reliability tradeoffs
- Performance monitors and Tools
- Is dynamic adaptation possible (e.g., for latency/BW trade-offs)?
- Role of the network interface (how many, how transparent?)

Brainstorming: Top 10 Challenges

For each challenge:

1. Probability that the challenge will not be solved by relying on current technology trends: high, med, low
2. Impact that the lack of a solution for this challenge will have on the ability of the HPC community to build an exascale computer by 2016: high, med, low. That is, HIGH means that if we don't have a solution for this problem, there is no workaround solution that will allow us to build the exascale system
3. Approximate NRE cost for a solution: high (greater than $15M), med ($5-10M), low (less than $5M)