DASFAA 2011 Panel on
Challenges in Managing and Mining Large, Heterogeneous Data

# NoSQL vs. Parallel DBMS for Large-scale Data Management

Apr. 24th, 2011

**Kyu-Young Whang**

KAIST Distinguished Professor, ACM/IEEE Fellow
Department of Computer Science
KAIST

# Contents

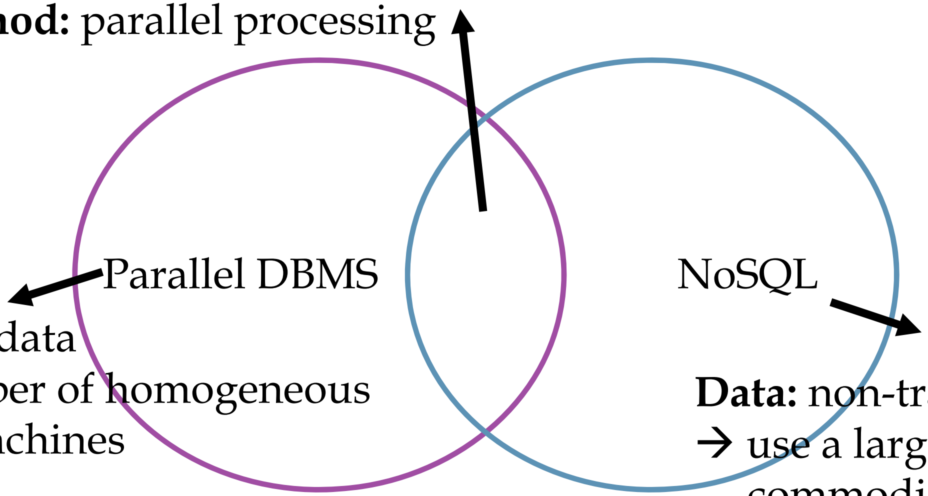
- NoSQL Systems and Parallel DBMSs
  - Comparison of NoSQL Systems and Parallel DBMSs
  - Map of NoSQL Systems
  - Research Challenges
- Projects at KAIST
  - ODYS: a Massively-Parallel Search Engine
  - Odysseus/DFS: a Relational DBMS on Top of HDFS

# NoSQL vs. Parallel DBMS

- NoSQL systems
  - Description
    - "Non-relational, distributed data stores that often did not attempt to provide ACID guarantees" [Wik11]
    - e.g., GFS, BigTable, MapReduce

- Parallel DBMSs
  - Description
    - "Systems attempt to exploit recent multiprocessor computer architectures in order to build a high-performance and high-availability database server" [Val93]
  - Classification
    - Shared memory architecture
    - Shared disk architecture
    - Shared nothing architecture

**Common Goal:** handling large-scale data management and processing
**Method:** parallel processing



Parallel DBMS

NoSQL

**Data:** transactional data
→ use a small number of homogeneous and powerful machines

**Data:** non-transactional data
→ use a large number of heterogeneous commodity machines

| data type | characteristics | consistency requirement | relevant strategy |
|---|---|---|---|
| transactional data | We assume relationship exists among items<br>An operation involves multiple data items | two-phase commit | a parallel DBMS with a small number of machines |
| non-transactional data | We assume no relationship among data items | eventual consistency | a NoSQL system with a large number of machines |

