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

Pyramid

Methods and Tools for Big Data

Prof. Abdelkader Hameurlain

Hameurlain@irit.fr

Informatics Research Institute of Toulouse IRIT

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, Ozsu 11, …]

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

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

➡ Data Management Systems DMS

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 FMS: *Storage Device Independence*
- Uni-processor DB Systems DBMS [Codd 70]: *Prog-Data Independence*
- Parallel DBMS [Dew 92, Val 93]: *High Perf., Scalable & Data Availability*
- Distributed DBMS [Ozs 11]: *Transparency of Location, Frag., Replication*
- Data Integration Systems [Wie 92]: *Uniform Access to Data Sources*

  Characteristics =<Distribution, *Heterogeneity, Autonomy*>
- Data Grid Systems [Fos 04]: *Sharing of Available Resources*
- Mobile Database Systems : *Decentralized Control & Scalability*
- Cloud Data Mana. Systems [Aba 09, Sto 10]: *Objective ?*

  Characteristics *=<Elasticity, Fault-Tolerant >*

➡ **Evolution or Crossroad ?**

Methods & Tools for Big Data

I. From File Mana. Systems FMS to Database MS DBMS

- Motivations, Objectives, Organizations & Drawbacks
- Databases & Rel. DBMS: Motivations & Objectives

II. Parallel Relational DBMS

- Motivations & Objectives
- Characteristics and Challenges

III. From Distributed DBMS to Data Integration Systems DIS

- Motivations , Objectives & Designing of Distributed DB
- Distributed Query Processing & Soft. Architecture
- Mediator-Wrappers Architecture & Query Processing Methodologies

IV. Cloud Data Management Systems CDMS

- Motivations, Objectives & Main Characteristics of CDMS
- Classification of CDMSs : 3 Generations (G1, G2 & G3)
- Advantages & Weakness of MR Systems & Parallel DBMSs
- Comparison between Parallel DBMSs & MR Systems

V. Conclusion & References

I.1. File Management Systems (1/2)

- **File Concept**

  ➡ ***Program and Storage Device <u>Independence</u>***

  [Storage] <File> [Program/Application]

  ▶ **Software Eng. Requirements**

- **File Organization: 4 types**
  - **< Sequential /Indexed > Organization**
  - **< Hashing/Relative> Organization**

I.2. File Management Systems (2/2)

- **Access Methods AM**
  - **Sequential AM**
  - **Key AM :=<Indexed/Hashing> AM**
- **Drawbacks of FMS**
  - **Data description must be done in each program**
  - **Relationships/Links between files are materialized (➔ New files)**

➡ Database Concept

I.3. Database and DBMS (1/2)

Concept of Database DB: Motivations

- **Separation between Data Structures (DB Schema) and Program**
- **Prog-Data Independence = <Physical & Logical> Independence**

Fundamental Objectives of a DB

- **Separation** of Data Description and Data Manipulation
- **Data Independence**: Logical & Physical
- **Procedural & Declarative** Interfaces/Languages
- **Query Processing and Optimization**
- **Data Integrity**/Sharing/Privacy/Security
- **Easy Data Administration**
- …

I.4. Database and DBMS (2/2)

- **Database Management System DBMS [Del 80, Date 86, Mir 02, Ull 89]**
  - **Software allowing users to interact with a DB**
  - **Implementation of main objectives of a DB**

- **Main Functions/Tools of DBMS**
  - **Data Description → DDL (Data Models : Concept. , Logical, Phys.)**
  - **Data Manipulation → DML (Querying and Updating)**
  - **Data Integrity/Sharing (Transaction & Concurrency)/Security**
  - **Data Administration, ….**
  - …….

  ➡ DB Design, Languages, and Methods (Query Processing, Transaction & Concurrency Control, Integrity, Security, Administration).

