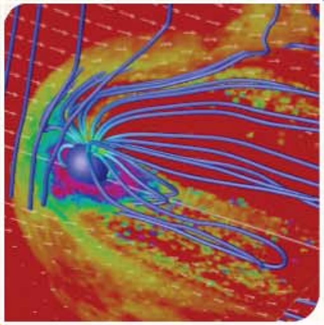
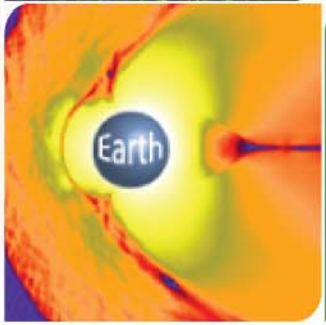
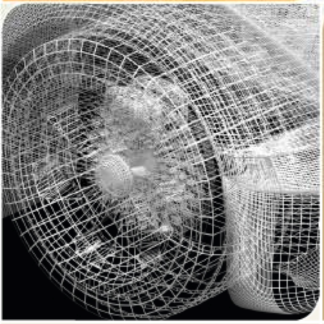


Large Scale Complex Network Analysis using the Hybrid Combination of a MapReduce Cluster and a Highly Multithreaded System

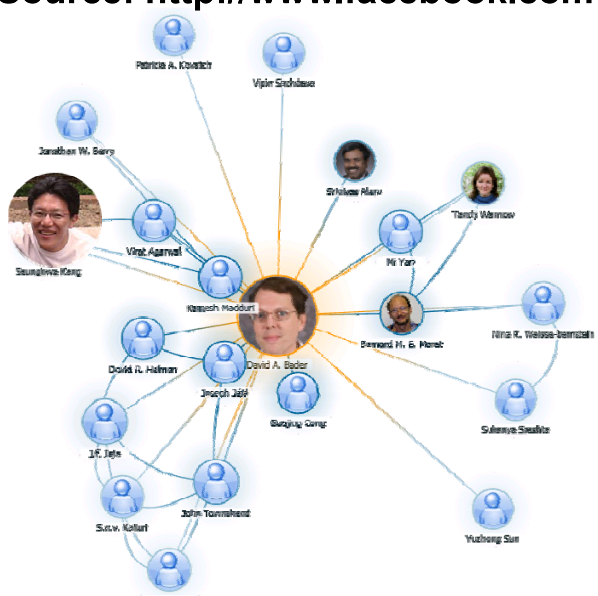
Seunghwa Kang **David A. Bader**

Georgia Tech College of Computing
Computational Science and Engineering



Various Complex Networks

Source: http://www.facebook.com



Source: http://academic.research.microsoft.com

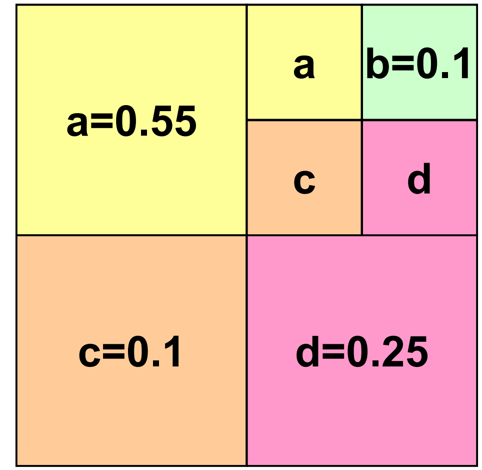
- **Friendship network**
- **Citation network**
- **Web-link graph**
- **Collaboration network**

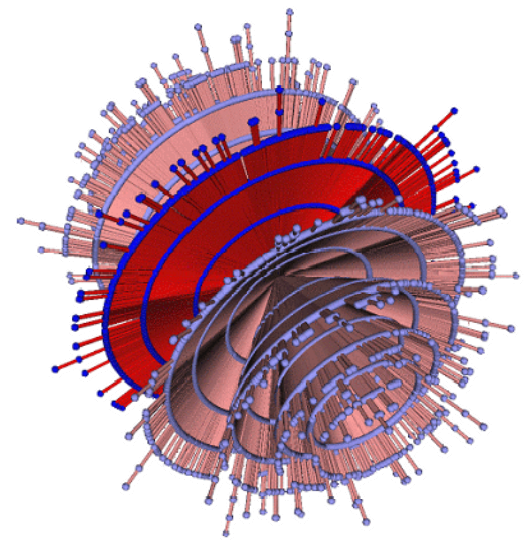
=> Need to extract graphs from large volumes of raw data.

=> Extracted graphs are highly irregular.

A Challenge Problem

- **Extracting a subgraph from a larger graph.**
  - **The input graph: An R-MAT* graph (undirected, unweighted) with approx. 4.29 billion vertices and 275 billion edges (7.4 TB in text format).**
  - **Extract subnetworks that cover 10%, 5%, and 2% of the vertices.**
- **Finding a single-pair shortest path (for up to 30 pairs).**





Source: Seokhee Hong

* D. Chakrabarti, Y. Zhan, and C. Faloutsos, "R-MAT: A recursive model for graph mining," SIAM Int'l Conf. on Data Mining (SDM), 2004.

