Zenith: a hybrid P2P/cloud for Big Scientific Data

Esther Pacitti & Patrick Valduriez
INRIA & LIRMM
Montpellier, France

Basis for this Talk

Zenith: Scientific Data Management on a Large Scale.

- Esther Pacitti and Patrick Valduriez, ERCIM News 2012(89): (2012).

Panel Session: Social Networks and Mobility in the Cloud.

- Amr El Abbadi and Mohamed F. Mokbel (Moderators), VLDB, Istanbul, 2012.

Principles of distributed Data Management in 2020?

- P. Valduriez. Int. Conf. on Databases and Expert Systems Applications (DEXA), Keynote talk, Toulouse, 2011.

Outline of the Talk

- Big data
- Scientific big data
- Application examples
- Our approach
- P2P networks
- Cloud computing
- Hybrid P2P/cloud
- Examples of techniques

3

Big Data: what is it?

A buzz word!

- With different meanings depending on your perspective
  - E.g. 100 terabytes is big for an OLTP system, but small for a web search engine

A simple “definition” (Wikipedia)

- Consists of data sets that grow so *large* that they become awkward to work with using on-hand database management tools
  - Difficulties include capture, storage, search, sharing, analytics and visualizing

How big is big?

- Moving target: petabyte (10^{15} bytes), exabyte (10^{18}), zetabyte (10^{21}), …
- For example, climate modeling data are growing so fast that they will lead to collections of hundreds of exabytes expected by 2020

But size is only one dimension of the problem

4

Why Big Data Today?

Overwhelming amounts of data generated by all kinds of devices, networks and programs

- E.g. sensors, mobile devices, internet, social networks, computer simulations, satellites, radiotelescopes, LHC, etc.

Increasing storage capacity

- Storage capacity has doubled every 3 years since 1980 with prices steadily going down
- 295 exabytes: an estimation for the data stored by humankind in 2007 (probably more than 1 zetabyte today)
  - M. Hilbert, P. Lopez. The World's Technological Capacity to Store, Communicate, and Compute Information. Science. Online Feb. 2011

Very useful in a digital world!

- Massive data can produce high-value information and knowledge
- Critical for data analysis, decision support, forecasting, business intelligence, research, (data-intensive) science, etc.

5

Big Data Dimensions

Scale

- Refers to massive amounts of data
- Makes it hard to store and manage, but also to analyze (big analytics)

Velocity

- Continuous data streams are being captured (e.g. from sensors or mobile devices) and produced
- Makes it hard to perform online processing

Heterogeneity

- Each organization tends to produce and manage its own data, in specific formats, with its own processes
- Makes data integration very hard

Complexity

- Uncertain data (because of data capture), multiscale data (with lots of dimensions), graph-based data, etc.
- Makes it hard to analyze

6

AppEx1: Dark Energy Survey

Context: CNPq-INRIA project with LNCC, Rio de Janeiro

DES: International astronomic project to explain:

- Acceleration of the universe
- Nature of dark energy

Data production

- DECam (570-Megapixel digital camera) takes images of 1GB (more than 1000/night)
- Images are analyzed; galaxies and stars are identified and catalogued
- Catalogs are stored in a database

Problem

- Big data
  - The size of the database is expected to be ≥ 30 Petabytes
  - Tables with 130 attributes (more than 1000 in total)
- Complex queries
  - Each query may contain many attributes (>10)

7

AppEx2: Risers Fatigue Analysis (RFA)

Context: CNPq-INRIA project with UFRJ and LNCC, Rio de Janeiro (and Petrobras)

- Pumping oil from ultra-deepwater from thousand meters up to the surface through risers

Problem

- Maintaining and repairing risers under deep water is difficult, costly and critical for the environment (e.g. to prevent oil spill)
- RFA requires a complex workflow of data-intensive activities which may take a very long time to compute