- **DB Models: <Hierarchical, Network, Relational & Object>**

I.5. Relational DB and Relational DBMS [Codd 70] (1/3)

Main Characteristics of Rel. DB

- **Structured Data: Relation Concept to describe <Entities & Links>**
  - ➔ **Data Model Definition**
- **Stored Data on Disk ➡ Input/Output Management**
- **Relational Algebra: Commutative, Internal Law**
- **From Procedural to 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.6. Relational DBMS: Query Optimization [Sel 79] (2/3)

- **Problem Position [Gan 92]:**

  q $\in$ Query , p $\in$ {Execution Plans}, $Cost_p$ (q):

  - Find p calculating q such as $Cost_p$ (q) is minimum
  - Objective : Find the best trade-off between

    Min (Response Time) & Min (Optimization Cost)

- **Optimizer Structure= < St, Sp, C> [Gan 92]**

  – St: Search Strategies (➔ Intelligence)
  - <Physical Optim., Parallelization, Resource Allocation, …>

  – Sp: Search Space (➔ Control)
  - Data Structures/Queries: Linear Spaces, Bushy Space
  - Type/Nature of Queries

  – C: Cost Models (➔ Knowledge)
  - <Metrics, System Environment Description>

I.7. Limitations of Uni-proc. Query Optimization Methods wrt Decision Support Systems /OLAP (RDBMS) (3/3)

- **Complex Queries**: *Number of Joins >6*
- **Size of Research Space [Tan 91]:** *Very Large (e.g. 2^{N-1})*
- **Optimization Cost [Lan 91]:** *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]**

II.1 Parallel Relational DB Systems [Dew 92, Val 93, Lu 94]

- **Motivations: Declarative Relational Languages (e.g. SQL)**
  - **Automatic Parallelization of <Intra-operation & Inter-operation> Parallelism = <Partitioned & Independent, Pipelined> //**
  - **Regular Data Structures : ➔ *Static Annotations***
  - **Decision Support Queries: Complex Queries, Huge DB (TB, PB, …)**

- **Objectives [Dew 92]:**
  - **Best Trade-off between 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**

II.2 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 Degree of each Rel. Operator (e.g. Join)?
  - Parallelization Strategies: <One-Phase, 2-Phases> Approaches
  - Resource Allocation: Data & Tasks Placement/Scheduling
  - Optimization of Data Communications: Plague of Parallelism

    (Shuffle Issue in MapReduce)

    ..... Towards Cloud Computing & Big Data Manag. **Why ?**

III. From Distributed DBMS to Data Integration Systems

- An Example of Distributed DB DDB
- Motivations & Objectives of DDB
- Designing of Distributed DB
- Distributed Query Processing

- An Example of DIS, Motivations & Objectives
- Mediator-Wrappers Architecture
- Query Processing Methodologies
- Restricted Access Relational Operators

IV. Cloud Data Management Systems CDMS

IV. 1 Motivations (1/2): Big Data, Cloud Computing & MapReduce

- "Big Data" : Generated from Specific Requirements of Web Appli + Tradit. Appli. : C. Sim, Sat. , Astro, Live Sc, IS, ….

Remarks: 46th Very Large DB; 39th Intl . Conf. On Data Manag. Parallel DBMS: <TERADATA, ➔ 1984; DB: 11 Terabytes ➔ 1996>

➡ Big Data ➔ "Moving Target " [Valduriez 2016]

- Big Data Characteristics: the 3 V's (Volume, Velocity, Variety)

➡ What are the 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, Chap. 18]

- Variety: Heterogeneity of Data Formats, Semantics & Resources

➡ Data Integration Systems [Wied 92]

However, why these systems are not naturally used?

