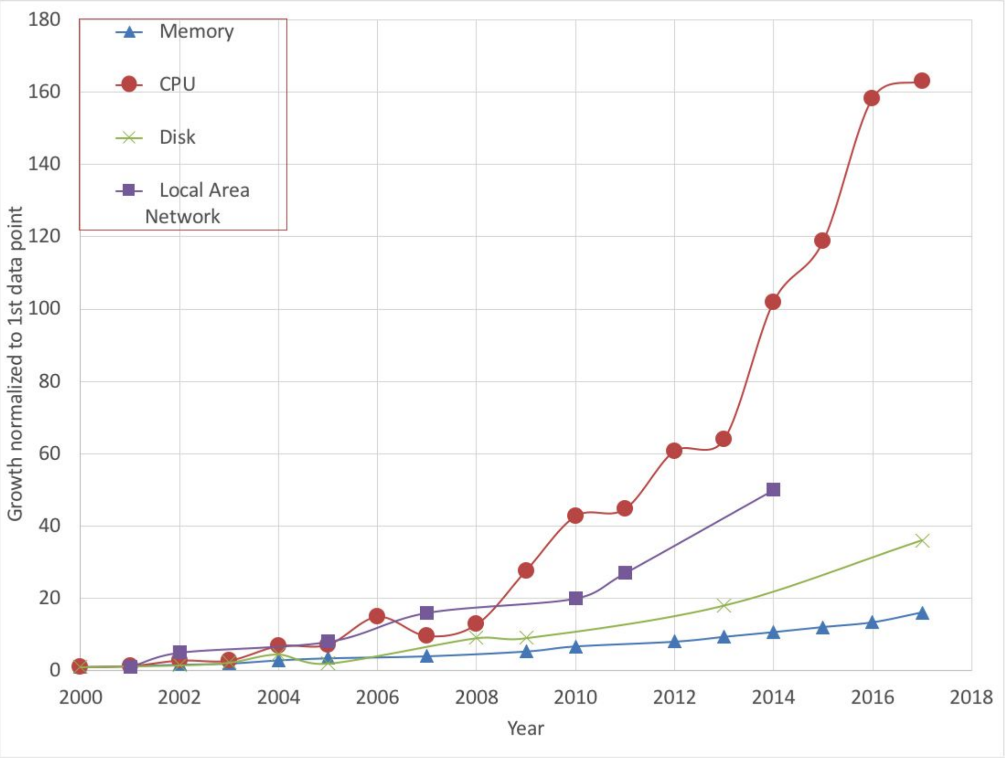
THE GOLDILOCKS NODE: GETTING THE RAM JUST RIGHT

Collaboration between Kove, Argonne, Illinois Institute of Technology, University of Reading

Data Centric Computing for the Next Generation BOF, ISC 19 Jun 2019

WOULDN'T IT BE NICE…

- If a facility could have
  - exactly the right combination of resources
  - put them together exactly the way each application needed them; and
  - upgrade each resource completely independent of the others?
- We already do it with disk (SAN)…
- Why not RAM?
  - We replace nodes to get newer CPUs, but throw away perfectly good RAM.
  - We build "big memory nodes" which are either poorly utilized or have a queue a mile deep.



Rethinking the Use of RAM

- **RAM Area Network (RAN)**
  - Put minimal RAM on the compute node (32GB-64GB)
  - Put a pool of RAM on the network
  - Allocate additional RAM when and where it is needed
    - Allow the RAM pool to be **persistent** as a cherry on top.
  - Early results suggest this could be a killer app for some applications (Deep learning)
    - Small GPU memory and computation time allow latency hiding.
  - *This topic will be the focus of the talk today*

- Alternative uses: RAM can be considered as **persistent storage**
  - e.g. when streaming visualization data through GPUs.
  - Fast additional storage tier **or** independent storage location
    - Should people care where data is stored?
  - Temporary storage for IO reordering to improve disk IO performance.
  - *We consider these aspects and more in the collaboration (out of scope for slides)*

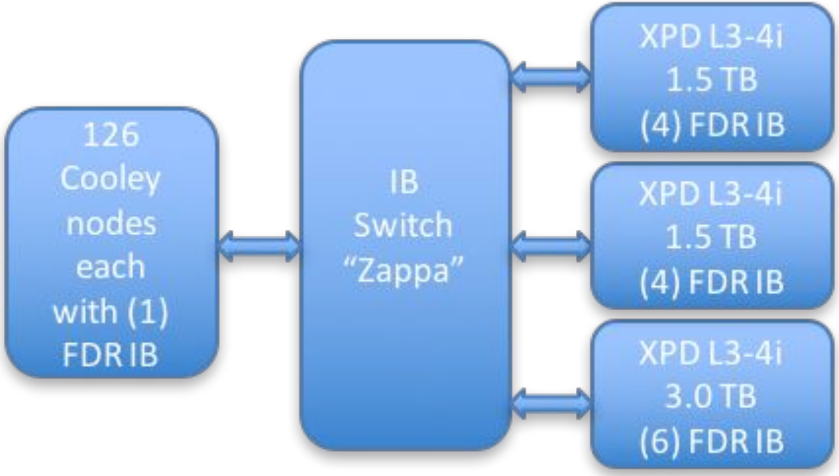
RAM AREA NETWORK TO THE RESCUE

- Advantages
  - Reduction in aggregate resources and therefore CAPEX and OPEX
  - No more “this node has unused RAM and that one is out”
  - No need for heterogeneous clusters (big memory nodes)
    - Maximum size limited by the motherboard
    - What if that still isn’t big enough?
  - Another alternative in the “deep memory hierarchy”
  - Changes in facility needs for RAM can by trivially and dynamically addressed
  - Decouple the purchases
- Disadvantages
  - Performance, particularly latency and jitter
  - Some apps/algorithms would need to be rewritten
  - Some will simply not be amenable to it
  - Power, maybe: More data movement vs. less aggregate offset each other

SOUNDS GOOD. HOW DO WE DO THAT?

- Management interface that allows dynamic allocation of memory ”volumes”
- Usage alternatives
  - – Xmem kernel driver
    - – Transparent access
      No changes to the application.
    - – Accomplished by intercepting malloc() and mmap() calls.
    - – Native QEMU/KVM integration starting in RHEL 7.5
    - – Can also be used as a block device
  - – Explicit memory management
    - •Needs modification of the application
    - •C, Java, and memkind interfaces

Prototyped at Argonne


126 Cooley nodes each with (1) FDR IB
IB Switch “Zappa”
XPD L3-4i 1.5 TB (4) FDR IB
XPD L3-4i 1.5 TB (4) FDR IB
XPD L3-4i 3.0 TB (6) FDR IB

Partners in the Collaboration

- Argonne Leadership Computing Facility (long-term collaboration with Kove)
  - *William (Bill) Allcock:* Prototyping of novel approaches for applications and systems
  - ALCF provides the development environment that includes XPD memory appliances
- Kove – small Chicago Tech Company
  - Produce the XPD memory appliances and software
  - Virtualization and got support into RHEL driver
- University of Reading
  - *Dr. Julian Kunkel* - Kove MPI-IO driver, cost modeling, monitoring
- Illinois Institute of Technology
  - *Dr. Zhiling Lan*
    - Multi-objective scheduling (how do we balance nodes, RAM, burst buffer, etc.)
    - Use of RAN in Machine Learning and Deep Learning applications.
  - *Dr. Xian-He Sun* – Memory performance modeling and optimization
- **Your name here? We are very interested in expanding the collaboration.**