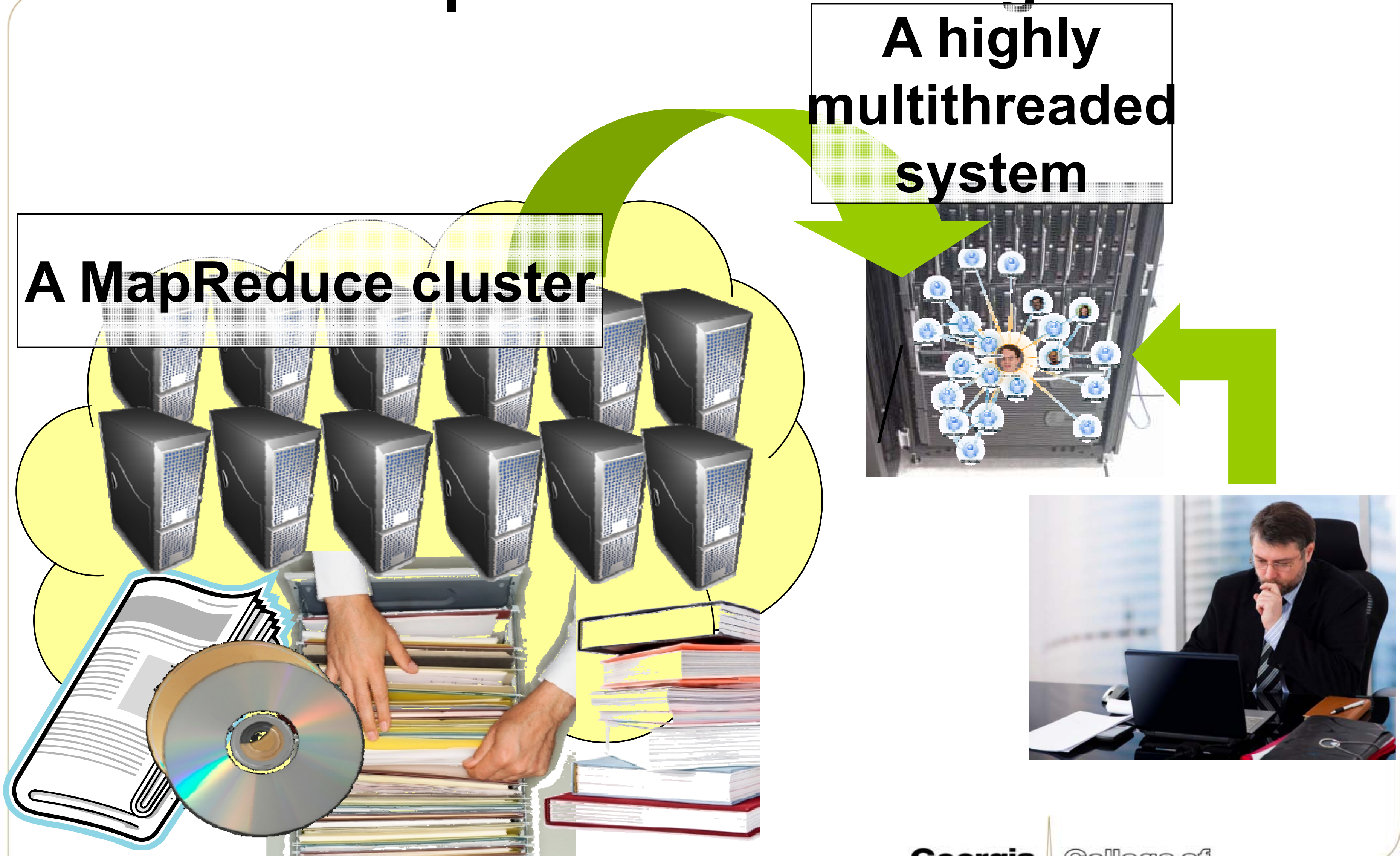
Presentation Outline

- Present the hybrid system.
- Solve the problem using three different systems: A MapReduce cluster, a highly multithreaded system, and the hybrid system.
- Show the effectiveness of the hybrid system by
  - Algorithm level analyses
  - System level analyses
  - Experimental results

Highlights

	A MapReduce cluster	A highly multithreaded system	A hybrid system of the two
Theory level analysis	Graph extraction: $W_{MapReduce}(n) \approx \theta(T^*(n))$ Shortest path: $W_{MapReduce}(n) > \theta(T^*(n))$	Work optimal	Effective if $\lvert T_{hmt} - T_{MapReduce} \rvert > n / BW_{inter}$
System level analysis	Bisection bandwidth and disk I/O overhead	Limited aggregate computing power, disk capacity, and I/O bandwidth	BW_{inter} is important.
Experi-ments	Five orders of magnitude slower than the highly multithreaded system in finding a shortest path	Incapable of storing the input graph	Efficient in solving the challenge problem.