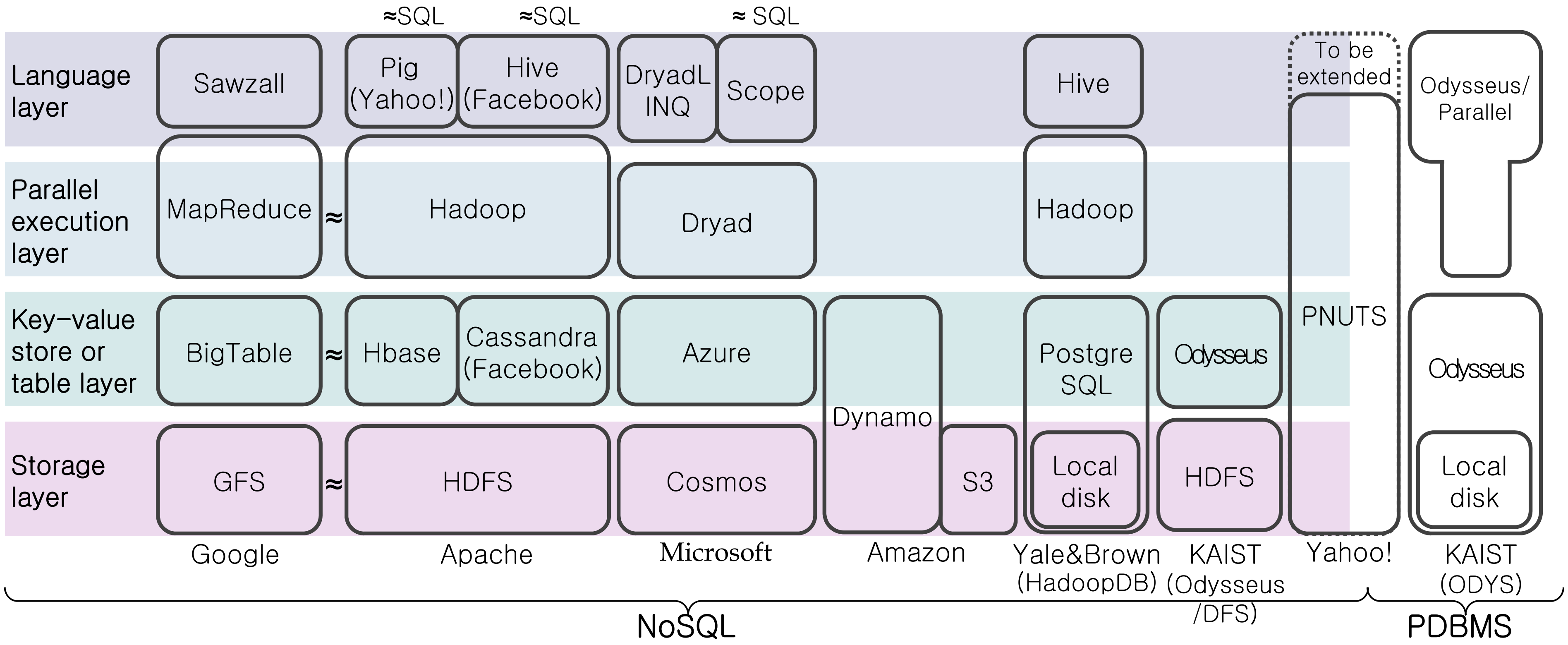
# NoSQL Systems vs. Parallel DBMSs

- NoSQL systems (e.g., Hadoop[Had])
  - Advantages
    - highly scalable
    - highly fault tolerant
    - inexpensive
    - easy to setup and use
  - Disadvantages
    - Weak functionalities
      - SQL
      - schemas
      - Indexes
      - query optimization
      - transactions

- Parallel DBMSs (e.g., Vertica[Ver])
  - Advantages
    - Strong functionalities
      - SQL
      - schemas
      - indexes
      - query optimization
      - transactions
  - Disadvantages
    - difficult to scale
    - expensive
    - not suitable where faults occur frequently
    - hard to setup and use

# Map of NoSQL systems

- Layers of NoSQL
  - Storage layer: replicated distributed storage for large-scale data
  - Key-value store or table layer: data storage storing data in the form of key-value pairs or tables
  - Parallel execution layer: parallel processing systems
  - Language layer: query interfaces



| | Google | Apache | | Microsoft | | Amazon | | Yale&Brown (HadoopDB) | KAIST (Odysseus /DFS) | Yahoo! | KAIST (ODYS) |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Language layer | Sawzall | ≈SQL<br>Pig (Yahoo!) | ≈SQL<br>Hive (Facebook) | DryadLINQ | ≈ SQL<br>Scope | | | Hive | | To be extended | Odysseus/ Parallel |
| Parallel execution layer | MapReduce ≈ | Hadoop | | Dryad | | | | Hadoop | | PNUTS | |
| Key-value store or table layer | BigTable ≈ | Hbase | Cassandra (Facebook) | Azure | | Dynamo | | Postgre SQL | Odysseus | | Odysseus |
| Storage layer | GFS ≈ | HDFS | | Cosmos | | | S3 | Local disk | HDFS | | Local disk |

