Pyramid

Evolution of Data Management Systems:

from Uni-processor to Large-scale Distributed Systems

Prof. Abdelkader Hameurlain

<Hameurlain@irit.fr>

Institut de Recherche en Informatique de Toulouse IRIT

Pyramid Team

Paul Sabatier University
118 Route de Narbonne
31062 – Toulouse , France

Outline
Evolution of Data Management Systems

▶ **Objective of Talk: <Why, Introduced Concepts, Relationship>**

- **File Management Systems**

- **Uni-processor Rel. DB Systems DBMS [Codd 70]**
- **Parallel DBMS [Dew 92, Val 93]**
- **Distributed DBMS [Ozs 11]**

- **Data Integration Systems [Wie 92]**

  Characteristics =<Distribution, Heterogeneity, Autonomy>

  ➡ **<Stable Systems, Not Scalable>**

- **Data Grid Systems [Fos 04]**

  Characteristics =<Large-scale, Unstable Systems (Dynamics of Nodes)>

- **? Data Cloud Systems [Agr 10/11/12, Chaud 2012, Sto 10]**

Main Problems of Data Management [Ozsu 11, Sto 98, …]

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

➔ Evolution of Query Processing & Optimization Methods

0.1 File Management Systems FMS (1)

- **File Concept**

  ➡ **Program and Storage Device Independence**

  [Storage] <File> [Program/Application]

  ▶ **File Manag.ement System**

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

0.2 File Management Systems (2)

- **Access Methods AM**
  - **Sequential AM**
  - **Key AM :=<Indexed/Hashing> AM**

- **Drawbacks of FMS**
  - **Data description must be done in each program**
  - **Relationships between files are materialized (New Files)**
    - **Software Eng. Requirements**

➡ **Database Concept**

▶ **Data Independence : <Physical & Logical> Indep.**

1.1 Database DB and DBMS

- **Concept of Database DB: Main Characteristics**
  - **Structured Data: Data Model Definition**
    - ➡ **Hierarchical/Network/Object/Relational Model**
  - **Stored Data on Disk: I/O Management**
    - ➡ **Query Processing & Optimization**
  - **Shared Data: Concurrency Control (Transactions, ...)**

- **Data Model DM:**
  - **What is the Objective of a DM?**
  - **What is the Wealth of a DM?**

➔ **Relational DBMS [Codd 70]**

1.2 Uni-proc. Relational DBMS

- **Relational Languages**
  - **Relational Algebra RA [Codd 72]: Basic Operations & Additional Operations**
  - **Fundamental Characteristics of RA:**
    - **Internal Law:** **Opi (Ri, [Rj])= Relation ➡ Querying Language**
    - **Commutative:** **R1xR2= R2xR1 ; SJ=JS**
  - **Algebraic Language/Rel. Algeb. Expression: P(S(J(Emp, Dept, ◷), c1), attr)**
  - **Declaratives Languages: SQL [Cham 76], QUEL [Sto 76], QBE [Zlo 77]**
    - **Specify "What do you want" ?**
    - **Without to specify "How to obtain the result" ?**

▶ **The System will find the Optimal Access Path**

➡ **Optimizer**

1.3 Uni-proc. Rel. DBMS: Query Processing

SQL Query Processing Phases

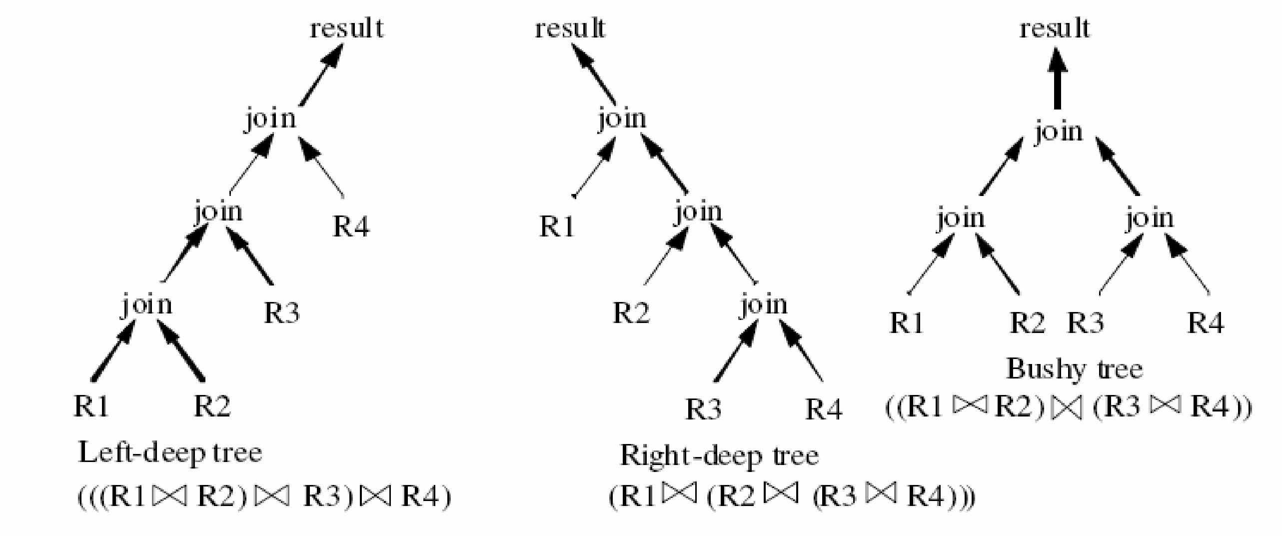
- **Decomposition: Syntax, Semantic, Authorization Control by using Metadata**

  ➡ Algebraic Tree*

- **Optimization: Generating an *Optimal Execution Plan* by using a *Cost Model***
- **Execution**

  ➡ Query Optimization Problem ?