A Hybrid System to Address the Distinct Computational Challenges

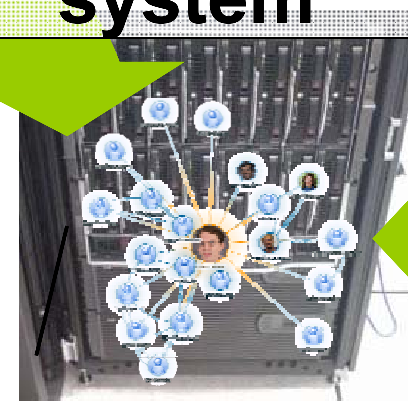


A Hybrid System to Address the Distinct Computational Challenges

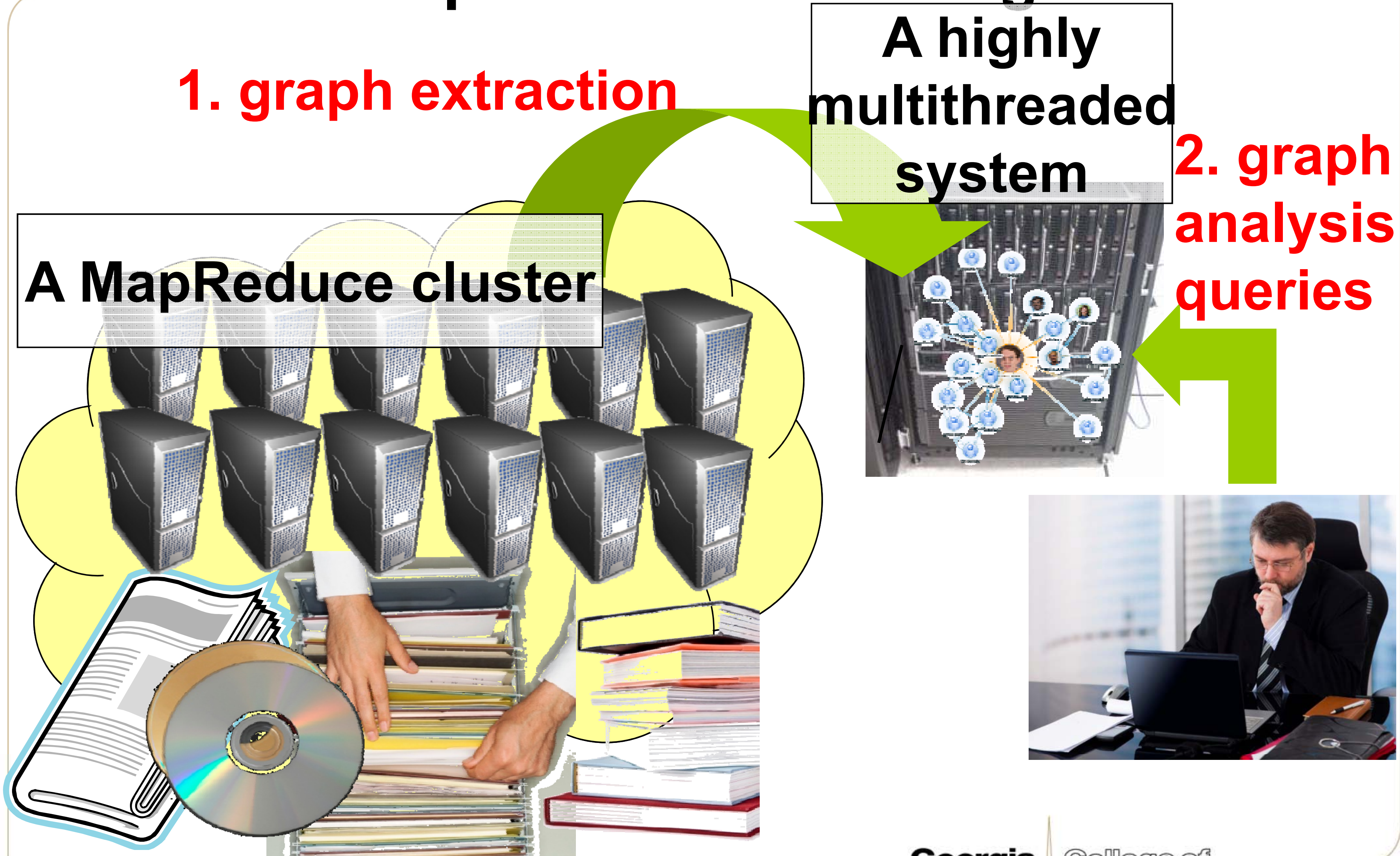
A highly multithreaded system

1. graph extraction

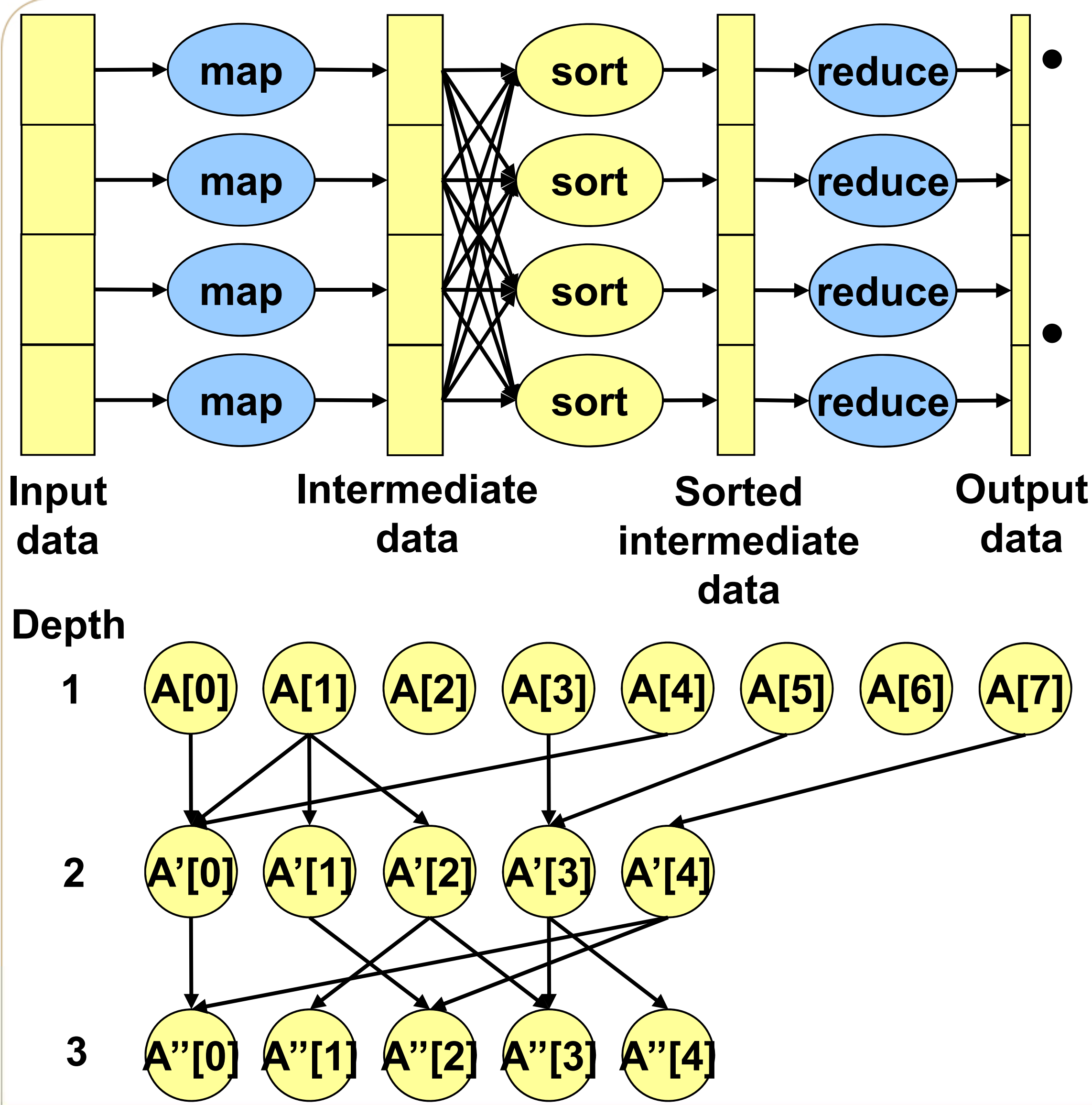
A MapReduce cluster



A Hybrid System to Address the Distinct Computational Challenges



The MapReduce Programming Model



- Scans the entire input data in the map phase.
- # MapReduce iterations = the depth of a directed acyclic graph (DAG) for MapReduce computation

Evaluating the efficiency of MapReduce Algorithms

- $W_{MapReduce} = \Sigma_{i = 1 \text{ to } k} (O(n_i \bullet (1 + f_i \bullet (1 + r_i)) + p_r \bullet \text{Sort}(n_i f_i / p_r))$
  - k: # MapReduce iterations.
  - n_i: the input data size for the ith iteration.
  - f_i: map output size / map input size
  - r_i: reduce output size / reduce input size.
  - p_r: # reducers
- Extracting a subgraph

  - $k = 1$ and $f_i << 1 \rightarrow W_{MapReduce}(n) \approx \theta(T^*(n))$, $T^*(n)$: the time complexity of the best sequential algorithm
- Finding a single-pair shortest path
  - $k = \lceil d/2 \rceil$, $f_i \approx 1 \rightarrow W_{MapReduce}(n) > \theta(T^*(n))$

Georgia Tech College of Computing

Evaluating the efficiency of MapReduce Algorithms

- $W_{MapReduce} = \Sigma_{i = 1 \text{ to } k} (O(n_i \cdot (1 + f_i \cdot (1 + r_i)) + p_r \cdot Sort(n_i f_i / p_r))$
  - k: # MapReduce iterations.
  - n_i: the input data size for the ith iteration.
  - f_i: map output size / map input size
  - r_i: reduce output size / reduce input size.
  - p_r: # reducers
- Extracting a subgraph

  - $k = 1$ and $f_i << 1 \rightarrow W_{MapReduce}(n) \approx \theta(T^*(n))$, $T^*(n)$: the time complexity of the best sequential algorithm
- Finding a single-pair shortest path
  - $k = \lceil d/2 \rceil$, $f_i \approx 1 \rightarrow W_{MapReduce}(n) > \theta(T^*(n))$

