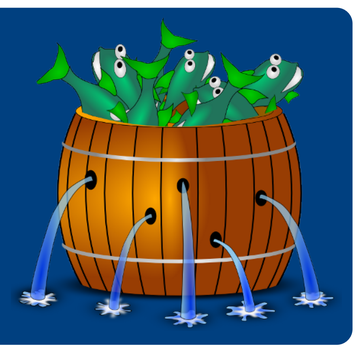
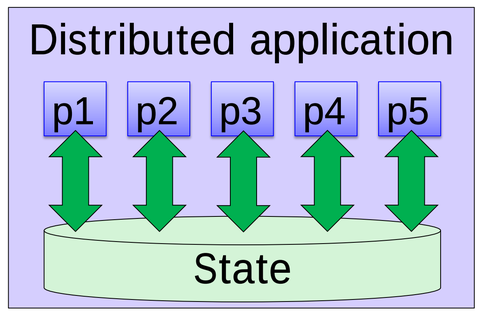
What does distributed computing look like on a multicore machine?



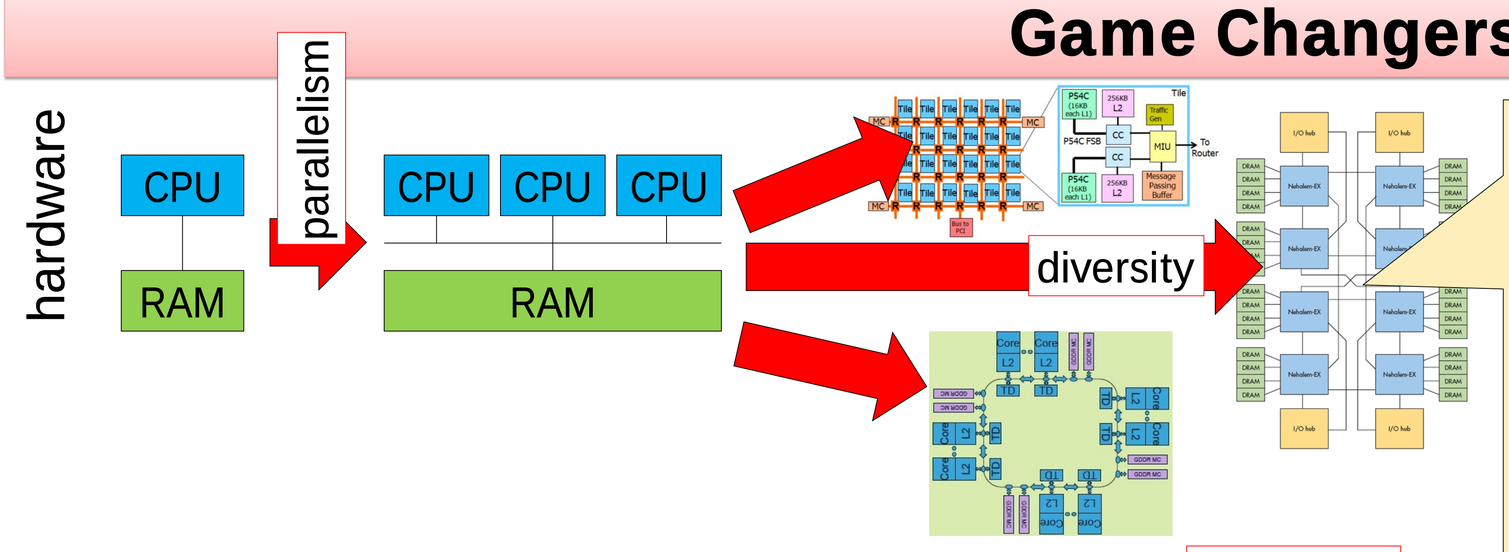
Stefan Kaestle, Timothy Roscoe — Systems Group, Department of Computer Science, ETH Zurich

Problem

Application of **traditional distributed computing principles** to enable scalability of distributed applications on **multicore machines** with complex, hybrid interconnect networks



Game Changers



- **Networks different** from traditional ones
  - Often: no failures, in-order delivery, etc.
  - **Different complexity measures**
    - Propagation negligible
    - concept of rounds meaningless
- **Hybrid** networks:
  - Combination of networks in one machine