- **Nature of execution plans (Data Structures): {LDT, RDT, BT}**



1.4 Uni-proc. Rel. DBMS: Query Optimization [Sel 79, Wong 76]

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

  q ∈ Query , p ∈ {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) et Min (Optimization Cost)**

- **Optimizer Structure= < Sp, C, St> [Gan 92]**
  - **Sp: Search Space**
    - **Data Structures: Linear Spaces, Bushy Space**
    - **Type/Nature of Queries**
  - **C: Cost Model**
    - **<Metrics, System Environment Description>**
  - **St: Search Strategies**
    - **<Physical Optim., Parallelization, Global Optim., …>**

1.5 Uni-proc. Rel. DBMS (1) : Query Optimization Methods

Optimization Process

- Logical Optimization: Rewriting of Algebraic Tree
- Physical Optimization [Swa 88, Ioa 89, Lan 91, ...]: Scheduling of Joins

  S1: Choice of appropriate algorithms for each relational operator

  S2: Scheduling of Joins: 2 Main Approaches

  - **Enumerative Search Approach**: <Breadth-First, Depth-First>
  - **Random Search Approach**: <Iterative Improvement, Simulated Annealing>
- Comparative Studies: intra-approach & inter-approach [Swa 89, Lan 91, ...]

  ➔ Advantages & Drawbacks : <Type of Queries, Size of Search Space>

  - Response Time (Optimal Execution Plan)
  - Optimization Cost

1.6 Uni-proc. Rel. DBMS : Query Optimization Methods

- **Limitations of (Uni-processor) Query Optimization Methods wrt <Decision Support Systems>**

- **Complex Queries*: Number of Joins >6 ?***
- **Size of Research Space [Tan 91]: *Very Large (e.g. 2^{N-1})***
- **Optimization Cost: *can be very expansive***
- **Optimal Execution Plan: *not guaranteed***

➡ ***Requirement in Hight Performance HP (e.g. Response Time)***

➡ **Introducing a New Dimension:**

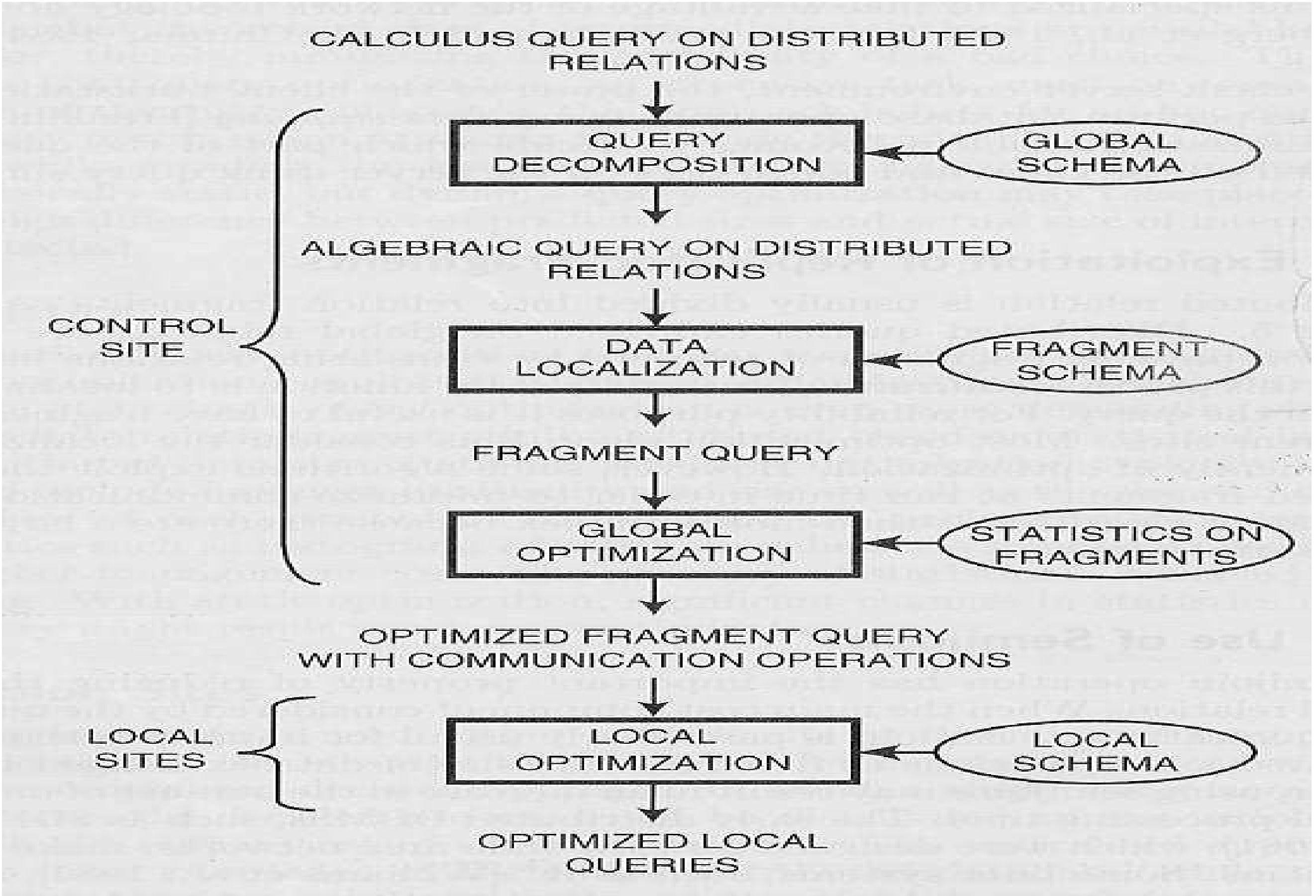
Parallelism (Mutli-processor Architecture)

2. Parallel Relational DB Systems

3. Distributed DB & Distributed Query Processing DQP (1)

Objective : Location/Fragmentation/Replication Transparency

Principle of DQP [Ozsu & P. Vald. 11]



3.1 Dist. Query Processing (1): Principles [Kos 00, Ozu 11, Sto 96]

- **Data Localization:**
  - **Fragmentation of Relations: Horizontal, Vertical, Hybrid**
  - **Location sites**
  - **Replication sites**
  - ▶ **How can we choose a relevant fragmentation strategy?**

- **Data Dictionary (Meta-data on DDB):**
  <Centralized, Replicated, Distributed>