Georgia Tech | College of Computing

Evaluating the efficiency of MapReduce Algorithms

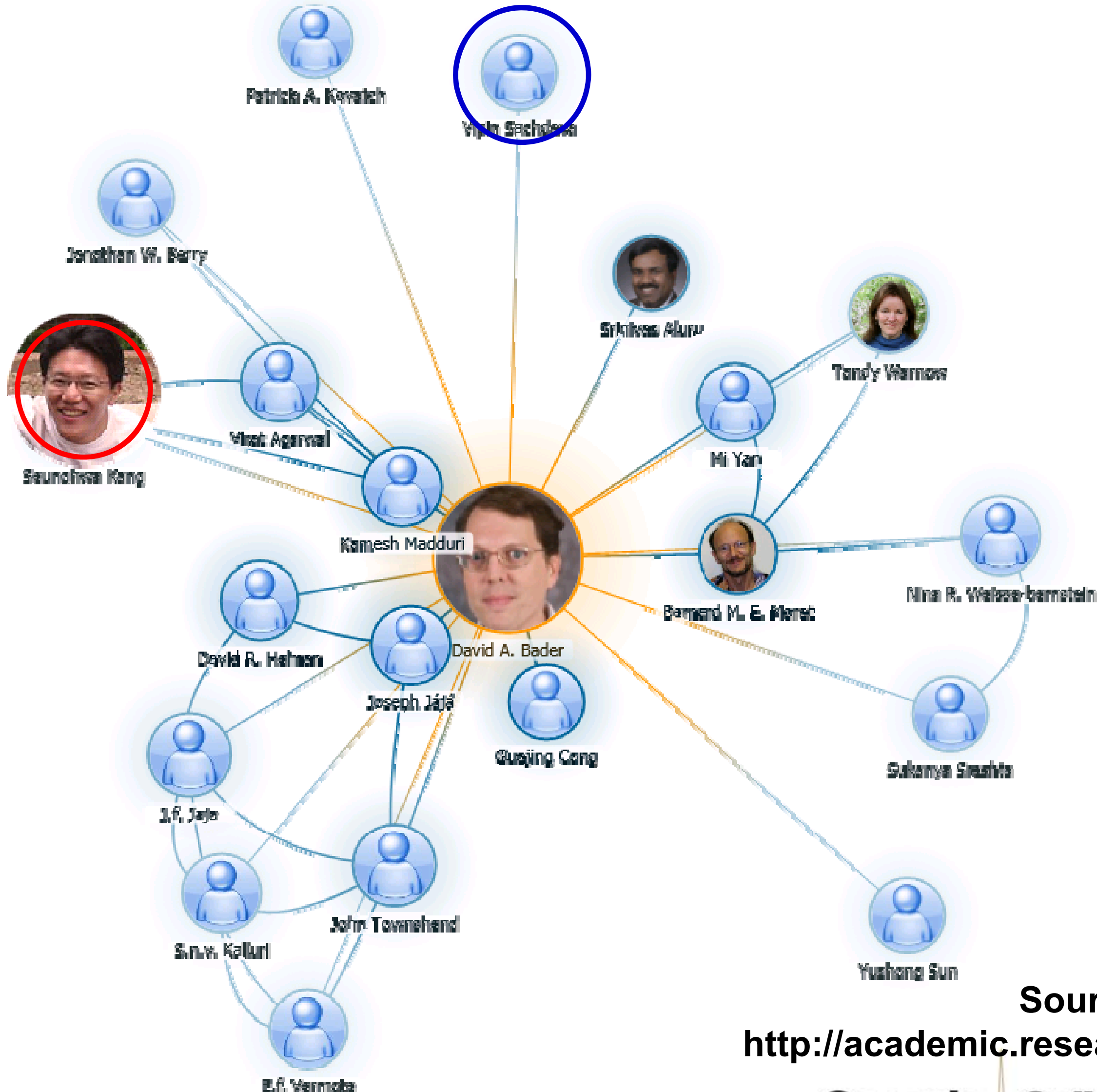
- $W_{MapReduce} = \Sigma_{i = 1 \text{ to } k} (O(n_i \bullet (1 + f_i \bullet (1 + r_i)) + p_r \bullet \text{Sort}(n_i f_i / p_r))$
  - k: # MapReduce iterations.
  - n_i: the input data size for the ith iteration.
  - f_i: map output size / map input size
  - r_i: reduce output size / reduce input size.
  - p_r: # reducers
- Extracting a subgraph
  - k = 1 and f_i << 1 → $W_{MapReduce}(n) \approx \theta(T^*(n))$, $T^*(n)$: the time complexity of the best sequential algorithm
- Finding a single-pair shortest path
  - k = ⌈ d/2 ⌉ , $f_i \approx 1$ → $W_{MapReduce}(n) > \theta(T^*(n))$

Georgia Tech | College of Computing

Bisection Bandwidth Requirements for a MapReduce Cluster

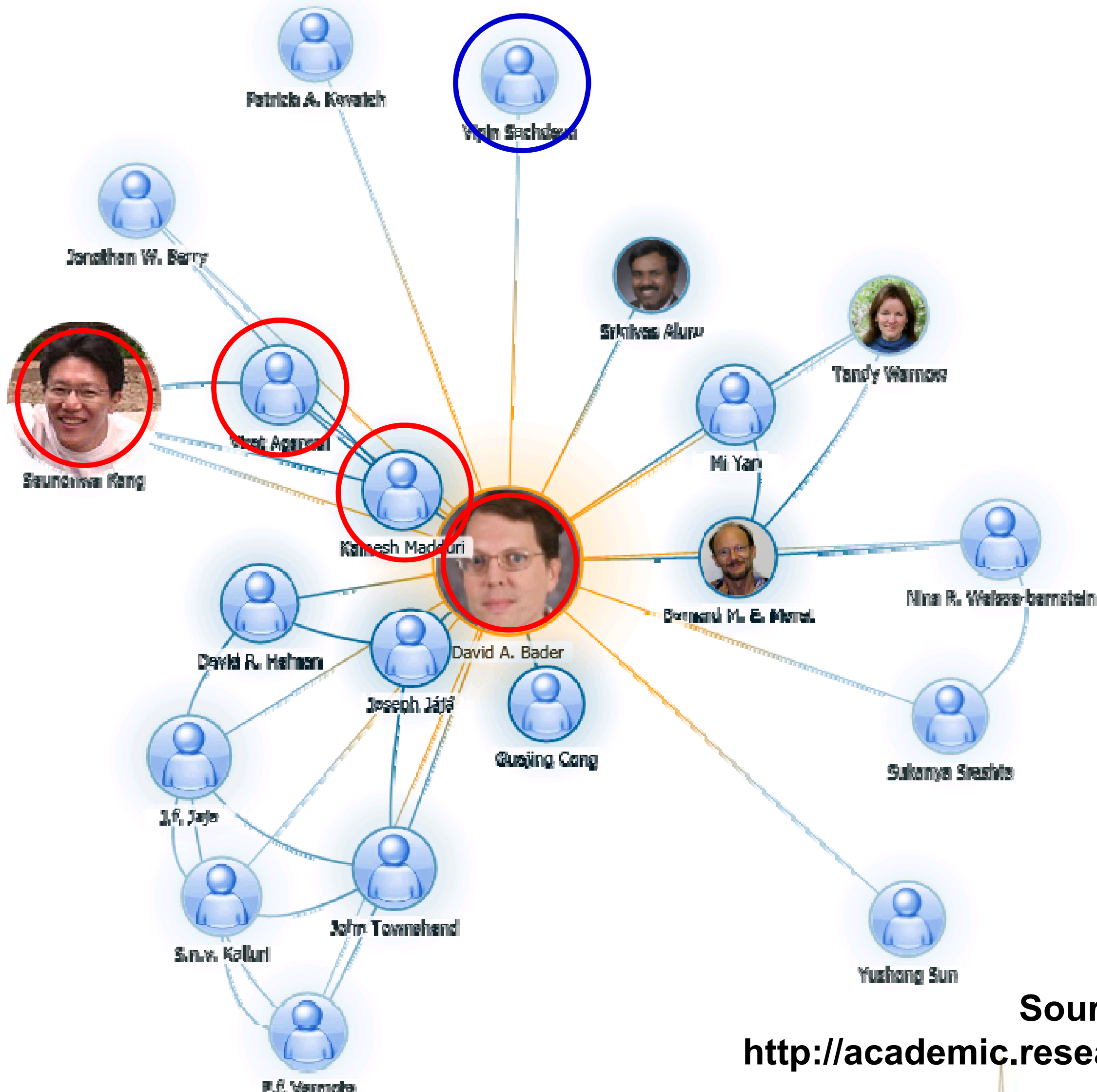
- The shuffle phase, which requires inter-node communication, can be overlapped with the map phase.
- If $T_{map} > T_{shuffle}$, $T_{shuffle}$ does not affect the overall execution time.
  - T_{map} scales trivially.
  - To scale $T_{shuffle}$ linearly, bisection bandwidth also needs to scale in proportion to a number of nodes. Yet, the cost to linearly scale bisection bandwidth increases super-linearly.
  - If $f \ll 1$, the sub-linear scaling of $T_{shuffle}$ does not increase the overall execution time.
  - If $f \approx 1$, it increases the overall execution time.

