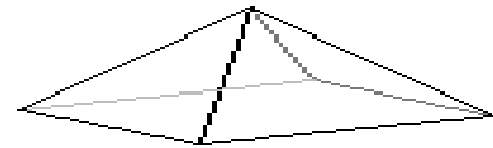




Institut de Recherche
en Informatique de Toulouse
CNRS - INP - UT3 - UT1 - UT2J



Pyramid

Evolution of Data Management Systems for Big Data Applications

Prof. Abdelkader Hameurlain

Hameurlain@irit.fr

Informatics Research Institute of Toulouse IRIT

Head of Pyramid Team*

**Paul Sabatier University PSU
Toulouse , France**

*** Query Processing & Optimization in Parallel & Large-scale
Distributed Environments**

0. Introduction (1/2) : Main Problems of Data Management

[Sto 98, Ozsú 11, ...]

“Data needs to be: <Captured, Stored, Queried, Processed and Turned in Knowledge>”

- **Data Modelling & Semantic**
- **Query Processing & Optimization (OLAP)**
- **Cost Models**
- **Concurrency Control/Transactions (OLTP)**
- **Replication & Caching**
- **Security & Privacy**
- **Monitoring Services**
- **Resource Discovery**
- **Autonomic Data Management (self-tuning, self-repairing, ...), ...**
- ...

➔ Data Management Systems

0. Introduction (2/2) : Evolution of Data Management Systems [Gra 96]

➔ ***"The present without past has not future"*** Fernand Braudel

▶ **<Concept** ➔ **Systems: *Objective*** [Ham 13]

■ **File Management Systems: *Storage Device Independence***

■ **Uni-processor (Rel.) DB Systems DBMS [Codd 70]: *Data Independence***

■ **Parallel DBMS [Dew 92, Val 93]: *High Performance & Data Avail.***

■ **Distributed DBMS [Ozs 11]: *Location/Frag./Replication Transparency***

■ **Data Integration Systems [Wie 92]: *Uniform Access to Data Sources***

Characteristics = <Distribution, ***Heterogeneity***, Autonomy>

■ **Data Grid Systems [Fos 04, Pac 07]: *Sharing of Available Resources***

■ **xxxxxxxxxxx**

■ **Cloud Data Manag. Systems [Aba 09, Sto 10]: *Pay-Per-Use* ➔ ***Economic Models*****

Characteristics = <***Elasticity, Fault-Tolerant*** >

➔ **Evolution Or Crossroad ?**

Evolution of Data Management Systems For Big Data Applications

Outline

I. From FMS to Parallel Relational DBMSs [Dew 1992, Val 93]

- ◆ File, Databases & Rel. DBMS: Objectives and Limitations
- ◆ Parallel DBMS: Motivations, Characteristics & Challenges

II. Big Data Management in Cloud Systems (Hadoop/MapReduce MR)

- ◆ Motivations ?
- ◆ Main Characteristics of Cloud Systems
- ◆ Parallel DBMS **Meet** MapReduce
- ◆ Advantages & Weakness of MR & Parallel DBMS
- ◆ Classification of CDMS & Evolution of DML
- ◆ Comparison between Parallel DBMS & MR

III. Conclusion

I.1 From FMS to Parallel Relational DBMSs (1/4)

S1. File Management Systems FMS

■ File Concept

➔ **Storage Device Independence**

[Storage] <File> [Program/Application]

▶ **File Management System**

■ Drawbacks of FMS

- **Data description/structure must be done in each program**
- **Relationships between files are materialized (New Files)**
 - ▶ **Software Eng. Requirements**

➔ **Database Concept**

I.2 From FMS to Parallel Relational DBMSs (2/4)

S2: Databases DB and Relational DBMS [Codd 70]

■ DB Objectives:

- ▶ **Centralization** of Data Structures (DB Schema)
- ▶ **Prog-Data Independence** = <Physical & Logical> Independence

■ Main Characteristics (Rel. DB)

- **Structured Data: Relation Concept**
- **Relational Algebra: Commutative, Internal Law**
- **From Procedural → Declarative Languages: SQL [Cham76], QUEL [Sto 76], QBE [Zlo77]**
 - ▶ **The System will find the (near) Optimal Access Path**
 - ➔ **Optimizer** [Sel 79, Wong 76, Gan 92, ...]