- **Fragment Allocation : Alloc : F → S / ∀f ∈ F, ∃ s ∈ S, Alloc (f)= s**

  ▶ **What are the main parameters which impact on "Alloc" function?**

3.2 Dist. Query Processing & Optimization: Principles [Kos 00, Ozu 11, Sto 96]

- **Distributed Join Algorithms [Chiu 81, Val 81/84]:**
  - **Direct Join**: **R(Site1) Join S (Site2); Transfer the smaller relation**
  - **Semi-Join based Join**: **=<Project; Semi-Join; Join>**

    ➡ **Reducing Communication Costs**

- **Global Optimization**
  - ***Determining the optimal execution site*** **for each local sub-query considering data replication**
  - ***Scheduling of inter-site operators*** **minimizing a cost function**

    F = (CPU +I/O) + Comm

    ➡ **Reducing the Data Volumes Exchanged on the Network**

- **Local Optimization**
  - **Physical Optimization (Uni-processor Env.)**
  - **Parallelization (Parallel Env.)**

3.3 Distributed Query Optimization : Methods

- **Static Methods [Ber 81, <Loh 85, Mac 86>, Sto 96]: SDD-1, R*, Mariposa**
  - ▶ **Optimization of Inter-site comm. costs: <Direct Join, Semi-join based Join>**
    - **Direct Join:** ➔ **Minimizing the Data Volume Transferred Between 2 Sites**
    - **Semi-Join based Join** : ➔ **Reducing Communication Costs**
      - **+ Flexibility to Optimizers**
      - **- Increasing : <Size of Search Space & Local Processing Cost>**
    - ▼ **Strong Assumption (Cost Models) : Uniformity of Processors and Network**
  - ▶ **Mariposa DDBMS [Sto 96]: Economic Model based on "Bid" Principle**
    1. **Each Q decomposed into SQ_1, SQ_2, …, SQ_n**
    2. **For each SQ_i (i= 1..n) ---→{C_{i1}, C_{i2}, …C_{ij}} from j sites**
    3. **The broker notifies the winner site**
       - **based on the local Cost Models**
       - **Heterogeneity : Processors & Workload**
- **Dynamic Methods [Evren 97, Ozcan 97]** ➔ **[Kab 98]**

3.4 From Heterogeneous Dist. DB to Data Integration Systems (through Federated DB)

- **Heterogeneity**
  - Models ==> Pivot Model (e.g. Relational Model)
  - Semantic Conflicts (Integration of DB Schemes)
  - Servers (e.g. local DBMS, Processors, ...)

- **Autonomy of Data Sources**

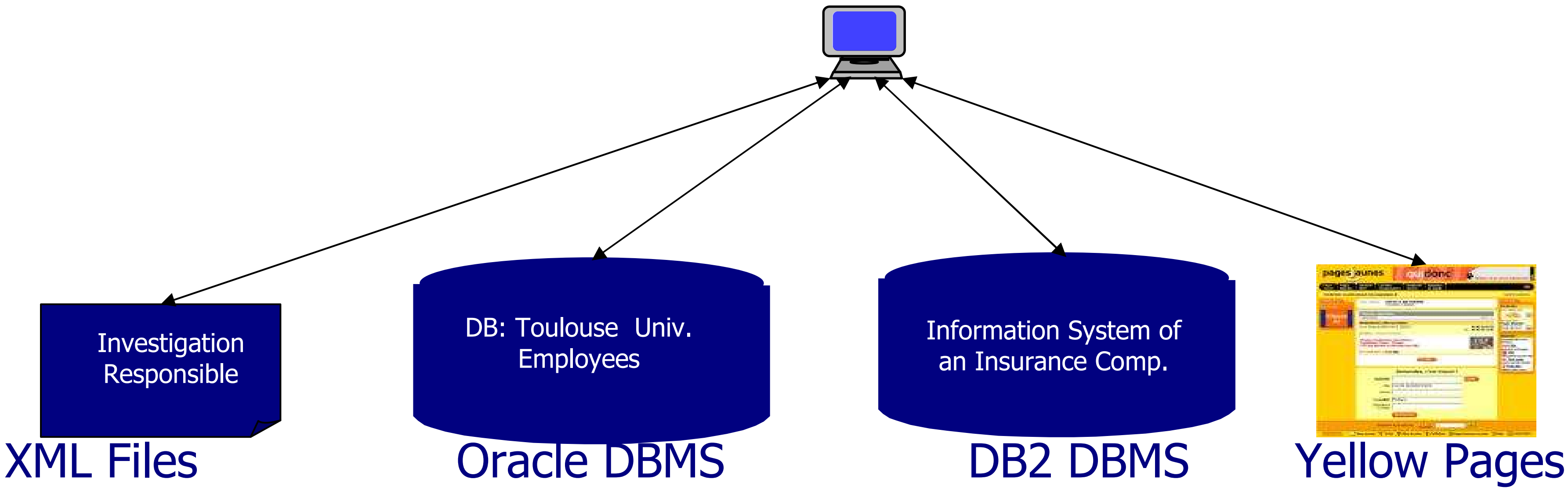
▶ **New requirements regarding to Data Sources**

- Data Sources can be structured in **different models** (Autonomy!)
  <Files, XML Files, Relational DB, Object BD, ...>

➡ **Virtual Data Integration Systems**: **Mediator-Wrappers** [Wied 92]