IV.1 Motivations (2/2): Towards Cloud Computing & MapReduce

- **Current Solutions (Infrastructures & Software) are:**
  Proprietary & Expensive

  ➡ **Open Source Alternatives, Simple Programming Model ! (e.g. MapReduce), Low Costs LC (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?**

  <Commodity Hard./LC, Data Replication, HDFS> ➡ **Fault-Tolerance**

- **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**

IV.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»
- **Strong Fault-Tolerance: (CH, Data Replication, HDFS (Hadoop Env)**
- **Performance Isolation [Nara 13]: Users ➔ Multi-tenant**

  & SLA (Service Level Agreement) Meeting
- **Ability to run on Commodity Hardware CH (Low Cost)**

  ➡ **New Context = <Dist., Large-scale, Stable, Multi-tenant, Service on-demand, Commodity Hardware >**

➡ **Introduction of Economic Models in the Resource Management**

IV.3 Classification of Cloud Data Manag. Systems

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

- **1st Generation G1: From MapReduce → SQL Like**
  - **MapReduce Systems →SQL-on-Hadoop Systems based on Type of Data Store**
  - **- Simple Queries= Selection Queries**
  - **-... Hive, MongoDB, Cassandra, Neo4j, SPARK, ...**

- **2nd Generation G2: Multi-tenant Par. RDBMS [Won15, Yin 18, ...]**
  - **Extension of Parallel Rel. DBMSs in the Cloud (→Elasticity)**
  - **- Complex Queries= Join Queries**
  - **- ... Amazon Redshift, Azure SQL DW, Google BigQuery, Snowflake DW, ...**

- **Latest Generation G3: The Meeting between G1 & G2**

<MR Systems and MP RDBMSs> based on the concepts: <Federated & Data Integration>

- **Multistore /Polystores Systems : <Polybase [Dew 13], SCOPE [Zho 12] , CoherentPaas Proj. [Bon 15]; BigDAWG [Sto 15], [Lec 18], ...>**

IV.4 First Gen G1.: Classification of Cloud Data Mana. Systems

- **Classification of NoSQL Systems: Type of Data Store**
  (Approx. 135 Systems)

  - **Key-value Store: <Azure Table Storage, DynamoDB, Redis, Riak, Voldemort, ...>**
  - **Document Store (XML, JSON): <MongoDB, CouchDB, RavenDB>**
  - **Column-family (Rel. DB, Data is stored in column):**
    <Hbase, Cassandra, Hypertable>
  - **Graph Databases (Social Networks):**
    <Neo4j, Infinity Graph, InfoGrid, ...>

IV.5 First Gen. G1 : From MR → SQL Like on-Hadoop Systems (1/2)

➡ Advantages and Weakness of MR

Advantages of MapReduce MR

- Scaling very well (to manage massive data sets)
- Strong Fault -Tolerance (Data Replication, HDFS)
- Mechanism to achieve Load-Balancing
- Support the Intra-Oper. & Independent Parallelisms (and the Pipeline //?)

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!**

- Prog-Data Independence is lost (Prog-Data Independence of DB Concept!)
- Extensive Materialization (I/O) (the Pipeline // is not implemented)
- Data Reshuffling (Repartitioning) between M & R ➡ Plague of Parallelism

IV.6 First Gen. G1 : From MR ➔ SQL Like on-Hadoop Systems (2/2)

➡ Advantages and Weakness of Par RDB Systems

Advantages of Par RDB Systems [Dew 92]

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

Weakness of Par RDB Systems :

- Run Only on Expensive Servers
- Fault - Tolerance
- Web Data Sets are not structured (Relational Schema)
- Communication Costs: Data Redistribution (=Reshuffling in MR)

IV.7 Comparison between Par RDBMS & MapReduce Systems (G1)

Systems Parameters	Par RDBMS	MapReduce Systems (Hadoop Env.)/(1st Genera.)
Type of Applications	OLAP & OLTP (ACID)	OLAP: Yes; **OLTP: Not suitable (Initially!)** ➔ New SQL
Data Models	Structured Data (Relational 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 -------------------- Maturity	Known in advance ------------------ Strong	SLA Negotiation ------------------------- Weak (at this moment!)

IV.8 Summary: Main Characteristics of Cloud DMS: G1 & G2

■ Main Characteristics of 1st G1: From MapReduce → SQL Like

- "One size does not fit all" : Systems are based on Type of Data Store
- Low Performance : <Selection Queries=one pass>;
- Extensive Materialization (I/O) (initially, the Pipeline // is not implemented)
- Loss of Data – Prog. Independence

Ind. Prod. : Bigtable, Hive, MongoDB, Cassandra, Neo4j, Riak, Hive/Tez, SPARK , ...