I.3 From FMS to Parallel Relational DBMSs (3/4)

→ Limitations of Uni-proc. Query Optimization Methods wrt **Decision Support Systems (RDBMS)**

- **Complex Queries:** *Number of Joins >6*
- **Size of Research Space [Tan 91]:** *Very Large (e.g. 2^{N-1})*
- **Optimization Cost [Lan91]:** *can be very expansive (e.g. Deterministic Strategies)*
- **Optimal Execution Plan:** *not guaranteed (e.g. Randomized Strategies)*
 - ➔ **Requirements in: High Performance HP & Resource Availability**
 - ➔ **Introducing a New Dimension: Parallelism**
- ➔ **Parallel Relational Database Systems [Dew 92]**

I.4 From FMS to Parallel Relational DBMSs (4/4)

S3: Parallel Relational DB Systems [Dew 92, Val 93, Lu 94]

■ Motivations: **Declarative Relational Languages** (e.g. SQL)

- Automatic Parallelization of **<Partitioned, Independent, Pipelined> //**
- Regular Data Structures : → **Static Annotations**
- Decision Support Queries: Complex Queries, Huge DB (TB, PB, ...)

■ Objectives [Dew 92]:

- **Best Trade-off Cost/Performance** wrt Mainframe
- **High Performance HP**
 - ◆ Minimizing the **Response Time**
 - ◆ Maximizing the Parallel System **Throughput**
- **Scalability** (**≠ Elasticity**)
 - ◆ Adding New resources (CPU, Memory, Disk)
 - ◆ Adding New Users (Applications)

➔ **Holding the Same Performance**
- **Resource Availability: Complex Queries, Fault-Tolerant**

I.5 Parallel Rel. DB Systems [Dew 92, Val 93, Ham 93, Lu 94]

■ Main Characteristics

- Parallel Architect. Models: SM, SD, DM= Shared-Nothing Archi.
- Parallelism Forms: <Partitioned, Independent, Pipelined>
- Data Partitioning:
 - Approaches: <Full Declustering, Partial Declustering>
 - Methods: <Round Robin, Range Partitioning, Hashing>

■ Main Challenges:

- Partitioning Degree of each Relation?
- Parallelism Degrees of Rel. Operators (e.g. Join)
- Parallelization Strategies: <One-Phase, 2-Phases> Approaches
- Resource Allocation: Data & Tasks Placement
- Optimization of Data Communications: Plague of Parallelism (Shuffle Issue in MapReduce)
..... Towards Cloud Computing & Big Data Manag. Why ?

II. Towards Cloud Data Management Systems CDMS

[Aba 09, Sto 10/13, Agr 10-12, Chaud 12, Zhou 12, Kald 12, Gra 13, LI 14, Unt 14, Norvag 14, Akba 15, Bon 15, Aba 16 ...]

Outline

- Big Data, Cloud Computing & MapReduce MR: **Motivations?**
- Main **Characteristics** of Cloud Systems [D. Agrawal et al. 2011]
- “Parallel Databases **Meet** MapReduce” [Zhou et al. 2012, VLDB Jo.]
- Advantages & Weakness of Parallel DBMS & MR
- Classification of Cloud Data Management Systems
- Evolution of DML & Comparison between // DBMS and MR

II.1 Big Data & Towards Cloud Computing (MR): **Motivations(1/3)**

■ “Big (Very Large?) Data” : Generated from

- Specific Requirements of **Web Applications** : Log Processing, Analysis of Streaming Sensor Data, Social Network, Document Indexing,.....
- Computer Simulations, Satellites, Astronomy, Live Science, IS, etc....

Remarks: **43rd** Intl. Conf. **on Very Large DB**; **36th** Intl . Conf. On **Data Management**.

Parallel DBMS: <TERADATA, → 1984; DB: **11 Terabytes** → 1996>

➔ **Big Data** → **“Moving Target ”** [Valduriez 2016]

■ **Big Data Characteristics [Val 14, Sto 13]: The 4 V's**

- **Volume:** Refers to Very Large Amounts of Data
- **Velocity:** Data Streaming (Producer-Consumer Dataflow in “real time”)
- **Variety:** Heterogeneity of Data Formats and Semantics
- **Veracity/Value:** Meaningful of the Results? (Data Mining)
- **Other V's:** ● **Validity:** Correction and accuracy of data? ● **Volatility:** Necessary period to store this data?