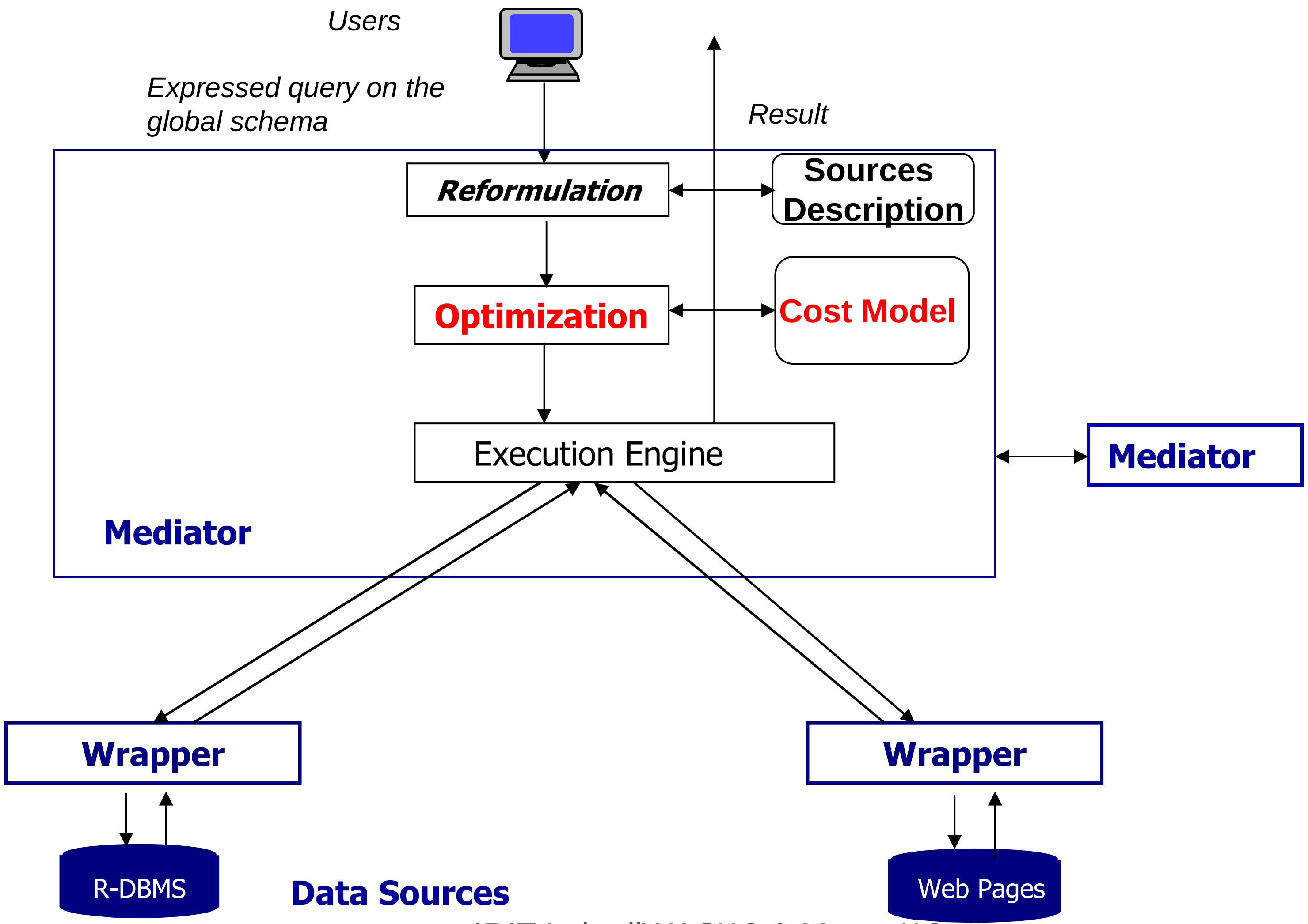
4.1 Data Integration System DIS (1)

- **Requirement**: **Data Integration arising from Several Sources**
- **Characteristics**: **<Distribution, Heterogeneity, Autonomy>**
- **Objectives**: **Efficient, Transparent & Uniform Access to Data Sources**



➡ Mediator-Wrapper Architecture [WIE 92]

4.1 Data Integration System DIS (2): Query Processing Steps



4.1 DIS (2): Query Processing & Special Operator

- **Main DIS:**
  - **Tukwila [Ive 99,], Tsimmis [Raj 95], Garlic [IBM, Haas 97], Disco [Tom 98], Le Select & Agora [Man 01], Piazza* [Hal 04], ...**
  - **Information Integrator (IBM), OLE-DB-Net (MS)**

- **Restricted Data Sources & DJ Operator**

  ➔ **New operator has been introduced [Gol 00, Man 02]**

  Dependent Join Operator: $\overrightarrow{\bowtie}$

$$\text{DAccess}\,(R(X^{b}, Y^{f}))\chi = \sigma\, X \in \chi(R(X, Y))$$

$$T \leftarrow \text{Scan}(R1(U^{f}, V^{f}))\ \overrightarrow{\bowtie}_{V=X}\ \text{DAccess}(R2(X^{b}, Y^{f})).$$

  - **Asymmetric Nature:** ➔ **Search Space is Reduced**
  - **Problem to Capture Valid Execution Plans [Man 02, Yerneni 99]**

- **Query Optimization Methods in DIS [Ams 98, Iv 99, Av 00, Arc 04, Oza 05, ...]**

4.2 Query Optimization Methods

- **Type of Query Optimization Methods [Cole 94]: <Static, Dynamic>**

  - **Principle of Dynamic Query Optimization [Hel 00]:**

S1: **Sub-optimality Detection of execution plans**

S2: **Modification of Sub-optimal Exec. Plans**

S3: **Repeat S1 & S2**

4.3 Dynamic Query Optimization: a State of the Art

- **Characteristics of Dynamic Query Optimization**
  - **Environment**: **uni-processor**, **Parallel** [Bru 97, Hon 92], **Distributed** [Bou 00, Ive 04], **Large Scale** [Jon 97, Arc 04, Oza 05, Gou 05/09, Da Sil 06, Liu 08,...]
  - **Type of Event**: **Estimation Errors** [Bru 97, Kab 98], **Delays in Data Arrivals Rates** [Ams 98, Bou 00]
  - **Modification Types**: **Re-optimization** [Bab 05, Bou 00], **Re-scheduling** [Ozc 97]
  - **Modification Level**: **Intra-operator** [Des 03], **inter-operator** [Ams 98]
  - **Nature of Decision-making**: **(by the Optimizer)**
    - **Centralized Control** [Kab 98, Ams 00, Bou 00, Mar 04, Da Sil 06, ...]:
      Unique Process (Optimizer) is charged to supervise, detect, and modify an execution plan
    - **Decentralized Control** [Ive 99, Urh 00/01, Jon 97, Arc04, Oza 05, Erg 07, ...]:
      Detection & modification are made by Several Process

➔ **<Large Scale Env., Nature of Decision Made by the Optimizer>**

4.4 Dynamic Query Optimisation Methods in DIS (3) :
[Am 98, Iv 99, Av 00, Urh 00/01/, Arc 04, Oza 05, ….]

- **Large Scale Environment**
  - **High number of data sources, users, & computing resources: <Distributed, Heterogeneous, Autonomous>**
  - **Low bandwidth and strong latency (➔ bottleneck)**
  - **Huge volume of data (GB → PB)**