➡ **Weak Fault-Tolerance (Pipeline Parallelism)**

■ Main Characteristics of 2nd G2: Multi-tenant Parallel RDBMS

- + High Performance (Partitioned, Indep., Pipelined //) : Complex Queries
- + Declarative Query Languages & Optimizer/Parallelizer
- + Minimization of Comm. Costs (in some cases!)
- - Poor Semantic (Relational Model, "No one size fits all"!)

In. Prod.: Amazon Redshift, Azure SQL DW, Google BigQuery, Snowflake DW

IV.9 Application: Petasky – Mastodons Project (CNRS, LIMOS/LIRIS)

"Benchmarking SQL on MapReduce systems using large astronomy databases"; A. Mesmoudi et al.; In: Intl journal PDBD, 34(3), 2016

- **Objectives:** **"They report on the capability of 2 MR systems (Hive and HadoopDB) to accommodate LSST data management requirements" in terms of loading & execution times : < Data Loading & Indexing and Queries (Selection, Group By, Join) >**
- **Conclusions [Mes 2016] :**
  - **"We believe that the MR model is efficient for queries that need one pass on the data (e.g. Selection and Group By)"**
  - **" We believe that MR model is not suitable for handling Join queries "**

V.1 Summary :
Evolution of Data MS: <Concept ➔ Systems: *Objective*>

- **File Management Systems:** *Storage Device Independence*
- **Uni-processor Rel. DB Systems DBMS** [Codd 70]: *Data -Prog. Indepen*
- **Parallel DBMS** [Dew 92, Val 93]: *High Perf., Scalable & Data Availability*
- **Distributed DBMS** [Ozs 11]: *Location/Frag./Replication Transparency*
- **Data Integration Systems** [Wie 92]: *Uniform Access to Het. Data Sources*

  Characteristics =<Distribution, Heterogeneity, Autonomy>

  ➡ <Stable Systems, Not Scalable (Except. // DBMS)>

- **Data Grid Systems** [Fos 04, Pac 07]: *Sharing of Available Resources*
- **Mobile Database Systems :** *Decentralized Control & Scalability*
- **Cloud Data Manag. Systems:** *<Pay-Per-Use>* ➔ Economic *Models*

  1st Gen. : SQL-on-Hadoop Systems; 2nd Gen.: Extension of Par RDBMS in the Cloud; 3rd Gen.: Multistore/Polystores Systems

  Characteristics =<Elasticity, Fault-Tolerance, High Performance>

V.2 Conclusion: Maturity of Cloud Data Manag. Systems

- **Query Languages**
  - **Declarative Languages**
  - **Standardization**
- **More Experimentation & Benchmarking**
  - **TPC – H & TPC - DS**
- **Administration & Tuning/Supervision Tools**
- **Consideration of several V's simultaneously:**

  For instance : Volume & Velocity (OLAP & OLTP ?)!
- **Let time do its work!**

V.3 Conclusion : Criteria for Choosing a Data Mana. System

- **C1: Price (Investment) VS Pay-Per-Use (Cloud Computing Platform)**

- **C2: Characteristics of Applications**
  - **Nature of Applications: OLAP, OLTP, Hybrid**
  - **Data Models/Structures: File, DB, XML, ...**
  - **Degree of Schema (Sem.) Evolution (Data – Prog Independency)**
  - **Template Queries: Type & Nature of Queries and Indexing**

- **C3: Characteristics of DM Systems**
  - **Environment: Uni-proc., Parallel, Distributed**
  - **Fundamental Functionalities: DDL, DML, Programming Languages (Java/C + SQL, R, ...), Consistency Constraints, ...**
  - **DMS Administration & Tuning**

  ....