➔ **What are the proposed solutions?**

II.1 Big Data, Cloud Computing & MapReduce: **Motivations(2/3)**

- **Big Data Characteristics: a Solution for “the 3 V’s” [Val 14] ?**
 - **Volume:** Refers to very large amounts of Data
 - ➔ **Parallel Database Systems [Dew 92]**
 - **Velocity:** Streaming Data
 - ➔ **Data Stream Management Systems [Ozu 11, Chapter 18]**
 - **Variety:** Heterogeneity of Data Formats and Semantics
 - ➔ **Data Integration Systems [Wied 92]**

However, why these systems are not naturally used?

II.1 Towards Cloud Computing & MapReduce: **Motivations (3/3)**

- **Current Solutions (Infrastructures & Software) are:**
 - Proprietary & Expensive**
 - ➔ **Open Source Alternatives, Simple Programming Model ! (e.g. MapReduce), Low Costs (Commodity Hardware CH)**
- **Ability to scale resources (up, down, out) dynamically on- demand :**
 - ➔ **Elasticity** ➔ **Pay-Per-Use PPU**
- **How the systems should react “strongly” to Failures?**
 - ➔ **Fault-Tolerance : <Commodity Hard., Data Replication, HDFS>**
- **Cloud Environments do not to be Owned nor Managed by a Customer (PPU Approach):** **Users** ➔ **Multi-tenant**
 - <Tenant, Provider> trough SLA (Service Level Agreement)**
 - ➔ **Performance Isolation**

II.2 Main Characteristics of Cloud Systems [Agra. et al. 2011]

- **Scalability (Infrastructure: Shared-nothing Architecture)**
- **Elasticity [Ozu 11]**
 - «The ability to scale resources out, up, and down dynamically to accommodate changing conditions»
 - ➔ **PhD: SLA-driven Cloud Elasticity Management Approach** [Y. Kouri, Dec. 13]; Dir. P. Cointe, Nantes, France
- **Strong Fault-Tolerance: (CH, Data Replication, HDFS (Hadoop Env))**
- **Ability to run on Commodity Hardware CH (Low Cost)**
- **Performance Isolation [Nara 13]: Users ➔ Multi-tenant & SLA (Service Level Agreement) Meeting**
- ➔ **New Context = <Dist., Large-scale, Stable, Multi-tenant, Commodity Hardware, Service on-demand>**
- ➔ **Introduction of Economic Models in the Resource Management**

II.3 Advantages & Weakness of Parallel DBMS & MR

“SCOPE: Parallel Databases Meet MapReduce” ; MS [Zhou 2012,...]

■ Objective : combines benefits from execution engines

- **Parallel DB Systems**

&

→ for Large-scale Data Analysis

- **MapReduce**

→ <Easy Programmability, Massive Scalability, HP >

■ Advantages of // DB Systems [Dew 92]

- Relational Schema (→ Easy Annotations)
- Declarative Query Language (→ Automatic Optimization Process)
- Sophisticated Query Optimizers-Parallelizers : {Partitioned, Indep., Pipelined //}
- +/- Comm. Costs : Avoid the **Data Redistribution** (+/-: in some cases)

■ Weakness of // DB Systems (in Massive Large Scale):

- Run Only on Expensive Servers
- Fault - Tolerance (in the case of massive // DB)
- Web Data Sets are not structured
- Communication Costs: **Data Redistribution (=Reshuffling in MR)**

II.3 Advantages & Weakness of Parallel DBMS & MR

“SCOPE*: Parallel Databases **Meet** MapReduce”; **MicroSoft**

■ Advantages of MR

- **Scaling very well (to manage massive data sets)**
- **Strong Fault -Tolerance (Data Replication, HDFS)**
- **Mechanism to achieve Load-Balancing**
- **Support the Intra-operation & Independent Parallelisms**

■ Weakness of MR: Side Applications

Developers:

- **Are forced to translate their business logic to MR model**
- **Have to provide implementation for the M & R functions**
- **Have to give the best scheduling of M & R operations**
- ➔ **More Hot Problems!**
- **+ Data Dependence (Data Independence of DB Concept!)**
- **+ Extensive Materialization (I/O)**
- **+ Data Reshuffling (Repartitioning) between M & R ➔ Plague of Parallelism**

*: **SCOPE Proposals (Structured Computations Optimized for Parallel Execution)**

II.4 Classification of Cloud Data Manag. Systems

■ **Early Generation** of Big Data Manag. Systems BDMS:

- **NoSQL Databases/MapReduce Systems**
based on **Type of Data Store**

■ **Next Generation** of BDMS (Evolution of NoSQL Systems):

- **New SQL = Scalable Power of NoSQL Systems +**
ACID Properties (of Rel. DBMS!)