<Map of NoSQL systems> (modified & extended from [Bud09])

# Research Challenges

- [Goal] Building large-scale systems that have the best of both worlds, i.e., high scalability, fault tolerance, and rich functionality on cheap hardware
- [NoSQL ➔ PDBMS] Supporting DBMS features including SQL, schemas, indexes, query optimization, and transactions in NoSQL systems
  - Language layers
    - DryadLINQ [YIF+08], Hive [TSJ+09], Pig [ORS+08], Scope [CJL+08]
  - Join, iteration [DQJ+10] [WSS+10] [VCL10] [YDHP07] [BHBE10]
- [PDBMS ➔ NoSQL] Achieving high scalability and high fault tolerance in Parallel DBMSs
  - HadoopDB [ABA+09]
  - GreenPlum [Waa09]
  - PNUTS [CRS+08]
  - NoSQL-style fault tolerance [YYTM10]
  - ODYS – a parallel DBMS with limited functionality (shared nothing) [Wha09] (KAIST)
- Supporting random read and write operations in append-only distributed file systems
  - BigTable [CDG+06](Google), HBase[Had] (open source), Megastore[FKL+08] (Google)
  - Odysseus/DFS: a relational DBMS on top of the distributed file system (HDFS) [Kan11] (KAIST)

# Projects at KAIST

- ODYS: a Massively-Parallel Search Engine [Wha09]
  - Building a massively-parallel DBMS using a DB-IR tightly-integrated DBMS can be an attractive alternative to a specialized search engine
    - A parallel DBMS with limited functionality
      - limited join
      - single-node transactions
    - Based on DB-IR tight integrated DBMS
    - Performance comparable to or better than those of large-scale commercial search engines
    - Scalability
    - A massively-parallel configuration possible (e.g., 300 nodes for indexing 30 billion Web pages)

- Odysseus/DFS: a Relational DBMS on Top of HDFS [Kan11]
  - Integrating a general-purpose relational DBMS rather than a key-value store (e.g., BigTable, Hbase) on top of a distributed file system (e.g., GFS, HDFS)
    - Comparable to BigTable
      - high scalability, fault tolerance, and load balancing of DFS
      - can be driven by MapReduce
    - Additional to BigTable
      - all the functionalities of the relational DBMS such as SQL, schemas, and indexes
    - Different from BigTable
      - relational table compared to key-value store