Thank you for your attention

==================

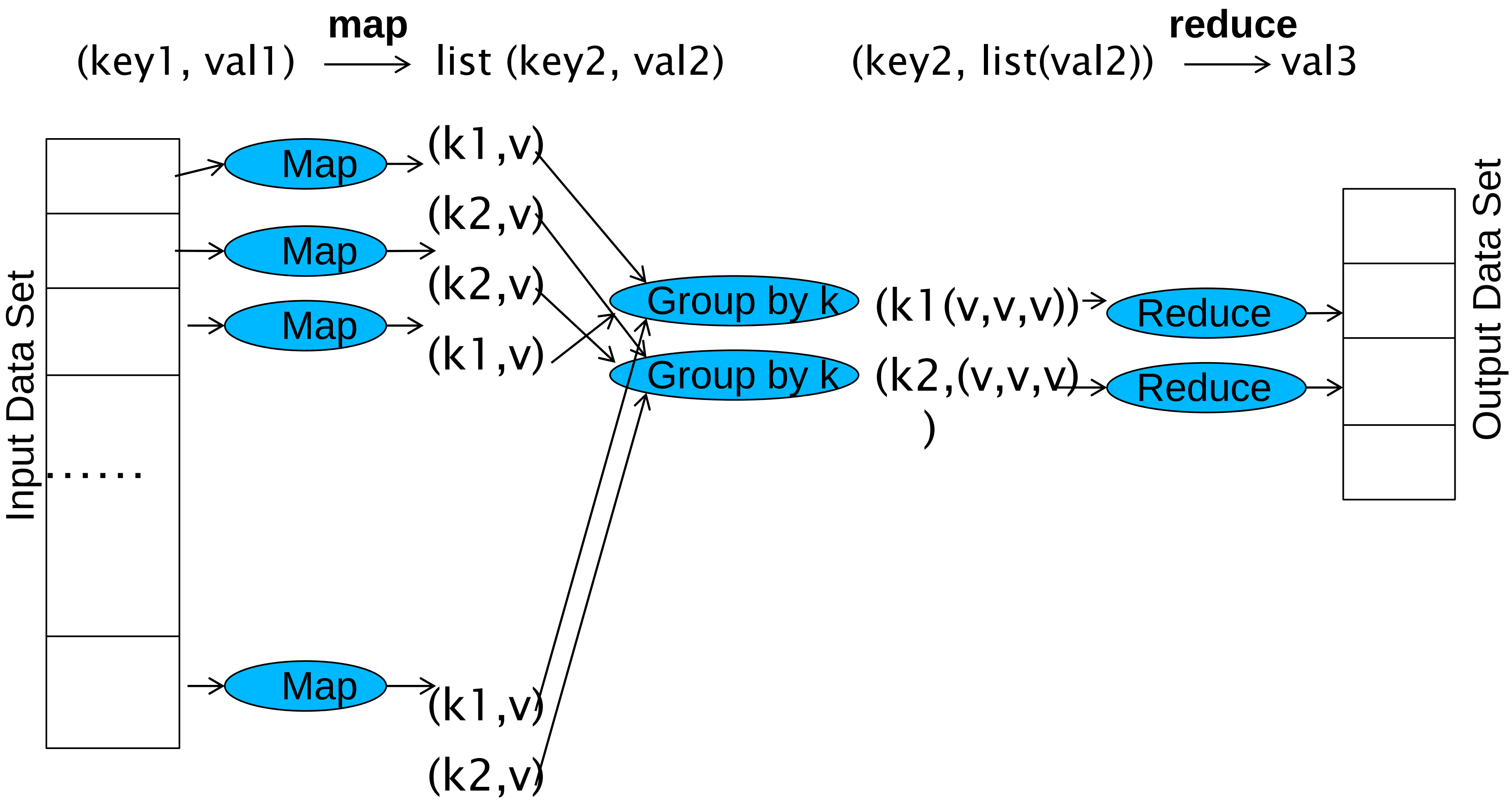
Contact: Hameurlain@irit.fr

Informatics Research Institute of Toulouse IRIT