■ **Latest Generation** of BDMS: **Data Integration Approach** based on **Mediator –Wrapper Architecture [Wied 92]**

- ➔ **Insure a Uniform Access to Heterogeneous, Autonomous, and Distributed Data Sources**

- **Multistore Systems :**

Polybase [Dew 13], SCOPE [Zho 12] , CoherentPaas Proj. [Bon 15]

II.5 Evolution of Data Manipulation Languages

Charact. → Nature of Languages	Functions (Power)	Advantages	Drawbacks
L1: Proc./Func. Languages (e.g. MapReduce) [Bigtable, PNUTS]	Filter & Project Google, Yahoo!	– Simplicity of Programming Model	– Complexity to read and optimize prog. – Data Str. Dependency (Rewriting similar code on different data sets)?
L2: P/FL with Relational Operators (RO) [PIG Latin, Jaql]	Rel. Operators Towards SQL func Yahoo!, IBM	– Prog. are more readable – Automatic Logical Optim. Proc.	Developers provide Scheduling of RO → No Physical Optimization
L3: Declarative Languages [HiveQL, SCOPE, CloudMdsQL,...]	Close to SQL + Specific Operators MS, FB, IBM & Goo	Automatic : – Optimization – Parallelization (→ avoid Data Reshuffling)	“Lack of statistics stored in The catalog” → “Blinds the optimization Process”

II.6 Comparison between // Rel. DBMS & MapReduce

Systems Parameters	DB & // Rel. DBMS	MapReduce (Hadoop Env.)/ <u>Cloud. Systems</u>
Type of Applications	OLAP & OLTP (ACID)	OLAP: Yes; OLTP: Not suitable (Initially!) → New SQL
Data Models	Data Structured (Data Schema)	Unstructured or semi-Structured , ...(more Flexible!)
Data Independence	Yes	No (Initially)
Query Languages	Declaratives	Procedurals (initially)
Optimization & Parallelization	Automatic Optim. & // Annotations: Easy	Explicit Optim. (initially) Annotations: Very difficult
Scalability & Elasticity	Scalable & Dynamic	Scalable & Elastic
Fault-Tolerance	Weak	Strong
Location -----	Known in advance -----	SLA Negotiation -----
Maturity	Strong	Weak (at this moment!)

III. Summary & Conclusion (1/2) :

Evolution of Data Management Systems

- **File Management Systems:** *Storage Device Independence*
- **Uni-processor Rel. DB Systems DBMS [Codd 70]:** *Data Independence*
- **Parallel DBMS [Dew 92, Val 93]:** *High Performance & Data Availability*
- **Distributed DBMS [Ozs 11]:** *Location/Frag./Replication Transparency*
- **Data Integration Systems [Wie 92]:** *Uniform Access to Data Sources*
Characteristics = <Distribution, Heterogeneity, Autonomy>
➔ <Stable Systems, Not Scalable (Except. // DBMS)>
- **Data Grid Systems [Fos 04, Pac 07]:** *Sharing of Available Resources*
Characteristics = <Large-scale, Unstable Systems (Dynamics of Nodes)>
- **Cloud Data Manag. Systems:** <Pay-Per-Use> ➔ *Economic Models*
[Aba 09, Sto 10/13, Agr 10 /12, Cha 12, Col 12, Kald 12, Zho 12, Sul 12, Gre 13, Li 14, Unt 14,...]
Characteristics = <Elasticity, Fault-Tolerance>

III. Conclusion (2/2): Some Questions

- **Cloud Data Manag. Systems CDMS : <Elasticity, Fault-Tolerance>**
 - ➔ **CDMS: Evolution or Crossroad ?**

- **Criteria for Choosing a Data Management System ?**

- **Maturity of Big/Cloud Data Mana. Systems ?**

- **Main Contributions of Big Data/Cloud Computing wrt:
<Concepts, Applications, Socials, Ethics> ?**