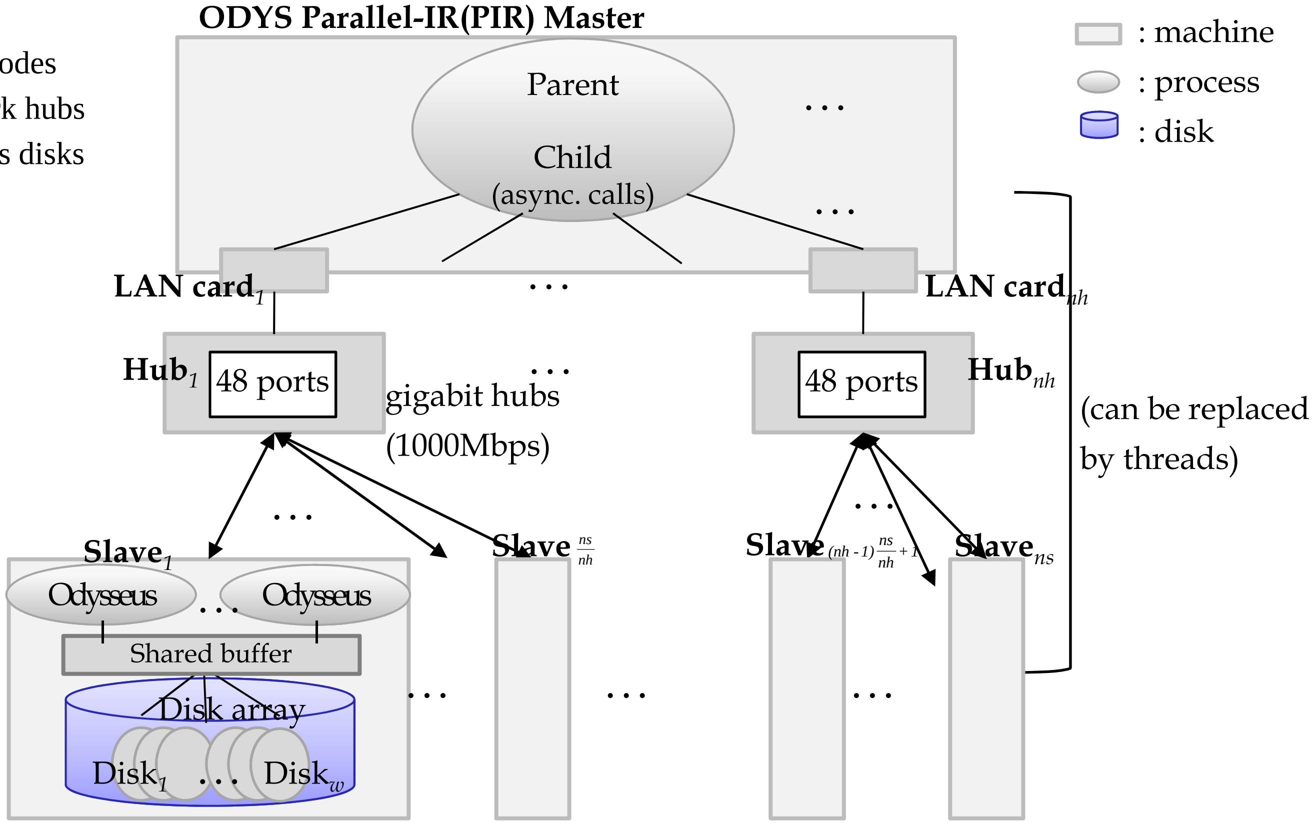
# A Massively-parallel Search Engine

- Building a massively-parallel search engine using a DB-IR tightly-integrated DBMS can be an attractive alternative to a specialized large-scale search engine such as Google
  - Efficiency : tight-coupling of DB and IR
  - Scalability: a massively-parallel configuration possible