A single-pair shortest path



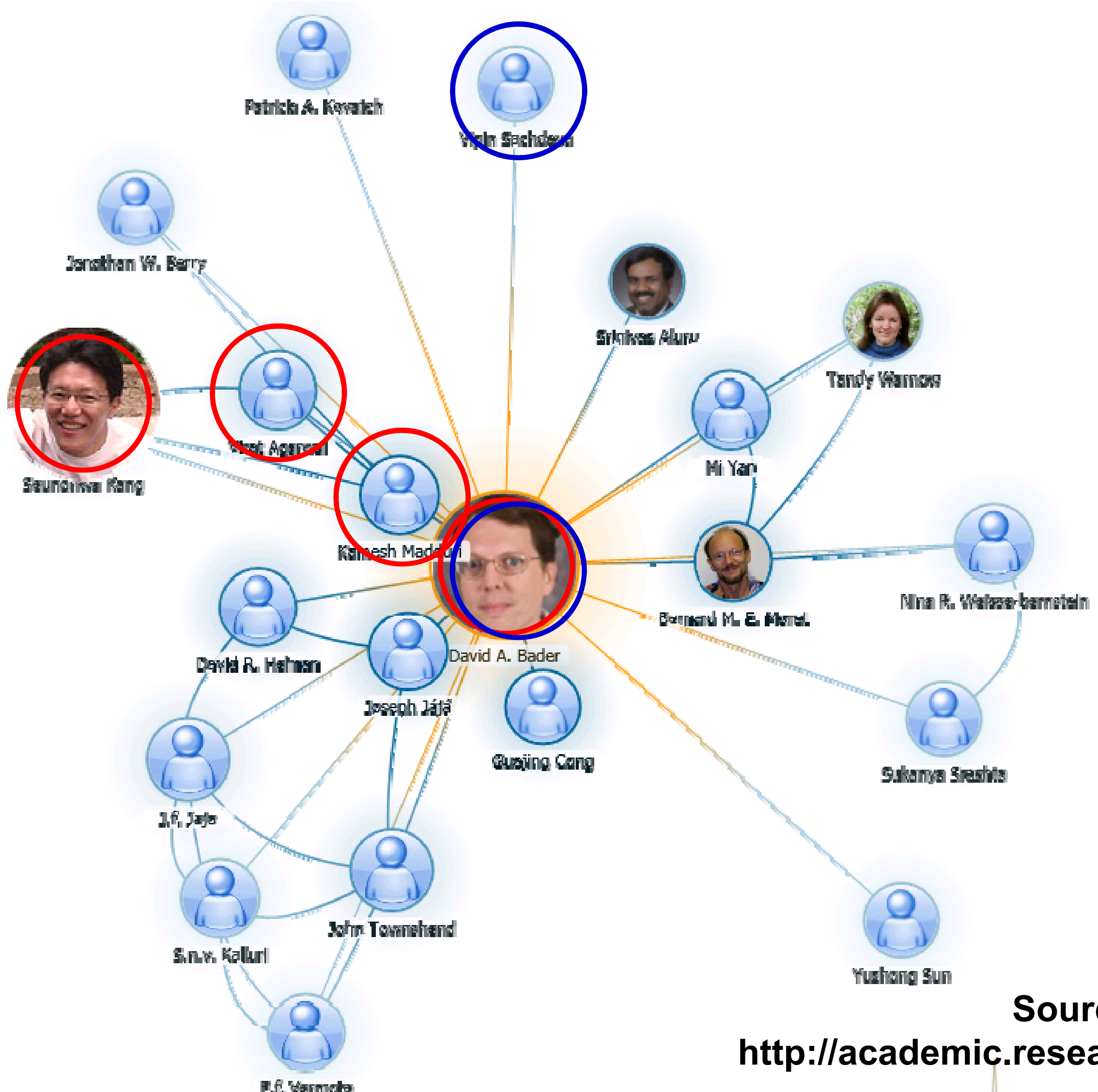
Source:
http://academic.research.microsoft.com

A single-pair shortest path



Source: http://academic.research.microsoft.com

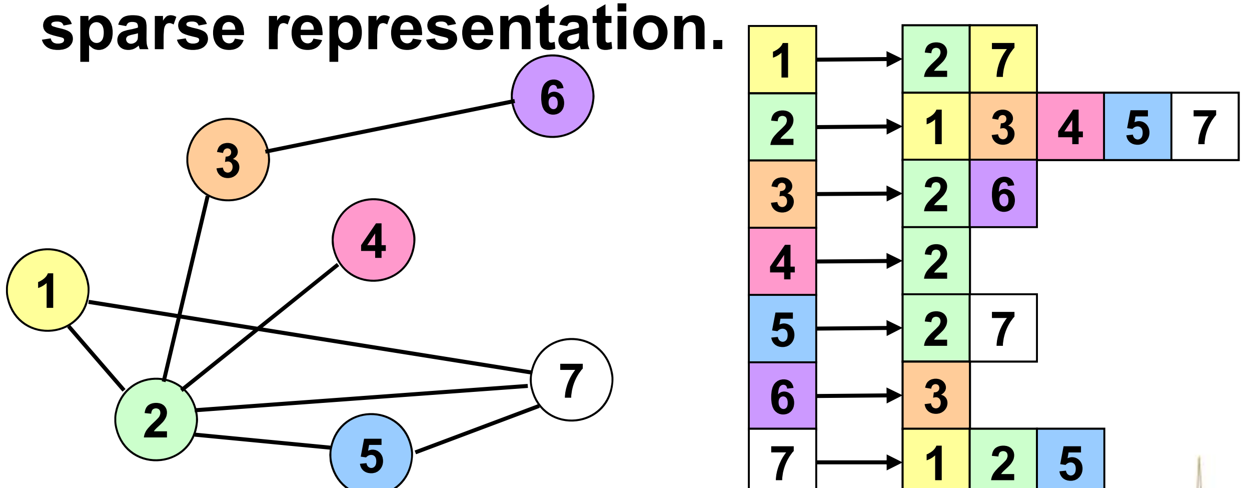
A single-pair shortest path



Source: http://academic.research.microsoft.com

Disk I/O overhead

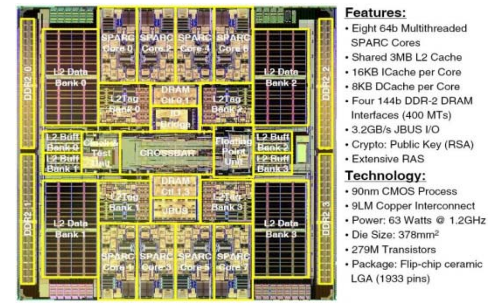
- Disk I/O overhead is unavoidable if the size of data overflows the main memory capacity.
- Raw data can be very large.
- Extracted graphs are much smaller.
  - The Facebook network: 400 million users × 130 friends per user → less than 256 GB using the sparse representation.