Thank you for your attention



Contact: hameurlain@irit.fr

Informatics Research Institute of Toulouse IIRIT
Paul Sabatier University, Toulouse, France

IV.1 References: // DB Systems

- **D.J. DeWitt, J. Gray**, “*Parallel Database Systems: The Future of High Performance DB Systems*” , in: *Comm. of the ACM*, Vol. 35, 1992, pp. 85-98.
- **P. Valduriez**, : “*Parallel Database Systems: Open Problems and News Issues*”, in: *Distributed and Parallel DB*, Vol. 1, pp. 137--165, Kluwer Academic, (1993)
- **H. Lu et al.**, “*Query Processing in Parallel Relational Database Systems*”, IEEE CS Press, 1994
- **D. Taniar et al.**, “*High Performance Parallel DB Processing and Grid Databases*”, Ed. Wiley, 2008
- **A. Gounaris et al.** ; “ *Adaptive Query Processing: A Survey* ”, Proc. of the 19th British National Conf. on DB, Sheffield, UK, July 2002, pp. 11-25
- **A. Hameurlain, F. Morvan** ; “ *Parallel query optimization methods and approaches: a survey* ”, Intl. Journal of Computers Systems Science & Engineering, CRL Publishing, Vol. 19, No.5, Sept. 2004, pp. 95-114

IV.2 References: Distributed DB Systems

- **M.T. Özsu, P. Valduriez, Principles of Distributed Database Systems , 3rd Edition, April 2011, Ed. Springer Verlag**
- **D. Kossman , The State of the Art in Distributed Query Processing, ACM Computing Surveys, Vol. 32, No. 4; 2002**
- **M. Stonebraker , Hellerstein J.M. : Reading in DB Systems, M. Kaufmann Publisher, 3rd Ed., 1998**
- **M. Stonebraker, et al.: Mariposa: A Wide-Area Distributed Database System. In:VLDB Jour., 5(1), pp. 48--63, Springer, (1996)**
- **P. Valduriez, Principles of Distributed Data Management in 2020? Invited Talk, in:Dexa 2011, Toulouse/France), LNCS 6860, pp. 1-11.**
- ...

IV.3 References: Cloud Computing & Data Management (1/3)

- **F. Afrati & Ullman; Optimizing Joins in a MR Environment; EDBT'2010**
- **F. Afrati & Ullman; Optimizing Multiway Joins in a MR Environment; IEEE TKDE 23(9), 2011, pp; 1282 – 1298.**
- **S. Agarwal, et al., « Re-optimizing data-parallel computing », In Proc. of USENIX NSDI Conf., 2012.**
- **D. Agrawal et al., “Big Data and Cloud Computing: New Wine or Just New Bottles?”, In: VLDB'2010 Tutorial, PVLDB, Vol. 3, No. 2, pp. 1647-1648.**
- **D. Agrawal et al., “Big Data and Cloud Computing: Current State and Future Opportunities”, In: EDBT 2011, Tutorial, March, Uppsala, Sweden.**
- **D. Agrawal, et al., « The evolving landscape of data management in the cloud », Int. J. Computational Science and Engineering 7(1), 2012.**
- **Blanas et al. ; A Comparison of Join Alg. for Log Processing in MR; SIGMOD'2010.**

IV.3 References: Cloud Computing & Data Management (2/3)

- K.S. Beyer et al., « **Jaql**: a script language for large scale semi-structured data analysis », Proc. of VLDB Conf., 2011.
- Campbell et al.; Cloudy Skies for Data Management, ICDE'201
- R. Chaiken et al., « **SCOPE**: easy and efficient parallel processing of massive data sets », Proc. of VLDB Conf., 2008.
- S. Chaudhuri, « **What next?: a half-dozen data management research goals for big data and the cloud** », Proc. of PODS 2012.
- F. Chang et al., « **Bigtable**: A Distributed Storage System for Structured Data », ACM Trans. Comput. Syst. 26(2), 2008.
- B. F. Cooper et al., « **PNUTS**: Yahoo!'s hosted data serving platform », Proc. of VLDB, 2008.