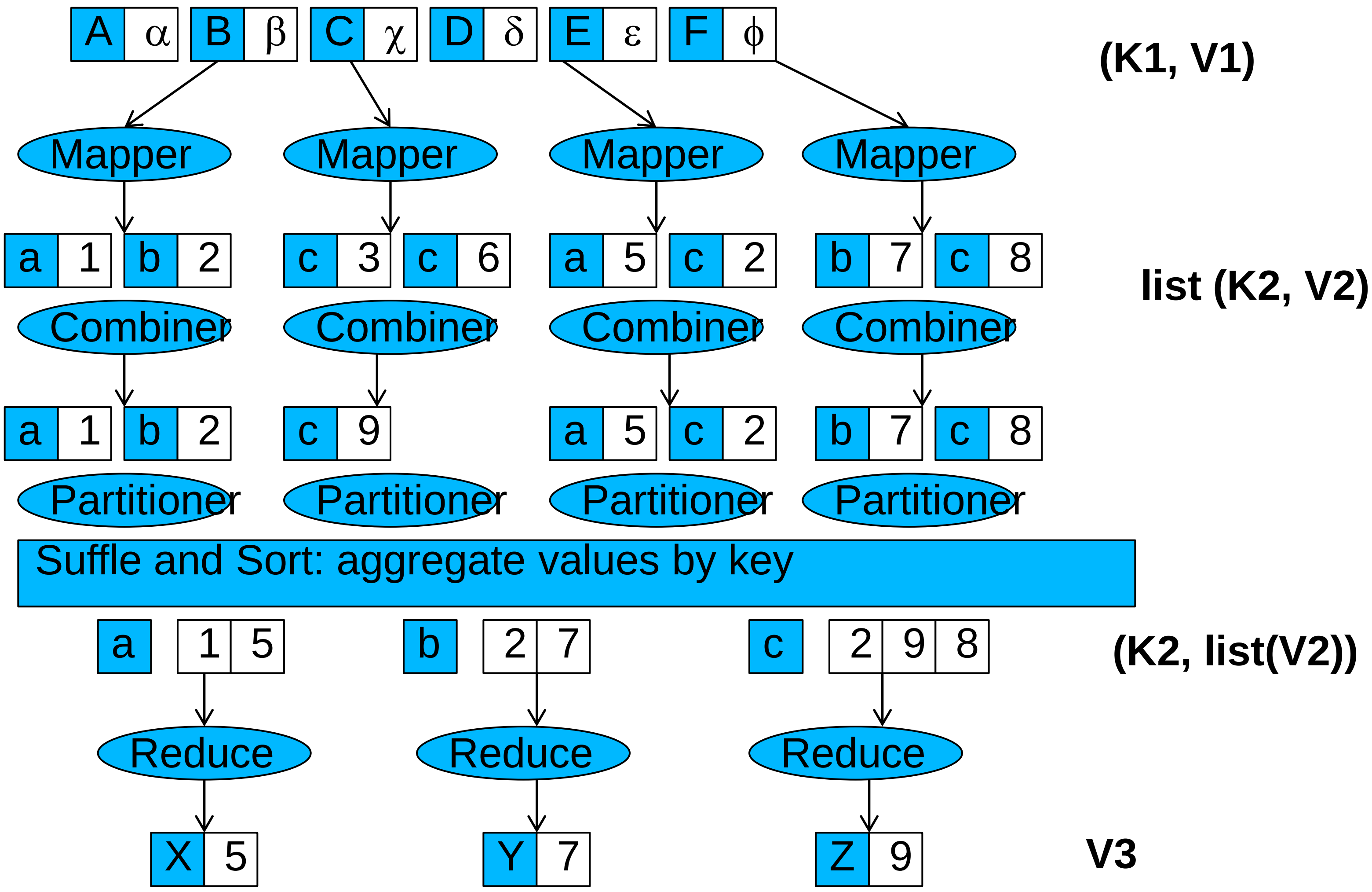
Pyramid Team

Paul Sabatier University
Toulouse , France

MapReduce Processing [Val 2010]



Combiner & Partitionner [Val 2010]