- **Nature of Decision-making:** (**by the Optimizer**)
  - **Centralized Control : Unique Process (Optimizer) is charged to supervise, detect, modify an execution plan**
  - **Decentralized Control : detection & modification by several process**

4.5 Dynamic Query Optimization Methods in DIS (1)

- **Why dynamic strategies have been decentralized?**
  - **Reducing Delays in Data Arrival Rates [Ams 98, Ive 99]**

    Two events are possible:
    - **Initial delay before the arrival of the first tuple**
    - **Bursty arrival: the data arrive in bursts**
  - **Reducing Network Traffic: Distant Interactions → Local Interactions**
  - **+ Large Scale Environment**

    ➨ **Bottleneck (Huge Volume of Data, Low Bandwidth and Strong Latency of Network, High Number of Users/Nodes)**

    ➨ **Minimizing the number of control messages**

4.5 Dynamic Query Optimization Methods in DIS (2)

■ **Why Dynamic Strategies (// & Dist.) do not Scale?**

- **Centralization of the Decision** (made by the optimizer): ➡ **Bottleneck**
- **Inaccuracy & <u>Obsolescence</u> of Estimations:**

  <Statistics, Predicate selectivity, Cost of Operators>
- **Unavailability of Resources: = <{CPU, Mem., Network, Data Sources}>**

  {**Overloading Site, Complex Sub-query, PB with Net.: Comm. Or Bandwidth**}

➡ <Decentralized, ➡Scalable>

▶ **What are the Advantages of Mobiles Agents in Dynamic Query Optimization? [Jo 97, Arc 04, Oza 05, Mor 10, …]**

4.6 Summary of Dynamic Query Optimization Methods in DIS

Nature of Decision-making → Motivation & Context ⇨	Centralized Control	Decentralized Control
Reducing Delays in Data Arrival Rates	Inter-op.: [Ams 98, Bou 00] Intra-op. : [Av00/Eddies; Ive04]	Intra-op.: [Iv99; Tukwila/DHJ] [Urh 00/01, XJoin]
Large Scale	X?	Mobile Agents: Inter-op. : [Jones 97] Intra-op.: [Arc 04, Oza05, …]

Nature of Decision-making: Centralized/decentralized Control
Motivations : Reducing Delays in Data Arrival Rates; Large Scale,
Modification Level: Intra-operator, Inter-operator

4.7 From Data Integration Systems DIS to Data Grid Systems

- **Characteristics of DIS: <Distribution, Heterogeneity, Autonomy>**
  - **DIS have been Extended to Large Scale LS Environement**

- **Assumption: Stability of Systems (from Uni-proc DBMS to LSDIS)**
  - ► **Strong Assumption in a Large Scale Env.**

- ➡ **Constraint: Unstability of Systems US ("Dynamicity" of Nodes)**
  - ► **A node can join, leave or fail at any time**

➡ **Characteristics of New Systems : < Dist. , Hete., Autono., LS, US>**

➡ **Grid Systems**

5. Grid Systems: Definition, Charact. & Evolution

- **Grid Systems GS [Fos 04]:**
  1. **"Coordinated Distributed Computing Infrastructure"**
  2. **" Resource sharing & coordinated problem solving in dynamic, multi-institutional virtual organizations "**

  ▶ **Distributed Systems with 2 Characteristics:**
  - **Large Scale (including: <Autonomy, Heterogeneity>)**
  - **Unstability of Systems ("Dynamics" of Nodes)**

  ➡ **Exploiting of Available Resources:**

  < CPU, Memory, Network & I/O Bandwidth, Data Sources, Services>

- **Evolution: From Grid Systems ➡ Data/DB Grid Systems [Taniar et al. 2008]**
  ➔ **New Problems of Data Management in GS [Pac 07]**

5.1 Data Management DM in GS (1): Main Problems

- **Main Problems of Data Management in GS [Pac 07]**

  - **Resource Discovery & Selection**
  - **<u>Query Processing & Optimization</u>**
  - **Monitoring Services**
  - **Replications & Caching**
  - **Cost Models**
  - **Autonomic Data Management (self-tuning, self-repairing, …)**
  - **Security Issues, …**

  ➔ **<u>Focus on Query Processing & Optimization</u>**

5.2 Data Management in GS (2): Distributed Query Optimization

- **Limitations of Dyn. Opt. Methods Developed in //, Dist. , DIS Systems:**
  - **Large Scale: ➔ Decentralized Control**
    ➔ +[Self-adaptable] : based on Paradigm of Mobile Agents
  - **"Dynamicity" of Nodes (Unstable System):**
    ➔ Insure a Permanent Access to Resources
- **Challenges**:
  - **Extend or design new: approaches and methods**
  - **Characteristics of proposed methods:**
    - **<Dynamic, Efficient, Scalable, Self-adaptable>**
  - **Development of large scale experimental platforms**
    - **Validation**
    - **Learning lessons**

5.3 Dynamic Query Optimization in Data Grid Systems (1)

[Gou 04a/04b/06, Cyb05, Bose 07, Da Sil 06/Porto 06, Liu 08, Soe 05, …]

- **Context: = <Large Scale, Unstable> + <Heterogeneity, Autonomy>**

  Heterogeneity of resources (CPU, Mem., Bandwidth Net. & I/O, Data sources)

- **2 Approaches based on :**
  1. **Extended "Classic" Approach: using Global Cost Model)**
  2. **Incentive-based Approach (Eco. & Reputation): using Local Cost Models**
- **Objective of Methods**
  - **Efficient Resource Allocation/Scheduling Methods**
  - **Integration of Parallelism Dimension**
  - **Heterogeneity consideration**
- **Constraint of Proposed Methods:**
  - **Forms of parallelism: {Partitioned, Independent, Pipelined Parallelism}**
  - **Communication Cost and Load balancing**

➡**Challenge: In this context, how can we describe a Cost Model ?**

5.3 Dynamic Query Optimization in Data Grid Systems (2)

- **Challenge : How can we describe accurately the system envir environment in a such context?**
  - **The quality of the predictability of virtual/physical resources has a strong impact on the decision made by the optimizers**