IV.3 References: Cloud Computing & Data Management (3/6)

- **J. Dean, G. Ghemawat, « MapReduce: simplified data processing on large clusters », Proc. of OSDI Conf., 2004.**
- **G. De Candia, et al., « Dynamo: amazon's highly available key-value store », Proc. of the 21st ACM Symp. on Operating Systems Principles, 2007.**
- **A. Floratou, et al., « Can the Elephants Handle the NoSQL Onslaught? », Proc. of the VLDB Endowment, 2012.**
- **A.F. Gates, et al., « Building a High-level Dataflow system on top of MapReduce: The Pig Experience », Proc.of VLDB Conf., 2009.**
- **S. Ghemawat, et al., « The Google File System », Proc. of the 19th ACM symposium on Operating Systems Principles, 2003.**
- **Hadoop. <http://hadoop.apache.org>**
- **F. Deprez et al., «Special Theme : Cloud Computing, Platforms, Software and Applications », in ERCIM News, Number 83, Oct. 2010, pp. 12 – 51.**

IV.3 References: Cloud Computing & Data Management (4/6)

- **T. Kaldewey, et al., « Clydesdale: structured data processing on MapReduce », Proc. of EDBT Conf., 2012.**
- **A. Lakshman, P. Malik, « Cassandra: a decentralized structured storage system », Operating Systems Review, 44(2), 2010.**
- **R. S. G. Lancelotte, P. Valduriez, « Extending the Search Strategy in a Query Optimizer », Proc. of VLDB Conf., 1991.**
- **V. Narasayya, et al., « SQLVM: Performance Isolation in Mutli-tenant Relational Database-as_a_Service », Proc of CIDR'13, January 2013, Asilomar, CA, USA**
- **C. Olston, et al., « Pig Latin: a not-so-foreign language for data processing », Proc. of Sigmod Conf., 2008.**
- **C. Collet et al.; « De la gestion des bases de données à la gestion de grands espaces de données », Comité Bases de Données Avancées; July 2012.**
- **Maria Indrawan-Santiago, « Database Research: Are We At A Crossroad », Proc. of NBIS 2012, Melbourne, Australia, Sept. 26-28; pp. 45-51.**

IV.3 References: Cloud Computing & Data Management (5/6)

- **A. Paramswaran, “An interview with S. Chaudhuri” , In: XRD Vol. 19, No. 1, Sept. 2012**
- **M. Stonebraker, et al., « MapReduce and Parallel DBMSs: friends or foes? », Commun. ACM 53(1), 2010.**
- **Thakar & Szalay; Migration a large Science DB to the Cloud, HPDC’2011**
- **A. Thusoo, et al., « Hive- a warehousing solution over a MapReduce framework », Proc. of VLDB Conf., 2009.**
- **A. Thusoo, et al., « Hive- a petabyte scale data warehouse using Hadoop », Proc. of ICDE Conf., 2010.**
- **Y. Yu et al., « DryadLINQ: a system for general purpose distributed data-parallel computing using a high level language », Proc. of OSDI Conf., 2008.**
- **J. Zhou, et al., « SCOPE : Parallel databases meet MapReduce », VLDB Journal, 2012.**
- **M.F. Sakr et al.; “Center of Gravity Reduce Task Scheduling to Lower MapReduce Network Traffic”; IEEE Cloud Conf. , 2012, pp. 49-58.**
- **S. Ibrahim, et al.; “ LEE: Locality/fairness-aware key partitioning for MapReduce in the Cloud”; Conf. on Cloud Computing Technology & Science; pp. 17 – 24.**

IV.3 References: Cloud Computing & Data Management (6/6)

- **F. Li et al., “Distributed Data Management Using MapReduce”; ACM CS, Vol. 46. No. 3, January 2014.**
- **G. Graefe et al. “Elasticity in Cloud Databases and Their Query Processing”; Intl Journal of Data Warehousing and Mining, Vol. 9, No. 2 April-June 2013**
- **P. Unterbrunner et al.; “High availability, elasticity, and strong consistency for massively parallel scans over relational data”; in VLDB Jo, Vol. 23, pp. 627-652, 2014.**
- **P. Valduriez, « Indexing and Processing Big Data”; Seminar: Mastodons Indexing Scientific Big Data, Paris, January 2014.**
- **C. Doulkeridis, K. Norvag, “A Survey of Large-scale Analytical Query Processing in MapReduce”; VLDB Journal, 23(3), 2014**
- **Liroz-Gistau et al. “ Data Partitioning for Minimizing Transferred Data in MapReduce” in: Globe Conf. , 2013, p. 1 – 12; Also, in: PhD Thesis, Dec. 2013**
- **A. Hameurlain, , «Large-scale Data Management Approaches: Evolution and Challenges ». In: ACOMP 2013 (Invited Talk), Ho Chi Minh City, Vietnam, 23-25 Oct. 2013.**

