# Data-intensive HPC: opportunities and challenges

*Patrick Valduriez*

# Big Data Landscape

Multi-$billion market!

No one-size-fits-all solution: SQL, NoSQL, MapReduce, …

No standard, except Hadoop

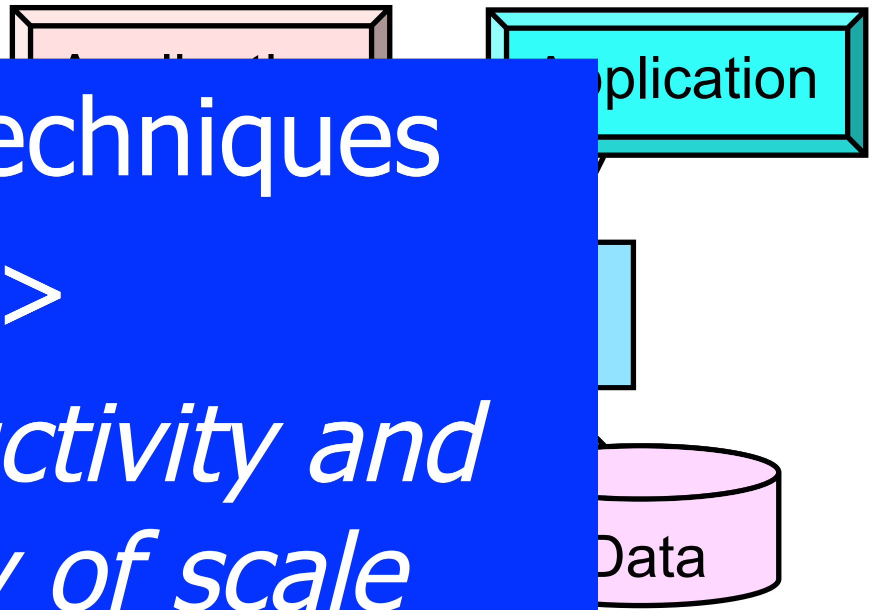
Keeps evolving

# Big Data Killer App.: *Analytics*

- Objective: find useful information and discover knowledge in data
  - Typical uses: forecasting, decision making, research, science, …
  - Techniques: data analysis, data mining, machine learning, …
- Why is this hard?
  - External data from various sources
    - Hard to verify and assess, hard to integrate
  - Low information density (unlike in corporate data)
    - Like searching for needles in a haystack
  - Different structures
    - Unstructured text, semi-structured document, key/value, table, array, graph, stream, time series, etc.
    - Hard to integrate
  - Simple machine learning models don't work
    - Reality and context matter
      - See "When big data goes bad" stories

# Data Management: basic principle

- *Data independence*
  - Hides implementation details
  - Provisi
    service
    - Sche
    - Quer
    - Opti
    - Auto
    - Inde
    - Tran
    - Consistency
    - Access control
    - …



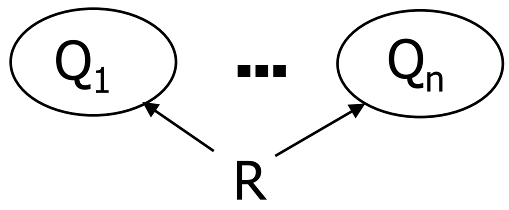
Generic techniques

=>

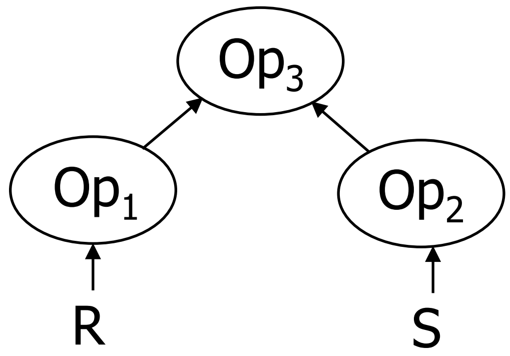
*User productivity and Economy of scale*

# Parallel data processing

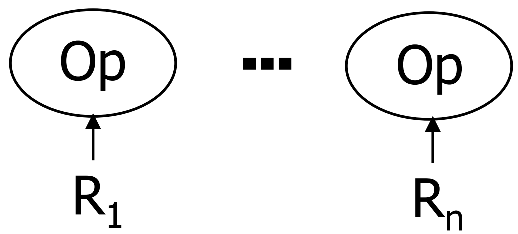
- Inter-query (task parallelism)
  - Different queries on the same data set (or data stream)
  - For lots of concurrent queries
  - e.g. Google search




- Inter-operation (task parallelism)
  - Different operations of the same query on different data sets
  - For complex queries
  - e.g. RDBMS