8

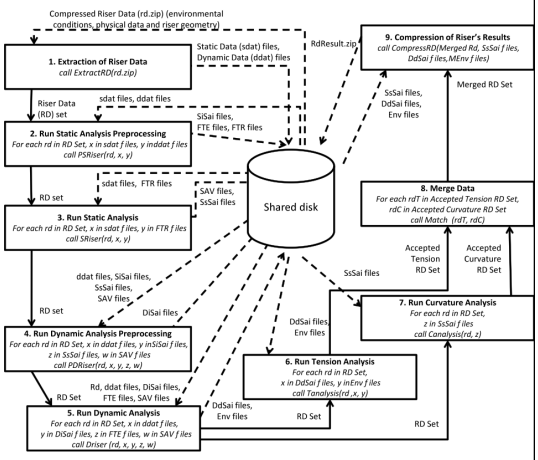
RFA Workflow Example

A typical RFA workflow

- Input: 1,000 files (about 600MB) containing riser information, such as finite element meshes, winds, waves and sea currents, and case studies
- Output: 6,000 files (about 37GB)

Some activities, e.g. dynamic analysis, are repeated for many different input files, and depending on the mesh refinements and other riser's information, each single execution may take hours to complete

The sequential execution of the workflow, on a SGI Altix ICE 8200 (64 CPUs Quad Core) cluster, may take as long as 37 hours



9

Scientific Data – *common features*

- **Massive scale, complexity and heterogeneity**
- Manipulated through complex, distributed *workflows*
- Important *metadata* about experiments and their provenance
- Mostly append-only (with rare updates)
  - Good news: no need for transactions
- Collaboration among scientists very important
  - e.g. biologists, soil scientists, and geologists working on an environmental project

10

Scientific Data – the Problem

Current solutions

- Typically file-based for complex apps
- Application-specific (ad hoc), using low-level code libraries
- Deployed in large-scale HPC environments
  - Cluster, grid, cloud

Problem

- Labor-intensive (development, maintenance)
- Cannot scale (hard to optimize disk access)
- Cannot keep pace (the data overload will just make it worse)

"*Scientists are spending most of their time manipulating, organizing, finding and moving data, instead of researching. And it's going to get worse*" (Office Science of Data Management challenge – DoE)

11

Why not Relational DBMS?

RDBMS all have a distributed and parallel version

- Scalability to VLDB
- With SQL support for all kinds of data (structured, XML, multimedia, streams, etc.)

But the "one size fits all" approach has reached the limits

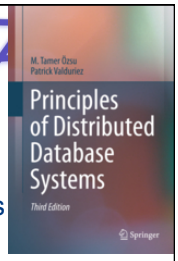
- Loss of performance, simplicity and flexibility for applications with specific, tight requirements

Not Only SQL (NOSQL) solutions (e.g. Google Bigtable, Amazon SimpleDB, MapReduce) trade data independence and consistency for scalability, simplicity and flexibility

- NB: SQL has nothing to do with the problem

12

Our Approach



Study end-to-end solutions
- From data acquisition and integration to final analysis

Capitalize on the principles of data management
- *Data independence* to hide implementation details
- Basis for high-level services that can scale
- Data partitioning: the basis for distributed and parallel processing

Foster declarative languages (algebra, calculus) to manipulate data and workflows, with user-defined functions

Exploit highly distributed environments
- Peer-to-Peer (P2P)
- Cloud

13

P2P Networks

Decentralized control
- No distinction between client and server
- Each peer can have the same functionality
- No need for centralized server

Massive scale
- Any computer connected on the internet can be a peer

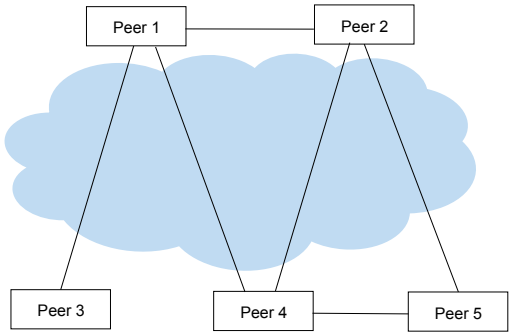
Autonomy and volatility
- Peers can join or leave the network at will, at any time

Many successful apps
- Content sharing (e.g. Bittorrent)
- Networking (e.g. Skype)
- Search (e.g. YaCy)
- Social networks (e.g. Diaspora)

14

P2P Architecture

- Highly distributed
- Dynamic self-organization
- Load balancing
- Parallel processing
- High availability through massive replication



15

P2P Benefits

Ease of installation and control

- No need for centralized admin.