A Highly Multithreaded System w/ the Shared Memory Programming Model

- **Provide a random access mechanism.**
- **In SMPs, non-contiguous accesses are expensive.***
- **Multithreading tolerates memory access latency.+**
- **There is a work optimal parallel algorithm to find a single-pair shortest path.**

Sun Fire T2000 (Niagara)



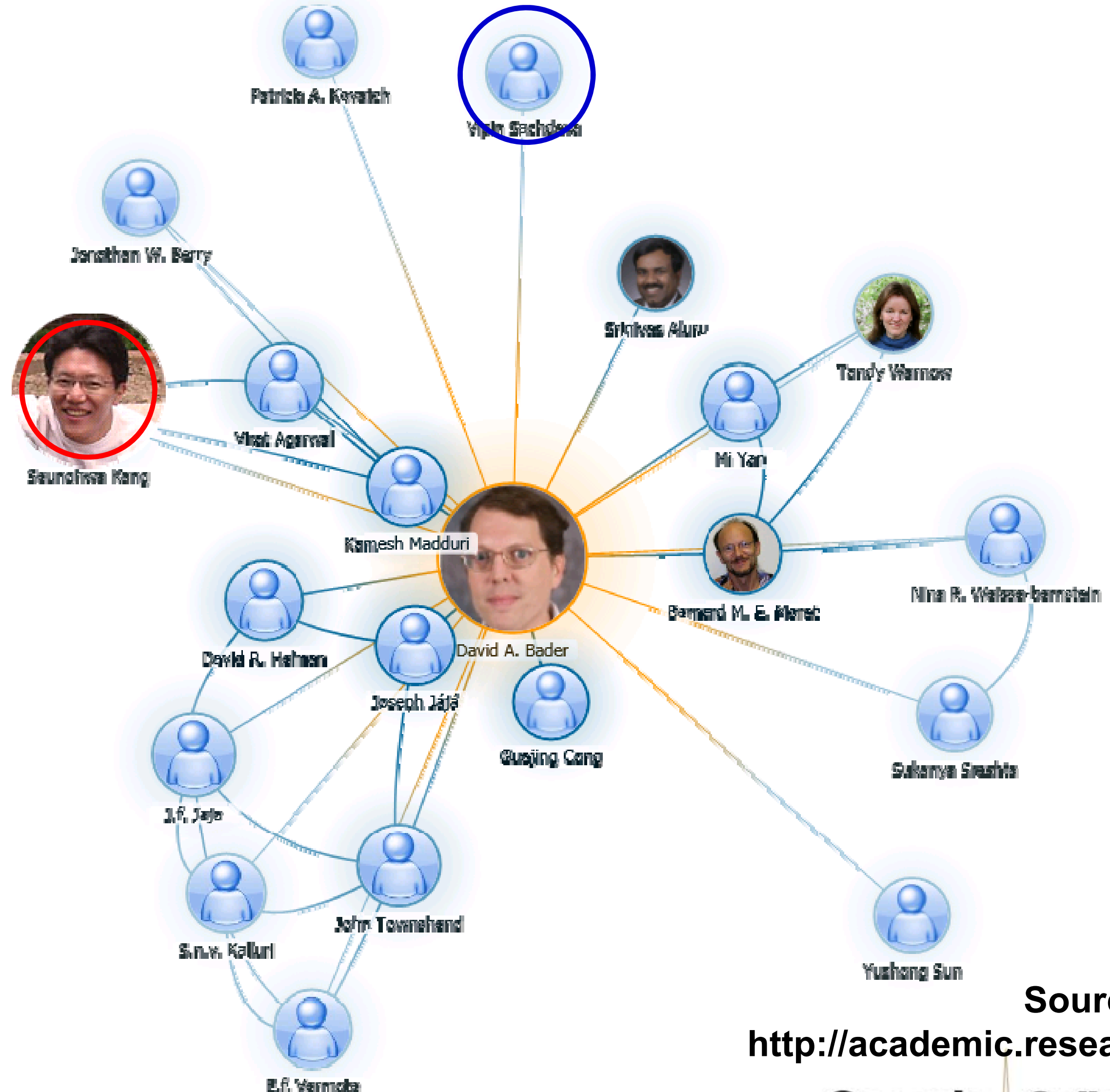
Source: Sun Microsystems

Cray XMT

Source: Cray

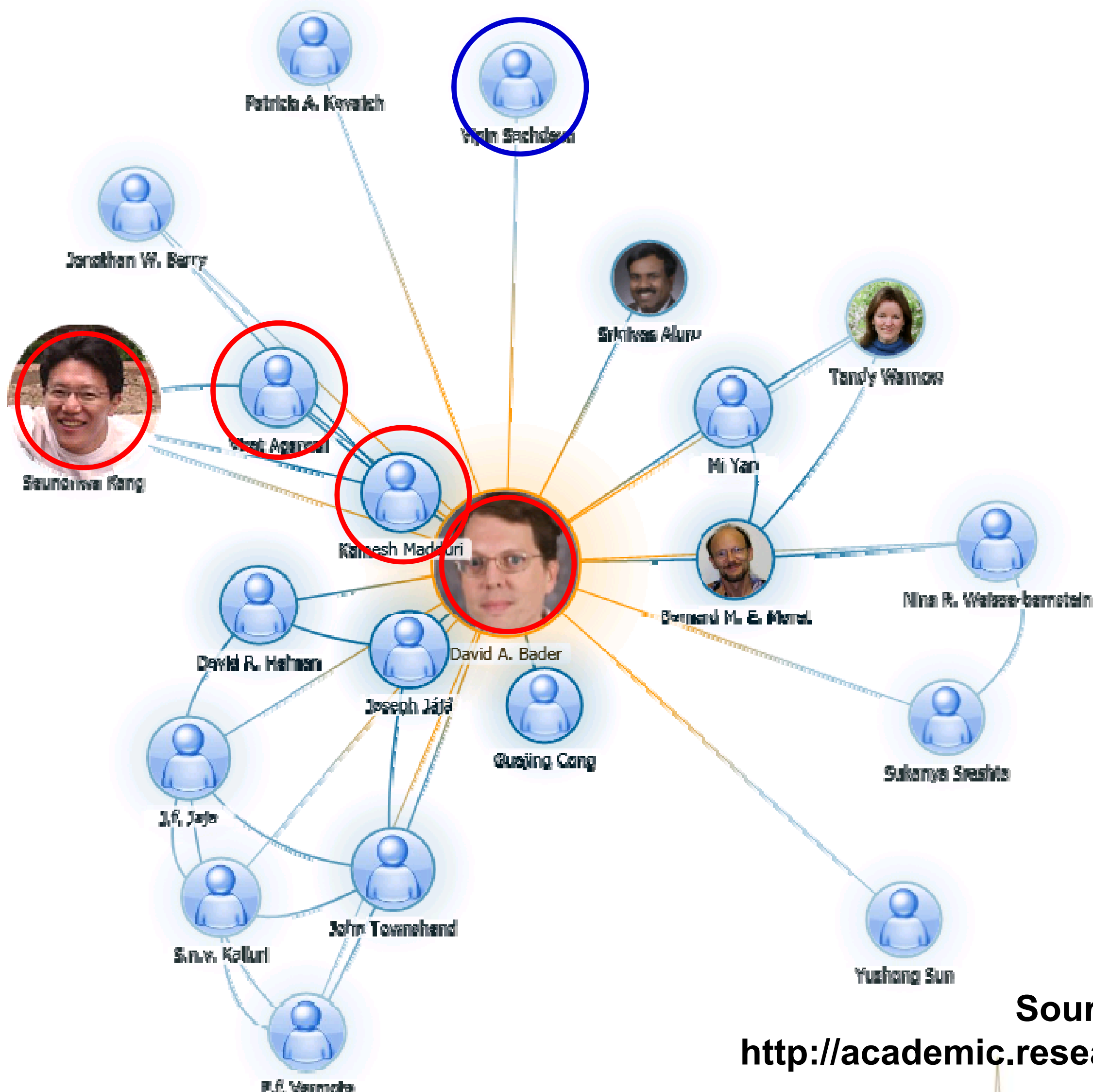
* D. R. Helman and J. Ja'Ja', "Prefix computations on symmetric multiprocessors," J. of parallel and distributed computing, 61(2), 2001.
+ D. A. Bader, V. Kanade, and K. Madduri, "SWARM: A parallel programming framework for multi-core processors," Workshop on Multithreaded Architectures and Applications, 2007.