- **Reuse distributed algorithms**
  - Group communication
  - Agreement
  - Leader election
- But: re-evaluate for multi-cores

Performance characteristics **hard to understand**, but essential to tune algorithm performance

Ideas

- **Model** multicore networks characteristics
  - Latency, synchrony, loss, reordering, caches, ..
- **Automatically** select **combination** of algorithms
  - based on model

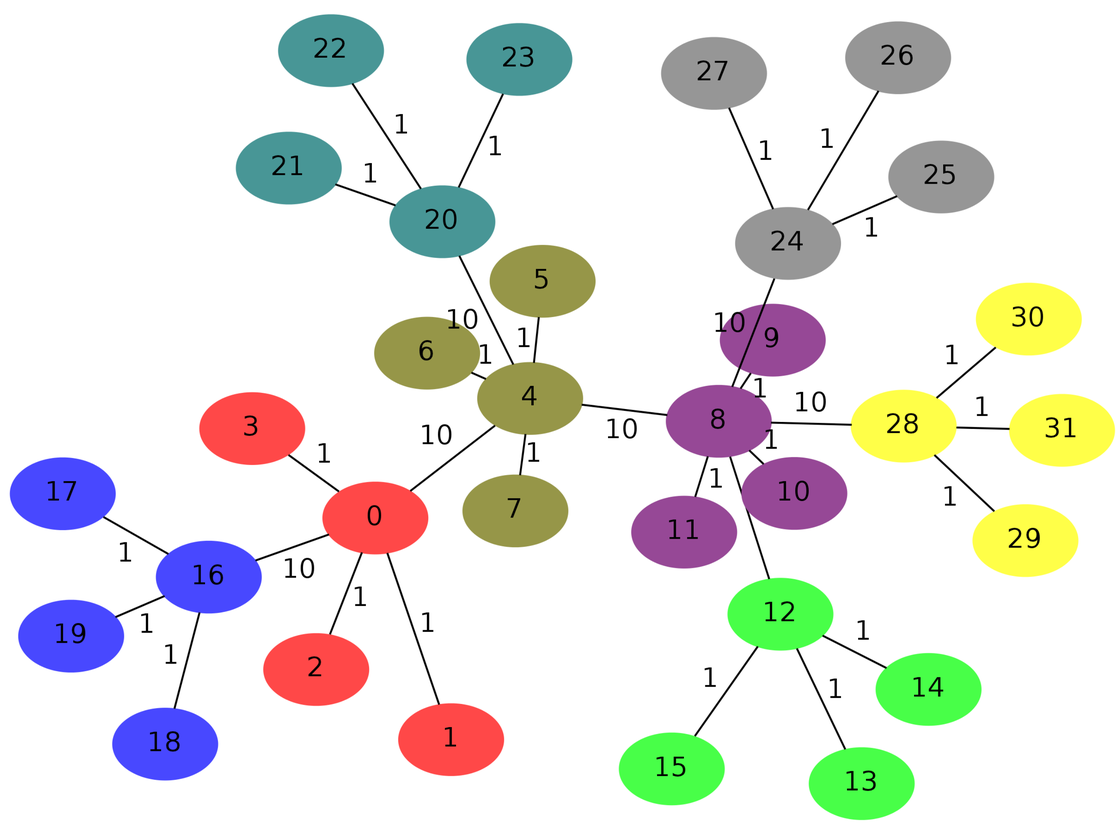
  → Easier and **less tedious for programmers**

Questions

- Applicability of existing distributed algorithms?
- Impact of new kind of network on existing research?
- What does the **programming model** look like?

Example: group communication on broadcast tree

- **Model:** fully meshed graph
  - Nodes: CPUs, edges: latency
- Apply MST algorithm
- **Differences** to traditional systems:
  - Send cost dominates → Parallelize send
- **Hybrid** → combine algorithms
  - E.g. shared caches within NUMA node
- Note:
  - Communicate only once to every NUMA node
  - Time complexity: $O(\log n)$ instead of $O(n)$
  - Traditionally: just one round for sequential send
- On top: build higher-level algorithms (e.g. consistency)



MST for 8x4x1 multicore (colors indicate NUMA nodes)

ETH Eidgenössische Technische Hochschule Zürich Swiss Federal Institute of Technology Zurich

inf Informatik Computer Science

We thank Microsoft for financial support

Systems@ETH Zürich