IV.4 References: Query Optimization Taking wrt Economic Aspects

- **Trummer, I., and Koch, C. Approximation Schemes for Many-Objective Query Optimization. In Proceedings of the ACM SIGMOD international conference (SIGMOD '14) (Snowbird, UT, USA, June 22-27, 2014). ACM Press, New York, NY, 2014, 1299-1310.**
- **Trummer, I., and Koch, C. A Fast Randomized Algorithm for Multi-Objective Query Optimization. In Proceedings of the ACM SIGMOD international conference (SIGMOD '16) (San Francisco, USA, June 26th - July 1st, 2016). ACM Press, New York, NY, 2016.**
- **Ortiz, J., de Almeida, V. T., and Balazinska, M. Changing the Face of Database Cloud Services with Personalized Service Level Agreements. In Proceedings of the Seventh Biennial Conference on Innovative Data Systems Research (CIDR '15) (Asilomar, CA, USA, January 4-7, 2015). Online Proceedings, www.cidrdb.org, 2015**
- **Killapi, H., Sitaridi, E., Tsangaris, M. M., and Ioannidis, Y. Schedule optimization for data processing flows on the cloud. In Proceedings of the ACM SIGMOD international conference (SIGMOD '11) (Athens, Greece, June 12-16, 2011). ACM Press, New York, NY, 2011, 289-300.**
- **Lang, W., Shankar, S., Patel, J. M., and Kalhan, A. Towards Multi-tenant Performance SLOs. In Proceedings of the IEEE 28th International Conference on Data Engineering (ICDE '12) (Washington, DC, USA, 1-5 April, 2012). IEEE Computer Society, 2012, 702-713**

IV.4 References: Data Replication in Cloud Env.

- **B.A. Milani, N.J. Navimipour. A comprehensive review of the data replication techniques in the cloud environments: major trends and future directions. Journal of Network and Computer Applications, 64 , pp. 229–238, (2016)**
- **Q. Wei, B. Veeravalli, B. Gong, L. Zeng, and D. Feng. CDRM: A Cost-Effective Dynamic Replication Management Scheme for Cloud Storage Cluster. Proc. of the IEEE Int. Conf. on Cluster Computing (CLUSTER), pp. 188-196, (2010).**
- **N. Bonvin, T. G. Papaioannou, K. Aberer. Autonomic SLA-driven Provisioning for Cloud Applications. Proc. of Int. Symp. on Cluster, Cloud and Grid Computing, pp. 434- 443, (2011).**
- **Z. Cheng, et al. ERMS: An Elastic Replication Management System for HDFS. Proc. of the IEEE Int. Conf. on Cluster Computing Workshops , pp. 32-40, (2012)**
- **W. Lang, S. Shankar, J. Patel, A. Kalhan. Towards Multi-Tenant Performance SLOs. IEEE Trans. On Knowledge and Data Engineering, V. 26, No. 6, pp. 702–713, (2014).**
- **J.-W. Lin, C.-H. Chen, and J.M. Chang, “QoS-Aware Data Replication for Data Intensive Applications in Cloud Computing Systems,” IEEE Trans. Cloud Computing, vol. 1, no. 1, pp. 101-115, Jan.June 2013**
- **F. R. C. Sousa, J.C. Machado. Towards Elastic Multi-Tenant Database Replication with Quality of Service. In Proc. of Int. Conf on Utility and Cloud Computing, UCC '12, pp. 168-175. IEEE Computer Society, Washington, DC, USA, (2012)**
- **G. Silvestre, S. Monnet, R. Krishnaswamy & P. Sens. AREN: A Popularity Aware Replication Scheme for Cloud Storage. Int. Conf. on Parallel and Distributed Systems, pp. 189–196, (2012).**
- **K. A. Kumar et al.. SWORD: Workload-Aware Data Placement and Replica Selection for Cloud Data Management Systems. The VLDB Journal, Special Issue paper, Vol. 23, N. 6, pp. 845-870, (2014)**