Scalability to very high numbers of nodes

Cheap

- Leverages existing resources

Efficient

- With structured networks (DHTs)

Data privacy

- One may keep full control over her own data

16

P2P Drawbacks

Decentralized admininistration
- Hard to control

Security
- In the presence of untrusted nodes

Churn and low reliability of user machines

Latency in unstructured networks
- Because of flooding

17

Cloud Computing

The vision
- On demand, reliable services provided over the Internet (the "cloud") with easy access to virtually infinite computing, storage and networking resources

Simple and effective!
- Through simple Web interfaces, users can outsource complex tasks
  - Data mgt, system administration, application deployment
- The complexity of managing the infrastructure gets shifted from the users' organization to the cloud provider

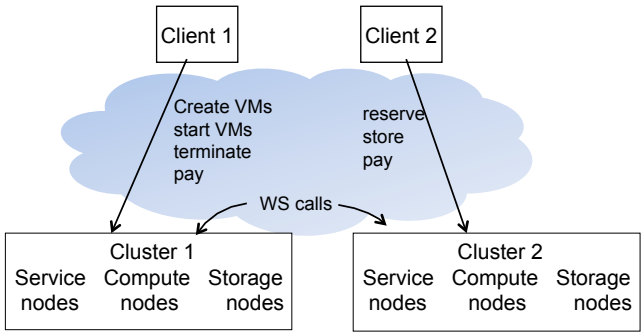
Capitalizes on previous computing models
- Web services, utility computing, cluster computing, virtualization, grid computing

Very successful business model
- Amazon, Microsoft, Google, IBM, etc.

18

Cloud Architecture

- Single point of access using Web services
  - Highly centralized
  - Big clusters
- Replication across sites for high availability
- Scalability
- Service level agreement (SLA), accounting and pricing essential



19

Cloud Benefits

Reduced cost
- Customer side: the IT infrastructure needs not be owned and managed, and billed only based on resource consumption
- Cloud provider side: by sharing costs for multiple customers, reduces its cost of ownership and operation to the minimum

Ease of access and use
- Customers can have access to IT services anytime, from anywhere with an Internet connection
- Centralized admin. simpler

Quality of Service (QoS)
- The operation of the IT infrastructure by a specialized, experienced provider (including with its own infrastructure) increases QoS

Elasticity
- Easy for customers to deal with sudden increases in loads by simply creating more virtual machines (VMs)

20

Cloud Drawbacks

Centralized architecture with single point of access
- Hard to move big data in and out
  - High bandwidth required
- Single point of failure

Loss of control
- Your data can get lost!
  - E.g. Amazon EC2 crash disaster in april 2012 permanently destroyed some customers' data
- Your data can get locked!
  - E.g. Megaupload shutdown by FBI in feb. 2012

Security and privacy
- Different privacy laws across countries
  - Make it hard to enforce with subcontracting of storage
- Business models of social networks like Facebook

21

Why P2P/cloud?

Can combine the best of both P2P and cloud for scientific data management

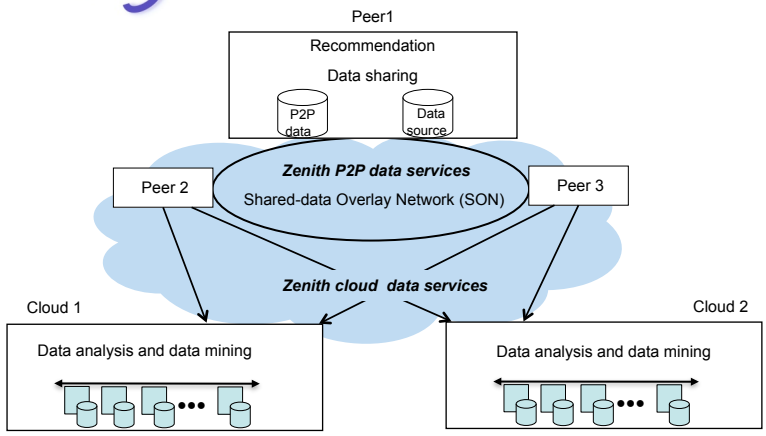
P2P
- Naturally supports the collaborative nature of scientific applications, with autonomy and decentralized control
- Peers can be the participants or organizations involved in collaboration and may share data and applications while keeping full control over some of their data