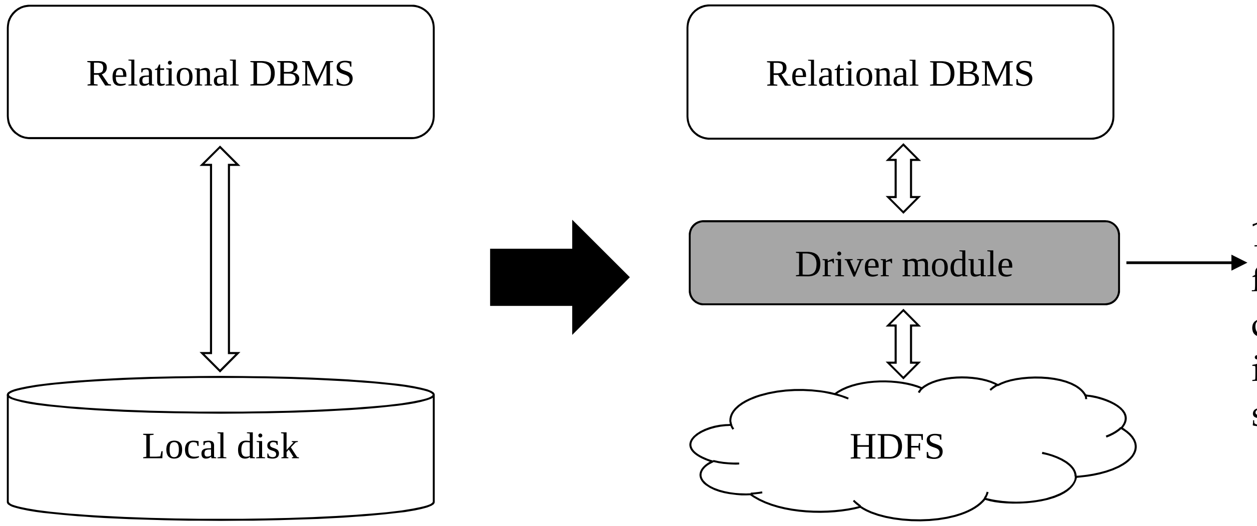
*ns*: the number of slave nodes

*nh*: the number of network hubs

*w*: the number of a slave's disks



# Odysseus/DFS: A Relational DBMS on top of HDFS

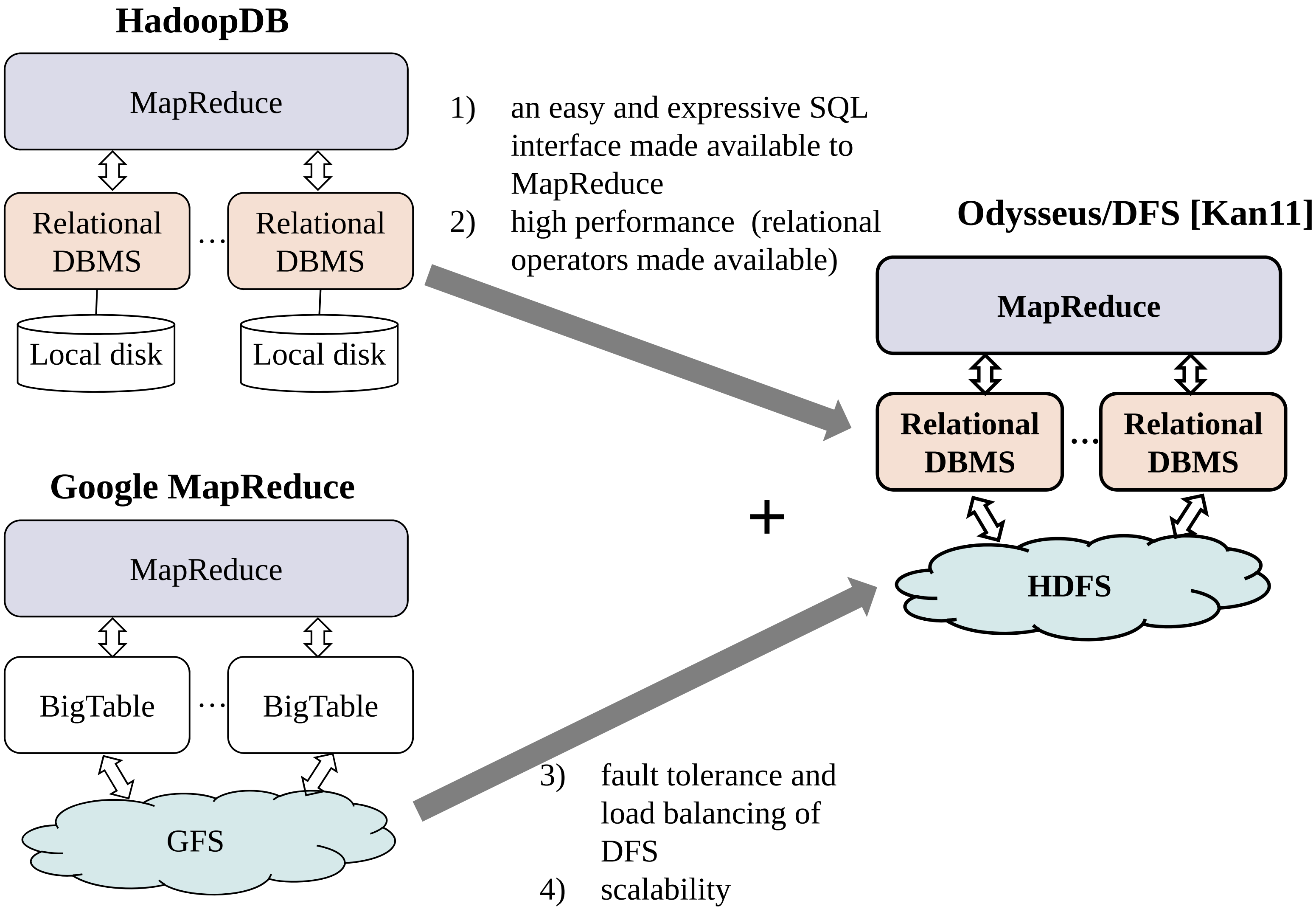


A relational DBMS on a local disk

A relational DBMS on HDFS

1. Make up for low functionality of HDFS compared to that of an O/S file, i.e., random as well as sequential read/write/update

# Parallelization of Architecture



**HadoopDB**

MapReduce

Relational DBMS … Relational DBMS

Local disk | Local disk

1) an easy and expressive SQL interface made available to MapReduce
2) high performance (relational operators made available)

**Google MapReduce**

MapReduce

BigTable … BigTable

GFS

+

**Odysseus/DFS [Kan11]**

**MapReduce**

**Relational DBMS** … **Relational DBMS**

**HDFS**

3) fault tolerance and load balancing of DFS
4) scalability
5) sharability

# Issues

No SQL vs. Parallel DBMS

- Best of both worlds

- What and How?