- **Context: = <Large Scale, Unstable> + <Heterogeneity, Autonomy>**

- **System Env. =<Profile of Sources, Charact. of CR, Cost Formula>**

  - **➡Difficult to obtain statistics and relevant info. on resources**
  - **➡Definition and Validation of Cost Formula**
    - **Large Scale: Obsolescence of Values**
    - **Heterogeneity of Data Sources and Nodes ➔ Inaccuracy of Estimations**
    - **Autonomy of Data Sources ➔ do not export all statistics**

6. Cloud Computing/Systems & Data Management (1)
[Ston 10, Agr 10/11, Chaud 12, Zhou 12, Kald 12, ...]

- **Main Characteristics of Cloud [D. Agrawal et al. 2011]**
- **Data Management in the Cloud: Main Problems [Chaud 2012]**
- **"Hot Debate" on: MapReduce versus Parallel DBMS: friends or foes**
  [M. Stonebraker et al., 2010, D. Agrawal et al. 2010, S. Chaudhauri 2012]
- **"Reconciling Debate" [Zhou 2012, Kaldewey et al. 2012/EDBT]**
  "SCOPE : Parallel Databases Meet MapReduce"
  [Zhou et al. June 2012, VLDB Journal]

6.1 Main Characteristics of the Cloud [Agrawal et al. 2011]

- **Scalability**
- **Elasticity [Ozu 11]**

  « The ability to scale resources out, up, and down dynamically to accodomate changing conditions »
- **Fault-Tolerance**
- **Self-Manageability: Self-Tuning, Self-Reparing**
- **Ability to run on Commodity Hardware**

6.2 Data Management in the Cloud: Main Problems [Chaud 2012]

- **Data Privacy: <Access Control, Auditing, Statistical Privacy)**
- **Approximate Results**
- **Query Optimization (mainly from MR heritage)**
  - **Optimization of User-Functions ?**
  - **Difficult to Construct Statistics on Volatile Data**
  - **Extensive Materialization (I/O)**
  - **Data Redistribution**
- **Performance Isolation for Muti-tenancy**

.....

6.3 Hot Debate: MR versus // DBMS

- "MapReduce and Parallel DBMSs: friends or foes?"

  [Stonebraker et al. 2010 Com of the ACM, Jan. 2010, Vol 3. No. 1]

  ◆ The performance results (between Hadoop (MR) and 2 // DBMSs) show that the DBMSs are substantially faster than the MR system once the data is loaded.

  ➡ Conclusion: "MR complements DBMSs since DB are not designed for ETL Extraction-Transform –Load tasks, a MR specialty "

- "Big Data and Cloud Computing: New Wine or just New Bottles?"
  [Agrawal 2010 et al. , Univ of California/SB] VLDB'2010 Tutorial

- "An Interview with S. Chaudhuri"; [Sept. 2012, XRD, Vol. 19, No. 1]

  "If I were to look at recent research publications, a disproportionately large fraction of them are focused on solving for MapReduce platforms the same problems we addressed for parallel database systems. We can and should do much more."

6.4 Reconciling Debate (1) [Zhou 2012, VLDB Journal, Kald 2012]
"SCOPE: Parallel (//) Databases meet MapReduce"

- **Objectives : combines benefits from execution engines**
  - **Parallel DB Systems**

  & ➔ **for Large Scale Data Analysis**
  - **MapReduce**

  ➔ **<Easy Programmability, Massive Scalability, HP (adv. Optimization)>**

- **Advantages of // DB Systems**
  - Relational Schema
  - Declarative Query Lang.
  - Sophisticated Query Optimizers {Partitioned, Independent, Pipelined //}

- **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 : Redistribution of Data

6.4 Reconciling Debate (2) [Zhou 2012, VLDB Journal, Kalde 2012]

"SCOPE: Parallel Databases meet MapReduce", MS

"Clydesdale: Structured Data Processing on MapReduce", IBM & Google

Advantages of MR

- Scaling Very Well (to massive data sets)
- Fault - Tolerance
- Mechanism to achieve Load-Balancing
- Support the Parallelism

Weakness of MR: Side Applications

Users :

- 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

- + Data Dependence
- + Extensive Materialization (I/O)

SCOPE Proposals (Structured Computations Optimized for Parallel Execution)?

6.4 Reconciling Debate (3) [Zhou 2012, VLDB Journal;]
"SCOPE: Parallel (//) Databases meet MapReduce"

SCOPE Proposals :

- **SCOPE uses MapReduce as Target Language**
- **Data Representation: Structured Streams and Unstructured Streams**
- **Query Language (DDL, ML) : Declarative Scripting Language (close to SQL)**
  - **Specific operators (close to Wrapper in DIS)**
    - ➨ **Data Independence (main Objective of DB!)**
  - **SQL-like extensions (Relational Operators + Aggregate Functions)**
- **Cost-based Optimizer (with few statistics!; Opp. Rule-based Optimizer)**
- **Fault-Tolerance (MR)**
- **Load-Balancing (MR)**
- **- Extensive Materialization (MR)**

7. Summary & Conclusion
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>**

- **Data Grid Systems [Fos 04, Pac 07]:** *Exploiting of Available Resources =*

  <Computing Resources CR, Data Sources, Services>

  Characteristics =<Large-scale, Unstable Systems (Dynamics of Nodes)>

- **? Data Cloud Systems:** *Elasticity , HP Isolation*

  [Sto 10, Agr 10/11/12, Chaud 12, Kald 12, Zhou 12,...]:

8. 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**

8. 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.
- …

8. References: Data Integration Systems DIS

- G. Wiederhold ; *Mediators in the Architecture of Future Information Systems*, Journal of IEEE Computer, IEEE Computer Society, 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 Intl. Conf. on Management of Data, 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).

