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

Pyramid

Data Management Systems: Evolution, State of the Art and Open Issues

Prof. Dr. 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, Ozs 16, …]

"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***>

- ....
- **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]:** ***Location/Frag./Replication Transparency***
- **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]:** ***Pay-Per-Use*** ➧ ***Economic Models***

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

➧ **Evolution or Crossroad ?**

Outline

I. Background & Fundamentals: [Codd 70, Sel 76, Dew 92, Val 93, ...]

- Databases & Uni-Proc. Rel. DBMS: Objectives & Limitations
- Parallel Rel. DBMS: Motivations, Characteristics & Challenges

II. Cloud Data Management Systems CDMS [Aba 09, Sto 10, ...]

- Motivations & Main Characteristics of CDMS
- Classification of CDMS : 3 Generations (G1, G2 & G3)
- Comparison between Parallel Rel. DBMSs & MR Systems

III. Main Contributions (Pyramid Team)

IV. Future Research Directions [Abadi et al. 19, The Seattle Report on DB Research]

V. Conclusion & References

<Context/Motivat./Pb. Position, State of the Art, Contributions, Open Issues>

I. Background & Fundamentals B & F

1. Databases DB and Relational DBMS [Codd 70]

- **DB Objectives:**
  - ▶ **Separation between Data Structures (DB Schema) & Control Struct.**
  - ▶ **Prog-Data Independence = <Physical & Logical> Independence**
- **DB Models: <Hierarchical, Network, Relational & Object>**
- **Main Characteristics (Rel. DB)**
  - **Structured Data: Relation Concept to describe <Entities & Links>**
  - **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. Background & Fundamentals B & F

2. Uni-proc. Rel. DBMS: Query Optimization [Sel 79]

- **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. Background & Fundamentals B & F

3. Limitations of Uni-proc. Query Optimization Methods wrt Decision Support Systems /OLAP (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, Val 93, ...]**

I.4. B & F: Parallel Relational DBMS (1/2) [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**

I.5. B & F: Parallel Rel. DBMS (2/2) [Dew 92, Val 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**
  - **Parallelism Degree of each Relation/Operator (e.g. Join)?**
  - **Parallelization Strategies: <One-Phase, 2-Phases> Approaches**
  - **Resource Allocation: Data & Tasks Placement/Scheduling**
  - **Optimization of Data Communications: Plague of Parallelism**
- **Weakness of Parallel Rel. DBMS**
  - **Run only on expensive servers**
  - **Weak Fault - Tolerance**
  - **Web Data Sets are not structured (Relational Schemas)**
  - **Communication Costs: Data Redistribution (=Reshuffling in MR)**

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

II.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, Buisness, ...

Remarks: 48th Intl. Conf. on Very Large DB; 41st Intl . Conf. On Manag. Of Data
Parallel DBMS: <TERADATA, → 1984; DB: 11 Terabytes → 1996>

➡ Big Data → **"Moving Target"** [Val 16]

- 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 16, Chap. 18]
  - **Variety:** Heterogeneity of Data Formats, Semantics & Resources
    ➡ Data Integration Systems [Wied 92]

However, why these systems are not naturally used?

II.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!**

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

- **Scalability (Infrastructure: Shared-nothing Architecture)**
- **Elasticity [Ozu 16]**

  «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 = <Service on-demand, Multi-tenant, Commodity Hardware >**

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

II.3 Classification of Cloud Data Manag. Systems CDMS

- **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: From Parallel RDBMS → Multi-tenant Par. RDBMS**
  - **Extension of Parallel Rel. RDBMSs with the "Cloud Concept"**
    → High Performance & Elasticity [Won15, Yin 18, ...]
    - **- Complex Queries= Join Queries**
    - **- Amazon Redshift, Azure SQL DW, Google BigQuery, Snowflake DW, ...**
- **3rd Generation G3: =<Distribution, *Heterogeneity, Autonomy*>**
  based on the concepts: <Multibase/Federated DB & Data Integration>
  - **Multistore/Polystores Systems: Polybase [Dew 13], SCOPE [Zho 12] , CoherentPaas Proj. [Bon 15], BigDAWG [Sto/Dug 15], [Sol 20], [Lec 18], ...**

II.4 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 **only** the Intra-Oper. & Independent Parallelisms (**Pipeline Par.?**)

■ 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 (Redistribution) between M & R ➡ **Plague of Parallelism**

II.4 First Gen. G1 : From MR ➔ SQL Like on-Hadoop Systems (2/2)

➡ Advantages and Weakness of Par. RDBMS

Advantages of Par. RDBMS [Dew 92]

- Relational Schemas (➔ 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. RDBMS

- Run only on expensive servers
- Weak Fault - Tolerance
- Web Data Sets are not structured (Relational Schemas)
- Communication Costs: Data Redistribution (=Reshuffling in MR)

II.5 Comparison between Par. RDBMS & MapReduce Systems (G1)

Systems Parameters	Par. RDBMS	MapReduce Systems (Hadoop Env.)/(1st Generation)
Type of Applications	OLAP & OLTP (ACID)	OLAP: Yes; **OLTP: Not suitable (Initially!)** ➔ NewSQL
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!)