A single-pair shortest path



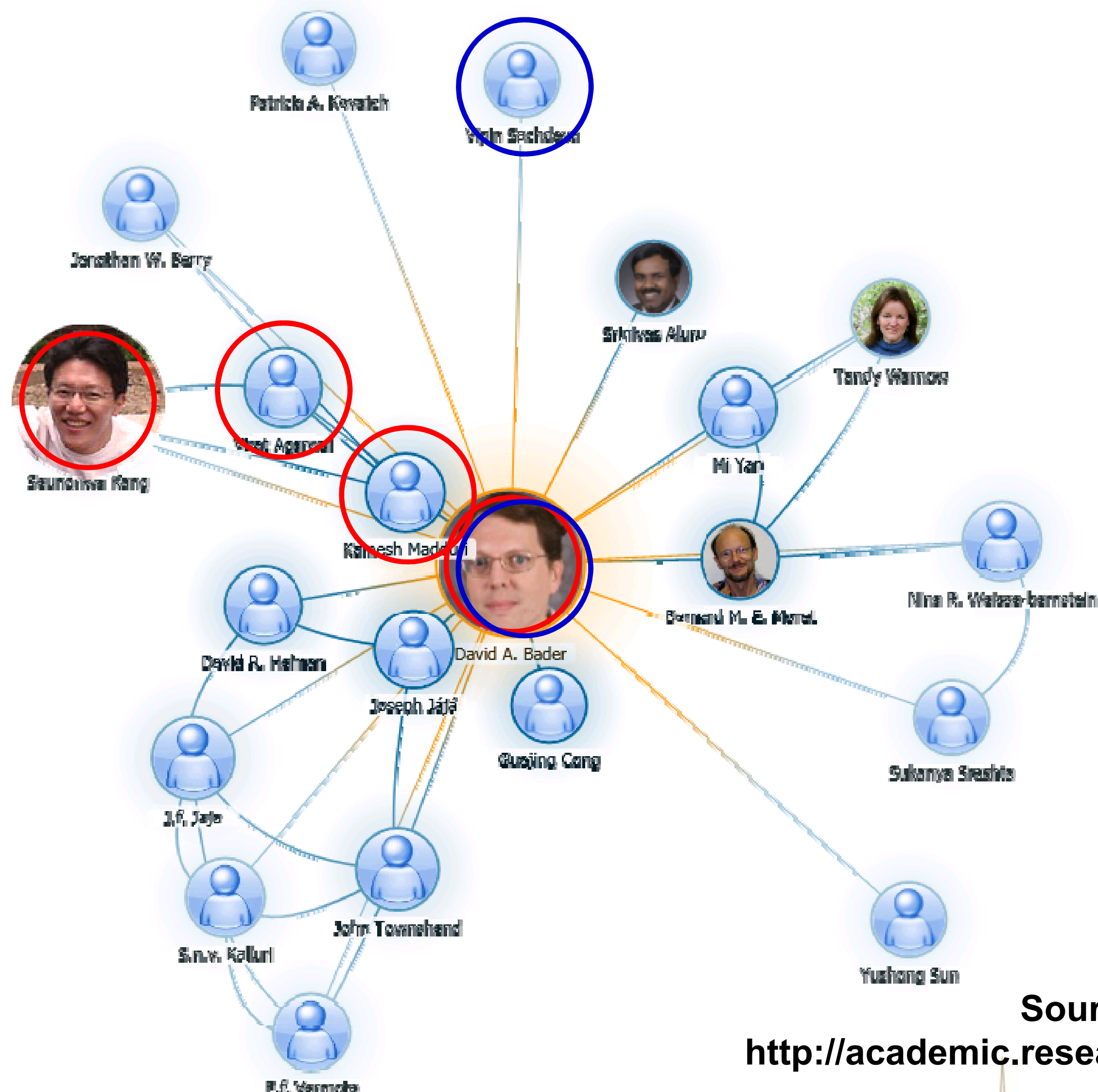
Source: http://academic.research.microsoft.com

A single-pair shortest path



Source: http://academic.research.microsoft.com

A single-pair shortest path



Source: http://academic.research.microsoft.com

Low Latency High Bisection Bandwidth Interconnection Network

- **Latency increases as the size of a system increases.**
  - A larger number of threads and additional parallelism are required as latency increases.
- **Network cost to linearly scale bisection bandwidth increases super-linearly.**
  - But not too expensive for a small number of nodes.
- **These limit the size of a system.**
  - Reveal limitations in extracting a subgraph from a very large graph.

The Time Complexity of an Algorithm on the Hybrid System

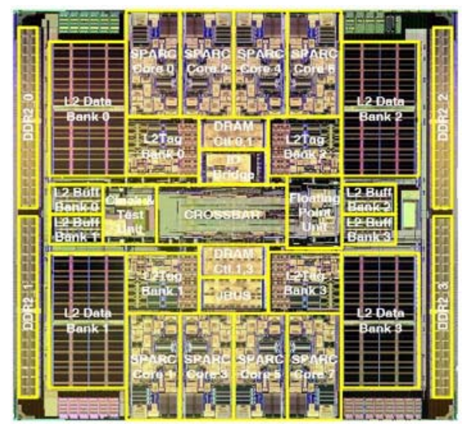
- $T_{hybrid} = \Sigma_{i = 1 \text{ to } k} \min(T_{i, MapReduce} + \Delta, T_{i, hmt} + \Delta)$
  - k: # steps
  - $T_{i, MapReduce}$ and $T_{i, hmt}$: time complexities of the i_{th} step on a MapReduce cluster and a highly multithreaded system, respectively.
  - Δ: $n_i / BW_{inter} \times \delta(i – 1, i)$,
  - n_i : the input data size for the i_{th} step.
  - BW_{inter}: the bandwidth between a MapReduce cluster and a highly multithreaded system.
  - $\delta(i – 1, i)$: 0 if selected platforms for the $i - 1_{th}$ and i_{th} steps are same. 1, otherwise.