8. References: Large Scale DIS & Mobile Agents

▶ Decentralized Control, Large Scale & Mobile Agents

- **A. Fuggetta, et al., *Understanding Code Mobility*. IEEE Trans. on Software Engineering Vol. 24, No. 5, 1998, pp. 342-361.**
- **R. Jones and J. Brown, *Distributed Query Processing via Mobile Agents*, On-line paper, http://www.cs.umd.edu/~rjones/paper.html. 1997**
- **A. Sahuguet, et al. ; *Distributed Query Optimization: Can Mobile Agents Help?*, accesible via : http://www.seas.upenn.edu/~gkarvoun/dragon/publications/sahuguet/, 2000.**
- **J.-P. Arcangeli, et al., et al., *Mobile Agent Based Self Adaptive Join for Wide-Area Distributed Query Processing*, in: Intl. Journal "Database Management", IGP, Vol.15, No. 4, Oct. 2004, pp. 25-44.**
- **A. Hameurlain, *Special Issue on Mobile Databases*, In: Intl Journal "Computer Systems Science and Engineering", March 2005.**
- **B. Ozakar , et al., *Mobile Join Operators for Restricted Sources*, in: Intl Journal Mobile Information Systems, IO Press, 1(3), 2005, pp. 167-184.**
- **B. Ergenc, et al., Robust Placement of Mobile Relational Operators for Large Scale Distributed Query Optimization. In : Intl. Conf. on Parallel and Distributed Computing, Applications and Technologies (PDCAT 2007), Adelaide, Australie, IEEE Computer Society, December 2007, pp. 227-235.**

8. References: Grid Systems & Resource Discovery (1)

- **I. Foster and C. Kesselman. The Grid: Blueprint for a New Computing Infrastructure. Morgan Kaufmann Publishers, 2004.**
- **P. Trunfio, et al., "Peer-to-Peer resource discovery in Grids: Models and systems", Future Generation Computer Systems, Elsevier, 2007, Vol. 23, N. 7, pp. 864-878.**
- **E. Pacitti, et al., "Grid Data Management: Open Problems and News Issues"; Intl. Journal of Grid Computing; Springer, 2007, Vol. 5, N. 3, pp. 273-281.**
- **D. Taniar et al., "High Perf. Parallel DB Processing and Grid Databases", Ed. Wiley, 2008.**
- **+ Mastroianni 05/08, Ranjan 08, Hameurlain et al. 2010..…. *(Overview papers)***

▶ Centralized/Hierarchical Structures

- **M. Antonioletti, et al., "The design and implementation of Grid database services in OGSA-DAI", Concurrency and Computation: Practice & Experience , Vol. 17, 2005, pp. 357–376.**
- **K. Czajkowski, et al., "Grid information services for distributed resource sharing", Proc of the 10th IEEE Symp. on HPDC, IEEE CS, 2001, pp. 181-194.**
- **C.Huang, et al., "Dart: A Framework for Grid-Based Database Resource Access and Discovery", LNCS, Book of Grid and Cooperative Computing , Vol. 3033, pp. 855-862.**
- **T. G. Ramos, et al., "An Extensible Resource Discovery Mechanism for Grid Computing Environments", Proc. Of the 6th IEEE Intl. Symposium on Cluster Computing and the Grid, 2006, pp. 115-122.**

…

8. References : Resource Discovery (2)

▶ P2P Techniques

- M. Cai et al., "MAAN: A Multi-Attribute Addressable Network for Grid Information Services", Proc. of 4th Intl. Workshop on Grid Computing, 2003, pp. 184 – 191.
- S. Cheema, et al., "Peer-to-peer Discovery of Computational Resources for Grid Applications", Proc. of Grid Computing Workshop, IEEE CS, 2005, pp. 179-185.
- M. Marzolla et al., "Peer-to-Peer for Discovering resources in a Dynamic Grid ", Journal of Parallel Computing, Vol. 33, N. 4-5, 2007, pp. 339-358.
- P. Trunfio, et al., "Peer-to-Peer resource discovery in Grids: Models and systems", Future Generation Computer Systems, Elsevier, 2007, Vol. 23, N. 7, pp. 864-878.

▶ Mobile Agents

- I. Foster, N. R. Jennings, and C. Kesselman. Brain meets brawn: Why grid and agents need each other. In *AAMAS '04: Proceedings of the 3rd Intl. Joint Conference on Autonomous Agents and Multi-agent Systems*, pages 8–15. IEEE, 2004
- J. Cao, et al, "Agent-Based Resource Management for Grid Computing", 2nd IEEE/ACM International Symposium on Cluster Computing and the Grid, Berlin, 2002, pp.350-351.
- S. Ding et al. "A Heuristic Algorithm for Agent-Based Grid Resource Discovery", Intl. Conf. on e-Technology, e-Commerce and e-Service, IEEE, 2005, pp. 222-225.
- G. Kakarontzas, I. K. Savvas, "Agent-Based Resource Discovery and Selection for Dynamic Grids", Proc of the 15th IEEE Intl. Workshops on Enabling Technologies, 2006, pp. 195-200.
- M. Yan, et al, "Study of Grid Resource Discovery Based on Mobile Agent", Proc. of the 3rd Intl. Conf. on Semantics, Knowledge and Grid, 2007, pp. 570-571.
- Yu et al., "Grid Resource Management Based on Mobile Agent", Proc. of the Intl. Conf. on Computational Intelligence for Modeling Control and Automation and Intl. Conf. on Intelligent Agents Web Technologies and Intl Commerce, 2006, pp. 255-256.

8. References: Distr. Query Optimization in Grid Systems (1)

OGSA-DQP, Parallelism & Scheduling

- **D. Taniar et al., "High Perf. Parallel DB Processing and Grid Databases", Ed. Wiley, 2008 (Book)**
- **A. Gounaris, A Vision for Next Generation Query Processors and an Associated Research Agenda, Invited Paper in Globe 09, LNCS 5697, Linz, pp. 1 – 11, 2009**
- **A. Gounaris, et al. "Adaptive Query Processing and the Grid: Opportunities and Challenges", In: Proc. of the 15th Intl. Dexa Workhop, IEEE CS , 2004, pp. 506-510.**
- **A. Gounaris, et al., "Resource Scheduling for Parallel Query Processing on Computational Grids", Proc. of the 5th IEEE/ACM Intl. Workshop on Grid Computing, 2004, pp. 396-401.**
- **A. Gounaris, et al., "Adapting to Changing Resource Performance in Grid Query", Proc. Of the Intl. VLDB Workshop (Data Management in Grids), LNCS 3836, 2005, pp. 30–44.**
- **N. Hu, et al. , "Dynamic Optimization of Sub Query Processing in Grid Database, Natural Computation", Proc of the 3rd Intl Conf. on Natural Computation, Vol. 5, 2007, pp. 8-13.**
- **J. Smith, et al., "Distributed Query Processing on the Grid", Proc. of the Third Workshop on Grid Computing, GRID 2002, LNCS 2536, 2002, pp. 279–290.**
- **M. N. Alpdemir et al., "Using OGSA-DQP to Support Scientific Applications for the Grid", Proc. of SAG 2004, LNCS 3458, 2005, pp. 13-24.**