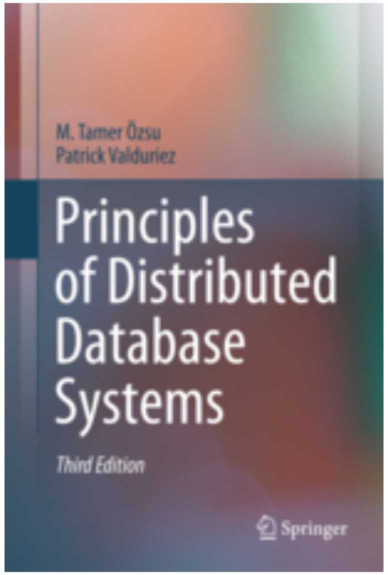


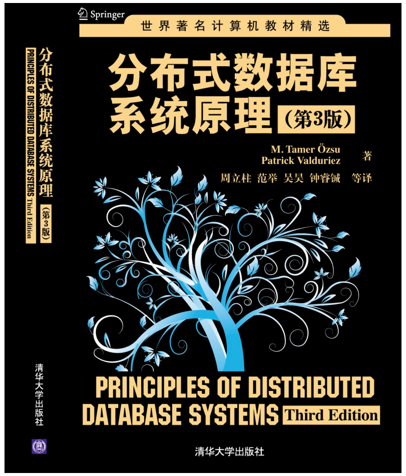

- Intra-operation (data parallelism)
  - The same operation on different data partitions of the same data set
  - For ops on big data sets
  - e.g. MapReduce



# (Structured) Database Systems

- Volume
  - *Parallel database systems*
    - E.g. Oracle Exadata database machine
- Velocity
  - *Data stream management systems*
    - E.g. Streambase
- Variety
  - *Data integration systems*
    - E.g. DB2 Information Integrator



# HPC: from simulation to big data analytics (BDA)

- Simulation: traditional HPC with big data
  - e.g. weather forecasting, climate modeling
- HPC applications need to deal with more and more data
  - More input data, e.g. from more powerful scientific instruments or sensor networks
  - More output data, e.g. with smarter mathematical models and algorithms
- Analytics: a newer, complementary market for HPC
  - Analytics methods applied to established HPC domains in industry, government, academia
  - High-end commercial analytics pushing up into HPC
  - Shorter journey from science to business, e.g. from cancer genomics to personalized medicine

# Some BDA Killer Apps

- Real-time processing and analysis of raw data from high-throughput scientific instruments
- Uncertainty quantification in data, models, and experiments
  - E.g. to measure the reliability of simulations involving complex numerical models
- Fraud detection across massive databases
  - Applicable in many domains (e-commerce, banking, telephony, etc.)
- National security
  - Signal intelligence (SIGINT), anomaly detection, cyber analytics
  - Anti-terrorism (including evacuation planning), anti-crime
- Health care/medical science
  - Drug design, personalized medicine
  - Epidemiology
  - Systems biology
- Social network analysis
  - Modeling, simulation, visualization of large-scale networks

# Cyber Analytics*


- Goal: identify malicious activity in high-throughput streaming data
  - More than 10 billion transactions/day
  - Tens of millions of unique IP addresses observed each month
  - Adjacency matrix may contain over a quadrillion elements but is sparse, with billions of values
  - Tens of TBs to PBs of raw data
  - Patterns can span seconds, months
- Current data analysis tools operate on thousands to hundreds of thousands of records

* J.R. Johnson. High Performance Computing for Data Intensive Science. 2014.

Pacific Northwest NATIONAL LABORATORY

# Risers Fatigue Analysis (RFA)*

- Context: joint project between Zenith, UFRJ and LNCC, Rio de Janeiro (and Petrobras)
  - Pumping oil from ultra-deep water from thousand meters up to the surface through risers
- Problems
  - RFA requires a complex workflow of data-intensive activities which may take like 40 hours (SGI Altix ICE 8200 with 64 CPUs Quad Core cluster)
    - Input: 1,000 files (about 1 GB)
    - Riser information, such as finite element meshes, winds, waves and sea currents, and case studies
  - Output: 6,000 files (about 100 GB)
  - 10 activities (dynamic analysis of risers movements, tension analysis, curvature analysis, merging data from previous activities, etc.)
- An algebraic approach (inspired by relational algebra) and a parallel execution model
  - 4-fold improvement versus hard-coded parallel workflow

* E. Ogasarawa, J. Dias, D. Oliveira, F. Porto, P. Valduriez, M. Mattoso. An Algebraic Approach for Data-Centric Scientific Workflows. VLDB 2011.

# BDA vs. HPC

| | BDA | HPC |
|---|---|---|
| Computing model | **Data-centric**: move tasks to data and **Reduce** | **Compute-centric**: move data to tasks and **Accelerate** |
| Data storage | Uniform storage (sharding) on disks | Hierarchical storage (disks, tapes, etc.) |
| Parallel file management | Designed for few big files, e.g. HDFS | Designed for many small files, e.g. Lustre |
| Programming model | Algebraic operators, e.g. MapReduce, Spark | MPI versus OpenMP |
| Languages | Java, Python, C++ | C, C++ |

- Opportunities
  - HPC: smarter file systems, with metadata and indexes; high-level programming frameworks, e.g. MapReduce-MPI
  - BDA: compute-centric nodes to run heavy tasks (machine learning)

# Challenges for Data-intensive HPC

- Data-intensive HPC = HPC + BDA
- Collaboration from two big research communities
  - Data management and HPC
  - Is it a billion$ market?
- Some hard problems
  - Architecture: how does it fit with IT, data centers and cloud? No one-size-fits-all solution
  - Data storage: how to combine uniform and hierarchical storage models?
  - Data movement between applications and tasks
  - HPC data-oriented programming frameworks
  - Tools for data cleaning, data integration, data analysis

# Thanks