Test Platforms

- A MapReduce cluster
  - 4 nodes
  - 4 dual core 2.4 GHz Opteron processors and 8 GB main memory per node.
  - 96 disks (1 TB per disk).
- A highly multithreaded system
  - A single socket UltraSparc T2 1.2 GHz processor (8 core, 64 threads).
  - 32 GB main memory.
  - 2 disks (145 GB per disk)
- A hybrid system of the two

Source: http://hadoop.apache.org/

Sun Fire T2000 (Niagara)

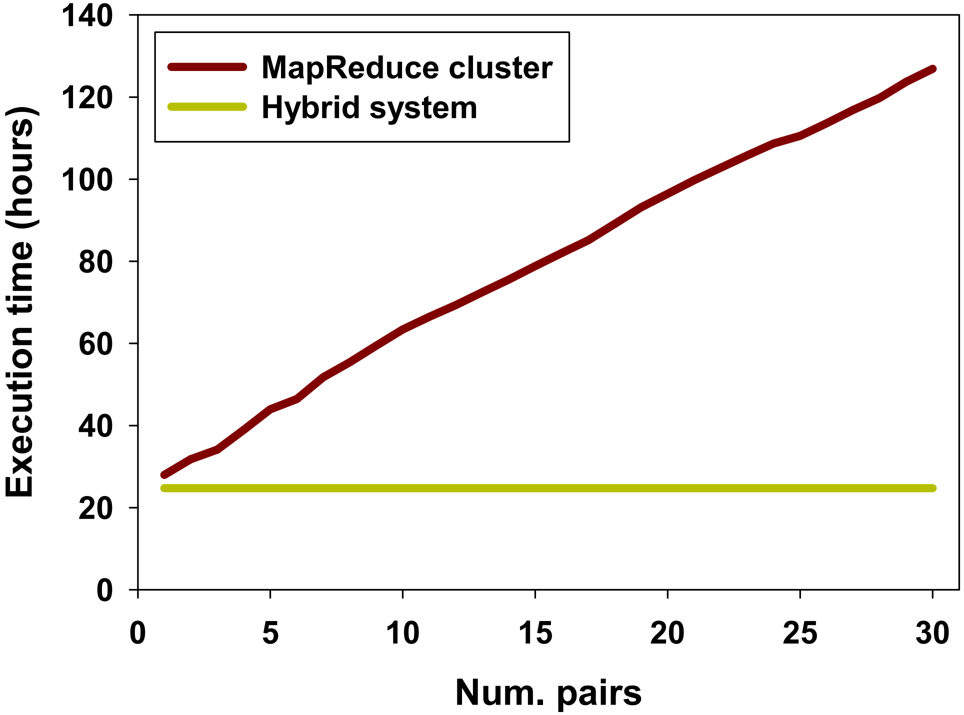


Features:
- Eight 64b Multithreaded SPARC Cores
- Shared 3MB L2 Cache
- 16KB ICache per Core
- 8KB DCache per Core
- Four 144b DDR-2 DRAM Interfaces (400 MTs)
- 3.2GB/s JBUS I/O
- Crypto: Public Key (RSA)
- Extensive RAS

Technology:
- 90nm CMOS Process
- 9LM Copper Interconnect
- Power: 63 Watts @ 1.2GHz
- Die Size: 378mm^2
- 279M Transistors
- Package: Flip-chip ceramic LGA (1933 pins)

Source: Sun Microsystems

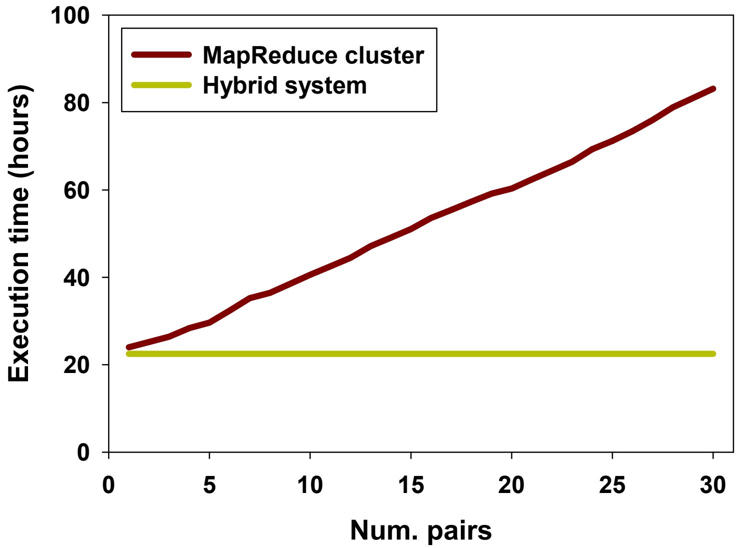
A subgraph that covers 10% of the input graph

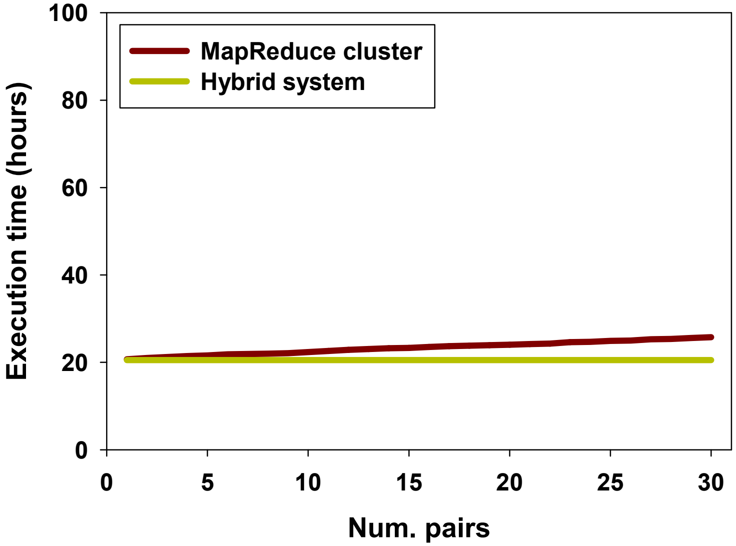


	MapReduce	Hybrid
Subgraph extraction	24	24
Memory loading	-	0.83
Finding a shortest path (for 30 pairs)	103	0.00073

Once the subgraph is loaded into the memory, the hybrid system analyzes the subgraph five orders of magnitude faster than the MapReduce cluster (103 hours vs 2.6 seconds).

Subgraphs that cover 5% (left) and 2% (right) of the input graph





	MapReduce	Hybrid
Subgraph extraction	22	22
Memory loading	-	0.42
Finding a shortest path (for 30 pairs)	61	0.00047

	MapReduce	Hybrid
Subgraph extraction	21	21
Memory loading	-	0.038
Finding a shortest path (for 30 pairs)	5.2	0.00019

Conclusions

- We identified the key computational challenges in large-scale complex network analysis problems.
- Our hybrid system effectively addresses the challenges by using a right tool in a right place in a synergistic way.
- Our work showcases a holistic approach to solve real-world challenges.

Acknowledgment of Support

NSF
National Institutes of Health
Department of Energy • United States of America
NASA
Department of Defense • United States of America

DARPA
Oak Ridge National Laboratory
Pacific Northwest National Laboratory
Sandia National Laboratories

IBM®
Microsoft® Research
nVIDIA.
SONY

CRAY
Sun microsystems
XILINX®
TOSHIBA