Cloud
- For very-large scale data analysis or very large workflow activities, cloud computing is appropriate as it can provide virtually infinite computing, storage and networking resources

P2P/cloud also enables the clean integration of the users' own computational resources with different clouds

22

P2P/cloud architecture



Combines the best of P2P and cloud

- P2P for data sharing between collaborative participants
- Cloud for elastic parallel data processing

23

AppEx1: Dark Energy Survey

Problem

- How to store big data (tables with 1000 of attributes) in order to process complex queries efficiently

Our approach

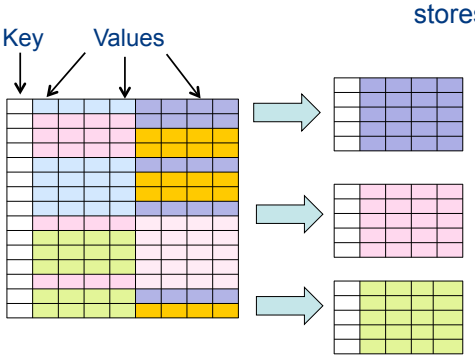
- Partition the datasets based on access patterns to co-locate data items accessed together, while minimizing inter-partition correlation

24

Data Partitioning

Vertical partitioning

- Basis for column stores (e.g. Vertica)



Horizontal partitioning

- Basis for key-value stores (e.g. Bigtable)

25

Dynamic Workload-Based Partitioning

Main idea: gradually adapt the partitioning to the new data (instead of repartitioning)

When a new data arrives, choose the best partition based on the affinity of data to partitions

- Affinity(d, p) : the number of queries that access both *d* and some data from partition *p*

Challenge: how to choose the best partition?

We developed an algorithm called DynPart

- That computes Affinity(d, p) based on the affinity of queries with partitions
  - Its complexity is O(|Q| * |P|)
- M. Liroz-Gistau, R. Akbarinia, E. Pacitti, F. Porto, P. Valduriez. Dynamic Workload-Based Partitioning for Large-Scale Databases. DEXA (2) 2012: 183-190.

26

AppEx2: RFA

Problem with data-centric scientific workflows

- Typically complex and manipulating many large datasets
- Computationally-intensive and data-intensive activities, thus requiring execution in large-scale parallel computers
- However, parallelization of scientific workflows remains low-level, ad-hoc and labor-intensive, which makes it hard to exploit optimization opportunities

Solution

- An algebraic approach (inspired by relational algebra) and a parallel execution model that enable automatic parallelization of scientific workflows
  - E. Ogasarawa, J. Dias, D. Oliveira, F. Porto, P. Valduriez, M. Mattoso. An Algebraic Approach for Data-Centric Scientific Workflows. VLDB 2011

27

Algebraic Approach

Activities consume and produce relations

- E.g. dynamic analysis consumes tuples, with input parameters and references to input files and produces tuples, with analysis results and references to output files

Operators that provide semantics to activities

- Operators that invoke user programs (map, splitmap, reduce, filter)
- Relational expressions: SRQuery, Join Query

Algebraic transformation rules for optimization and parallelization

An execution model for this algebra based on self-contained units of activity activation

- Inspired by tuple activations for hierarchical parallel systems [Bouganim, Florescu & Valduriez, VLDB 1996]

28

Challenges and Research Directions

Scalable query processing with big data

Scientific workflow management with big data

Data provenance management

Decentralized recommendation for big data sharing

Decentralized data mining with privacy preservation

29