References: DB Management Systems

- E. F. Codd, “A Relational Model of Data for Large Shared Data Banks” In: Commun. ACM 13(6): 377-387 (1970).
- P. G. Selinger et al. : « Access Path Selection in a Relational Database Management System ». In: SIGMOD Conference 1979: 23-34.
- J. Gray, “Evolution of Data Management”, In: Computer 29(10): 38-46 (1996).
- G. Gardarin and P. Valduriez, “Relational Databases and Knowledge Bases”; Addison Wesley, 1989;
- C. Delobel et M. Adiba; « Bases de Données et Systèmes Relationnels »; Dunod Informatique, 1985.
- G. & O. Gardarin , « Le Client-Serveur »; Ed. Eyrolles, 1996

References: Parallel DBMS

- 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
- S. Ganguly, Waqar Hasan, Ravi Krishnamurthy Query Optimization for Parallel Execution. In: SIGMOD Conference 1992: 9-18.
- 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

References: Distributed DBMS

- M.T. Özsu, P. Valduriez, Principles of Distributed Database Systems , 4th Edition, April 2016, 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 Journal, 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

- G. Gardarin et P. Valduriez ; « SGBD Avancés - BD objets, déductives, réparties »; Ed. Eyrolles, 1989

- G et O. Gardarin , « Le Client-Serveur »; Ed. Eyrolles, 1996;

References: Data Integration Systems DIS

- G. Wiederhold ; Mediators in the Architecture of Future Information Systems, Journal of IEEE Computer, IEEE CS, Vol. 25, N°3, March 1992, pp. 38-49.
- J.-M. Hellerstein et al. ; Adaptive query processing: Technology in evolution, IEEE Data Engineering Bulletin, IEEE Computer Society, Vol. 23, N°2, June 2000, pp. 7-18.

➔ Centralized Control

- Avnur, R., Hellerstein, J.-M.: Eddies: Continuously Adaptive Query Processing. In: Proc. of the ACM SIGMOD , Vol. 29, pp. 261--272, ACM Press, (2000)
- Ives, Z.G., et al. ; Adapting to Source Properties in Processing Data Integration Queries, Proc. of the ACM SIGMOD, ACM Press, Paris, France, June 2004, pp. 395-406.
- Amsaleg, L., Franklin, M., Tomasic, A.: Dynamic query operator scheduling for wide-area remote access. In: Distributed and Parallel Databases, 6(3), pp. 217--246, Kluwer Academic Publishers, 1998
- Bouganim, L., Fabret, F., Mohan, C., Valduriez, P.: A dynamic query processing architecture for data integration systems. In: Journal of IEEE Data Engineering Bulletin, 23(2), pp. 42--48, IEEE CS, 2000

➔ Decentralized Control

- Ives, Z.-G., et al..: An adaptive query execution system for data integration. In: Proc. of the ACM SIGMOD Intl. Conf. on Management of Data, pp. 299--310, ACM Press, (1999). Tukwila/DHJ
- Urhan, T., Franklin, M.: XJoin : A reactively-scheduled pipelined join operator. In:IEEE Data Engineering Bulletin, 23(2), pp. 27--33, IEEE CS, (2000).
- Urhan, T, Franklin, M.: Dynamic pipeline scheduling for improving interactive query performance. In: Proc.of 27th Intl. Conf. on VLDB, pp. 501--510, M. Kaufmann, (2001)

References: Cloud Computing & Data Management (1/6)

- 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.

References: Cloud Computing & Data Management (2/6)

- K.S. Beyer et al., « Jaql: a scipt 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.

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 Map-Reduce: 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.

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. Lanzelotte, 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.

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 Jounal, 2012.
- M.F. Sakr et al.; "Center of Gravity Reduce Task Scheduling to Lower MapReduce Network Trafic"; 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.