# References

- **[AAB+05]** Serge Abiteboul, Rakesh Agrawal, Phil Bernstein, Mike Carey, Stefano Ceri, Bruce Croft, David DeWitt, Mike Franklin, Hector Garcia Molina, Dieter Gawlick, Jim Gray, Laura Haas, Alon Halevy, Joe Hellerstein, Yannis Ioannidis, Martin Kersten, Michael Pazzani, Mike Lesk, David Maier, Jeff Naughton, Hans Schek, Timos Sellis, Avi Silberschatz, Mike Stonebraker, Rick Snodgrass, Jeff Ullman, Gerhard Weikum, Jennifer Widom, and Stan Zdonik, “The Lowell Database Research Self-Assessment,” *Comm. of ACM*, Vol. 48, No. 5, pp. 111-118, May 2005.

- **[ABA+09]** Azza Abouzeid, Kamil Bajda-Pawlikowski, Daniel Abadi, Alexander Rasin, Avi Silberschatz, "HadoopDB: An Architectural Hybrid of MapReduce and DBMS Technologies for Analytical Workloads," In *Proc. 35th Int'l Conf. on Very Large Data Bases (VLDB)*, Aug. 2009.

- **[BHBE10]** Yingyi Bu, Bill Howe, Magdalena Balazinska, and Michael Ernst, "HaLoop: Efficient Iterative Data Processing on Large Clusters," In *Proc. 36th Int'l Conf. on Very Large Data Bases (VLDB)*, Sept. 2010.

- **[BR99]** Ricardo A. Baeza-Yates and Berthier Ribeiro-Neto, *Modern Information Retrieval*, ACM Press, Addison-Wesley, 1999.

- **[Bud09]** Mihai Budiu, presentation slides, 2009. available at http://budiu.info/work/dryad-talk-berkeley09.pptx.

- **[CDG+06]** Fay Chang, Jeffrey Dean, Sanjay Ghemawat, Wilson C. Hsieh, Deborah A. Wallach, Mike Burrows, Tushar Chandra, Andrew Fikes, and Robert E. Gruber, “BigTable: A Distributed Storage System for Structured Data,” In *Proc. 6th Symposium on Operating Systems Design and Implementation (OSDI ’06)*, pp. 205-218, Dec. 2006.

- **[CJL+08]** Ronnie Chaiken, Bob Jenkins, Paul Larson, Bill Ramsey, Darren Shakib, Simon Weaver, and Jingren Zhou, "SCOPE: Easy and Efficient Parallel Processing of Massive Data Sets," In *Proc. 34th Int'l Conf. on Very Large Data Bases (VLDB)*, Aug. 2008.

- **[CRS+08]** Brian Cooper, Raghu Ramakrishnan, Utkarsh Srivastava, Adam Silberstein, Phil Bohannon, Hans-Arno Jacobsen, Nick Puz, Daniel Weaver, and Ramana Yerneni, "PNUTS: Yahoo!'s Hosted Data Serving Platform," In *Proc. 34th Int'l Conf. on Very Large Data Bases (VLDB)*, Aug. 2008.

- **[CRW05]** Surajit Chaudhuri, Raghu Ramakrishnan, and Gerhard Weikum, “Integrating DB and IR Technologies: What is the Sound of One Hand Clapping?,” In *Proc. 2nd Biennial Conf. on Innovative Data Systems Research*, Asilomar, California, pp. 1-12, Jan. 2005.

- **[DQJ+10]** Jens Dittrich, Jorge Quiane, Alekh Jindal, Yagiz Kargin, Vinay Setty, and Jörg Schad, "Hadoop++: Making a Yellow Elephant Run Like a Cheetah (Without It Even Noticing)," In *Proc. 36th Int'l Conf. on Very Large Data Bases (VLDB)*, Sept. 2010.