III. Main Contributions of Pyramid Team (IRIT Lab, Toulouse)

- **Parallel RDBMS**
  - **Estimation of Parallelism Degree of a Relational Operator [Ham 92]**
  - **Minimization of Communication Costs Inter-operation [Ham 93]**

- **Mobiles Databases**
  - **Mobile Relational Algebra [Oza 08, Mor 11]**
  - **Embedded Cost Model [Hus 05]**

- **Cloud Data Management Systems & SLA (DBaaS)**
  - **Integration of Performance Aspects in SLO /SLA for DBaaS [Yin 18]**
  - **Integration of Economic Aspects in the Optimization Process meeting SLO [Mok 20, Tos 21]**

  ➡ **Cost Models : = < High Performance, Cost-effectiveness>**

IV.1 Future Research Directions (1/3)

- **New Context in CC= <Service on-demand, Multi-tenant, Commodity Hardware>**

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

- **Research Challenges [Abadi et al.2019; "The Seattle Report on DB Research"]**

  RC1: "Data Science"

  < End-to-End Processing of Data/Data-to-Knowledge Pipeline, Data Context & Provenance, Data Manag. in support of Machine Learning, ...>

  RC2: "Data Governance"

  <Data Use Policy & Data Sharing, Data Privacy , Ethical Data Science, ...>

  RC3: "Scalable Big/Fast Data Infrastructures"

  <New Hardware (CPU/GPU), Parallel & Distributed Processing, *Query Proc. & Optimization*, Cost-efficient Storage, Consistency (NewSQL) , HTAP (Hybrid Transaction Analytical Processing), Metrics & Benchmarks, ...>

  RC4: "Cloud Services"

  <*Elasticity*, Multi-tenancy, Performance Isolation, Multistore/Polystores Systems, Leveraging Machine Learning, Auto-Tuning, ...>

IV.2 Future Research Directions (2/3)

- **Open Issues wrt *Query Processing and Optimization***

  P1: Elastic Resource Allocation & Dynamic Data Replication

  [Kouri 13, Gra 13, Unter 14, Wong 15, Tan 16, Yin 18, Mok 20, …]

  P2: Data Skew & Load Balancing

  [Ram 12, Guf 12, Kwon 12/13, Elm 14, Akba 15, ….]

  P3: Data Partitioning & Redistribution (Reshuffling Issue in MR)

  (Optimization of Data Comm. in // DB Systems) [Chu 15, Lir 13, Sakr 12, …]

  P4: Big Data Indexing [Val 14, …., Knuth 73]

  ➔ [Val 14] "Indexing and Processing Big Data"

  In: Mastodons Indexing Scientific Big Data, Paris, January 2014.

IV.3 Future Research Directions: New Context & Open Issues (3/3)

- **P1: Elastic Query Optimization [...,Yin 18, Mok 20, ...]**
  - Resource Allocation: Scheduling/Placement of Tasks
  - Dynamic Data Replication
  - Cost Models :=<High Performance, Cost-effectiveness>

➡ **Designing of Dynamic Execution Models:**

Efficient (Tenant) & Cost-effective (Provider)

➡ **Objective Function:** Find the best trade-off between

- Multi-tenant Satisfaction (QoS (e.g. Response Time)) &
- Cost-effectiveness of Provider Services <IaaS, PaaS, DBaaS/ SaaS>

V. Summary & Conclusion (1/3) :

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>

- **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 with "Cloud Concept"; 3rd Gen.: Multistore/Polystores Systems

  Characteristics =<Elasticity, Fault-Tolerance>

➔ **Evolution or Crossroad?**

V. Conclusion (2/3): Impacts of CDMS on Scientific & Social Aspects

1. Scientific Aspects (1/2)

"The Beckman Report on Database Research" [Abadi et al. 2016]

- "Many early Big Data Mana. Systems BDMS Abandoned of DBMS Principles (e.g. Declarative Programming and Transactional Data Consistency) in favour of Scalability/Elasticity & Fault-Tolerance on Commodity Hardware" .

- "The latest generation of DBMS is rediscovering the value of these principles and is adopting concepts and methods…"that have been mastered by the DB Community DBC.
  - ➡ "Building these systems on these principles, the DBC is well positioned to drive improvements …..." .

V. Conclusion (3/3): Impacts of CDMS on Scientific & Social Aspects

2. Scientific Aspects (2/2)

<..., Concepts, Approaches, Methods, Tech./Tools> &<Applications>

- **New "Concept" introduced by the Cloud Computing CC**
  In terms of: <Data Models, DM Languages> ?

  ➡ **Economic Models in the Resource Management (Elasticity)**
  Rationalization & Cost-effectiveness!

3. Feedback from Industry and Institutions: Evaluation of benefits and social impacts ?

Thank you for your attention

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

Contact: Hameurlain@irit.fr

Informatics Research Institute of Toulouse IRIT

Pyramid Team

Paul Sabatier University
Toulouse , France