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. Mesmoudi et al.; In: Intl journal Parallel & Distributed BD, 34(3), 2016.
- J. Duggan, et al. The BigDAWG Polystore System. SIGMOD Record, Vol. 44, No. 2, 2015.
- P. G. Brown. Overview of scidb: large scale array storage, processing and analysis. In SIGMOD, pages 963–968. ACM, 2010
- U. Cetintemel, et al. S-Store: A Streaming NewSQL System for Big Velocity Applications. PVLDB, 7(13), 2014

References (1/2): Query Optimization; Multi-Objective & SLA

➔ Multi-Objective Query Optimization

- **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.**
- **Kllapi, 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.**

➔ SLA/SLO Papers

- **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**
- **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**

References (2/2): SLA & Multi-tenant DBMS

- F. Chong, G. Carraro, and R. Wolter. Multi-Tenant Data Architecture. Microsoft Corporation. June 2006.
- P.Wong, Z. He, and Eric Lo. Parallel Analytics as a Service. Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data (SIGMOD'13), 25-36.
- O. Schiller, B. Schiller A. Brodt, and B. Mitschang. Native Support of Multi-tenancy in RDBMS for Software as a Service. EDBT 2011, March 22–24, 2011, Uppsala, Sweden.
- Z. Tan, and S. Babu. Tempo: Robust and Self Tuning Resource Management in Multitenant Parallel Databases. Proc. of the VLDB Endowment, Vol. 9, No. 10, 2016, pp. 720-731.
- Petrie Wongy, Zhian He, Ziqiang Feng, Wenjian Xu, and Eric Lo. Thrifty: Offering Parallel Database as a Service using the Shared-Process Approach. In Proceedings of the ACM SIGMOD intl. conf. (SIGMOD'15), May 31–June 4, 2015, Melbourne, Victoria, Australia.
- S.Yin, A. Hameurlain, F. Morvan: « LA Definition for Multi-Tenant DBMS and its Impact on Query Optimization. », IEEE Trans. Knowl. Data Eng. 30(11): 2213-2226 (2018)
- M. M. Kandi, S.Yin, A. Hameurlain; « Resource auto-scaling for SQL-like queries in the cloud based on parallel reinforcement learning » Int. J. Grid Util. Comput. 10(6): 654-671
- M. M. Kandi, S. Yin, A.Hameurlain:SLA-driven resource re-allocation for SQL-like queries in the cloud. Knowl. Inf. Syst. 62(12): 4653-4680 (2020)
- D. T. Wojtowicz, S. Yin, F. Morvan, Abdelkader Hameurlain: Cost-Effective Dynamic Optimisation for Multi-Cloud Queries. CLOUD 2021: 387-397
- A. Hameurlain, R. Mokadem: Special Issue: Elastic Data Management in Cloud Systems. Comput. Syst. Sci. Eng. 32(4) (2017)

References: Data Replication in Cloud Env.(1/2)

- 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, June 2013
- U. Tos, R.Mokadem, A.Hameurlain, Tolga Ayav:; « Achieving query performance in the cloud via a cost-effective data replication strategy. » Soft Comput. 25(7): 5437-5454 (2021)
- R. Mokadem, A. Hameurlain, « A data replication strategy with tenant performance and provider economic profit guarantees in Cloud data centers. » J. Syst. Softw. 159 (2020)
- U. Tos, R.Mokadem, A. Hameurlain, T. Ayav, S. Bora; « Ensuring performance and provider profit through data replication in cloud systems. » Clust. Comput. 21(3): 1479-1492 (2018)

References: Data Replication in Cloud Env. (2/2)

- 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, Vol. 23, N. 6, pp. 845-870, (2014)
- Y. Mansouri, A.N. Toosi, R. Buyya. Cost optimization for dynamic replication and migration of data in cloud data centers. IEEE Transactions on Cloud Computing (2017).
- C.L. P. Chen and C- Zhang. Data-intensive applications, challenges, techniques and technologies: A survey on big data. Information Sciences, 275: pp. 314–347, (2014).
- P. Xiong, Y. Chi, S. Zhu, H. J. Moon, C. Pu, and H. Hacigumus. Intelligent Management of Virtualized Resources for Database Systems in Cloud Environment. Proc. of Int. Conf. of Data Engineering (ICDE), pp. 87–98. (2011).