8. References: Distributed Query Optimization in GS (2)

OGSA-DQP, Parallelism & Scheduling

- M. Antonioletti, et al., "The design and implementation of Grid database services in OGSA-DAI", Concurrency and Computation: Practice & Experience , Vol. 17, 2005, pp. 357–376.
- Gounaris, et al., "Resource Scheduling for Parallel Query Processing on Computational Grids", Proc. of the 5th IEEE/ACM Intl. Workshop on Grid Computing, 2004, pp. 396-401.
- S. K. Bose, et al., "Allocating Resources to Parallel Query Plans in Data Grids", Proc. of the 6th Intl. Conf. on Grid and Cooperative Computing, Vol. 00, 2007, pp. 210-220.
- Da Silva, et al.,"An adaptive parallel query processing middleware for the Grid", Concurrence and Computation: Pratique and Experience, Wiley InterScience, Vol. 18, 2006, pp. 621–634.
- S. Liu, H.A. Karimi, "Grid query optimizer to improve query processing in grids", Future Generation Computer Systems, 2008, in Press.
- F. Porto, et al., "An Adaptive Distributed Query Processing Grid Service", Proc. Of the Intl. VLDB Workshop (Data Management in Grids), LNCS 3836, 2005, pp. 45–57.
- K. M. Soe, et al.,"Efficient Scheduling of Resources for Parallel Query Processing on Grid-based Architecture", Proc. of the 6th Asia-Pacific Symposium, IEEE, 2005, pp. 276-281.

8. References: Cloud Computing & Data Management

- **J. Dean & Ghemawat; “MapReduce: Simplified Data Processing on Large Clusters”, OSDI’04**
- **M. Stonebraker et al., “MapReduce and Parallel DBMSs: friends or foes?” in: Com of the ACM, Jan. 2010, Vol 3. No. 1]**
- **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.**
- **A. Paramswaran, “An interview with S. Chaudhuri” , In: XRD Vol. 19, No. 1, Sept. 2012]**
- **S. Chaudhuri, What Next? A Half-Dozen Data Management Research Goals for Big Data and the Cloud, PODS’2012, May, Scottsdale, Arizona, USA, PP. 1-4**
- **D. Agrawal et al., “Big Data and Cloud Computing: Current State and Future Opportunities”, In: EDBT 2011, Tutorial, March, Uppsala, Sweden**
- **D. Agrawal & A. El Abbadi, “The evolving landscape of data management in the cloud”, In: IJCSSE, Vol. 7, No. 1, 2012, pp. 2-16.**
- **J. Zhou et al. , “SCOPE: Parallel Databases meet MapReduce”, In: VLDB 2012 Journal,**
- **T. Kaldewey et al., “Kladesdale: Structured Data Processing on MapReduce”, In: EDBT2012, pp. 15-25**
- **Campbell et al.; Cloudy Skies for Data management, ICDE’201**
- **Thakar & Szalay; Migration a large Science DB to the Cloud, HPDC’201**
- **Blanas et al. ; A Comparison of Join Alg. for Log Processing in MR, SIGMOD’2010**
- **Afrati & Ullman; Optimizing Joins in a MR Environment, EDBT’2010**

8. References: Cost Models

Approach: <Historical, Calibrating>

- **S. Adali, K., et al. , “*Query Caching and Optimization in Distributed Mediator Systems*”, Proc. of the 1996 SIGMOD Conf., pp. 137-148 *(Historical Approach)***
- **W. Du, et al. , “*Query Optimization in a Heterogeneous DBMS*”, VLDB 92, pp.277-291.**
- **G. Gardarin, et al. , “*Calibrating the Query Optimizer Cost Model of IRO-DB, an Object-Oriented Federated Database System*”, VLDB96, pp. 378-389**
- **S. Ganguly et al. “*Efficient and Accurate Cost Models for Parallel Query Optimization*”, in: *PODS’96* pp. 172-182.**
- **Y. E. Ioannidis and S. Christodoulakis, “*On the Propagation of Errors in the Size of Join Results*”, ACM SIGMOD, 1991, pp. 268-277**
- **H. Naacke, et al. “*Leveraging Mediator Cost Models with Heterogeneous Data Sources*”, ICDE 98, pp. 351-360**
- **M. Tork, et al. “*Cost Models DO Matter: Providing Cost Information for Diverse Data Sources in a Federated System*”, VLDB99, pp. 599-610**
- **Q. Zhu, et al. , “*Cost Estimation for Queries Experiencing Multiple Contention States in Dynamic Multidatabase Environments*”, Knowledge and Infor. Systems 5(1): 26-49, 2003 *(Calibrating App.)***
- **B. Ozakar, et al., “*Query optimization: Mobile Agents versus Accuracy of the Cost Estimation*”, In: Intl Journal “Computer Systems Science and Engineering”, 20(3), May 05**
- **Y. Slimani et al., *An Adaptable Cost Model for Distributed Query Optimization on the Grid, in: GADA, LNCS 3292, 2004, pp. 70-87.***
- **M. Hussein, et al. ; “*Embedded Cost Model in Mobile Agents for Large Scale Query Optimization*”, 4th Intl. Symp. on // and Distributed Computing, IEEE , Lille, July 2005, pp. 199-206**

8. References: DB Research & Perspectives

- **R. Agrawal et al., The Claremont Report on Database Research, in: SIGMOD Record, Vol. 37, No. 3, pp. 9-19, 2008**
- **M. Stonebraker, Technical Perspective – one size fits all: an idea whose time has come and gone, in: Commun. ACM, Vol. 51, No. 12, 2008**

Thank you for your attention

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

Contact: hameurlain@irit.fr

Institut de Recherche en Informatique de Toulouse IRIT

Paul Sabatier University, Toulouse, France