- **[FKL+08]** JJ Furman, Jonas S Karlsson, Jean-Michel Leon, Alex Lloyd, Steve Newman, and Philip Zeyliger, "Megastore: A Scalable Data System for User Facing Applications," *In 2008 ACM SIGMOD Int'l Conf. on Management of Data*, June 2008.

- **[GGL03]** Sanjay Ghemawat, Howard Gobioff, and Shun-Tak Leung, "The Google file system," In *Proc. 19th ACM Symposium on Operating Systems Principles*, pp. 29-43, Oct. 2003.

- **[Had]** Hadoop, http://hadoop.apache.org.

- **[Kan11]** Joon-Young Kang, Odysseus/DFS: Design and Implementation of a Driver Module for the Relational DBMS Based on the Distributed File System (in Korean), Master's Thesis, Department of Computer Science, KAIST, Feb. 2011.

- **[ORS+08]** Christopher Olston, Benjamin Reed, Utkarsh Srivastava, Ravi Kumar, and Andrew Tomkins, "Pig latin: a not-so-foreign language for data processing," *In 2008 ACM SIGMOD Int'l Conf. on Management of Data*, June 2008.

- **[TSJ+09]** Ashish Thusoo, Joydeep Sen Sarma, Namit Jain, Zheng Shao, Prasad Chakka, Suresh Anthony, Hao Liu, Pete Wyckoff, and Raghotham Murthy, "Hive - A Warehousing Solution Over a Map-Reduce Framework," In *Proc. 35th Int'l Conf. on Very Large Data Bases (VLDB)*, Aug. 2009.

- **[Val93]** Patrick Valduriez, “Parallel database systems: Open problems and new issues,” *Distributed and Parallel Databases*, Vol. 1, No. 2, pp. 137-165, 1993.

- **[VCL10]** Rares Vernica, Michael J. Carey, and Chen Li, "Efficient parallel set-similarity joins using MapReduce," *In 2010 ACM SIGMOD Int'l Conf. on Management of Data*, June 2010.

- **[Ver]** Vertica, http://www.vertica.com.

- **[Waa09]** Florian M. Waas, “Beyond Conventional Data Warehousing — Massively Parallel Data Processing with Greenplum Database (Invited Talk),” In Book *Business Intelligence for the Real-Time Enterprise*, Springer, Vol. 27, pp. 89-96, 2009.

- **[Weik07]** Gerhard Weikum, “DB&IR: both sides now,” In Proc. *2007 ACM SIGMOD Int’l Conf. on Management of Data*, pp. 25-30, Beijing, China, June 12-14, 2007.

- **[Wha09]** Kyu-Young Whang, “DB-IR Integration and Its Application to a Massively-Parallel Search Engine,” *The 18th ACM Conference on Information and Knowledge Management (CIKM 2009) (keynote speech)*, Hong Kong, China, Nov. 3, 2009.

- **[Wik11]** Wikipedia, 2011. available at http://en.wikipedia.org/wiki/NoSQL.

- **[WSS+10]** Guozhang Wang, Marcos Vaz Salles, Benjamin Sowell, Xun Wang, Tuan Cao, Alan Demers, Johannes Gehrke, and Walker White, "Behavioral Simulations in MapReduce," In *Proc. 36th Int'l Conf. on Very Large Data Bases (VLDB)*, Sept. 2010.

- **[YDHP07]** Hung-chih Yang, Ali Dasdan, Ruey-Lung Hsiao, and D. Stott Parker, "Map-reduce-merge: simplified relational data processing on large clusters," *In 2007 ACM SIGMOD Int'l Conf. on Management of Data*, June 2007.

- **[YIF+08]** Yuan Yu, Michael Isard, Dennis Fetterly, Mihai Budiu, Úlfar Erlingsson, Pradeep Kumar Gunda, and Jon Currey, "DryadLINQ: A System for General-Purpose Distributed Data-Parallel Computing Using a High-Level Language," In *Proc. 8th Symposium on Operating Systems Design and Implementation* (*OSDI '08*), Feb. 2008.

- **[YYTM10]** Christopher Yang, Christine Yen, Ceryen Tan, and Samuel Madden, "Osprey: Implementing MapReduce-style fault tolerance in a shared-nothing distributed database," In *Proc. IEEE 26th Int'l Conf. on Data Engineering (ICDE)*, Mar. 2010.