Databases in the Cloud

Seminar: Big Data Analytics

Winter Semester 2012-13

Moshfiqur Rahman

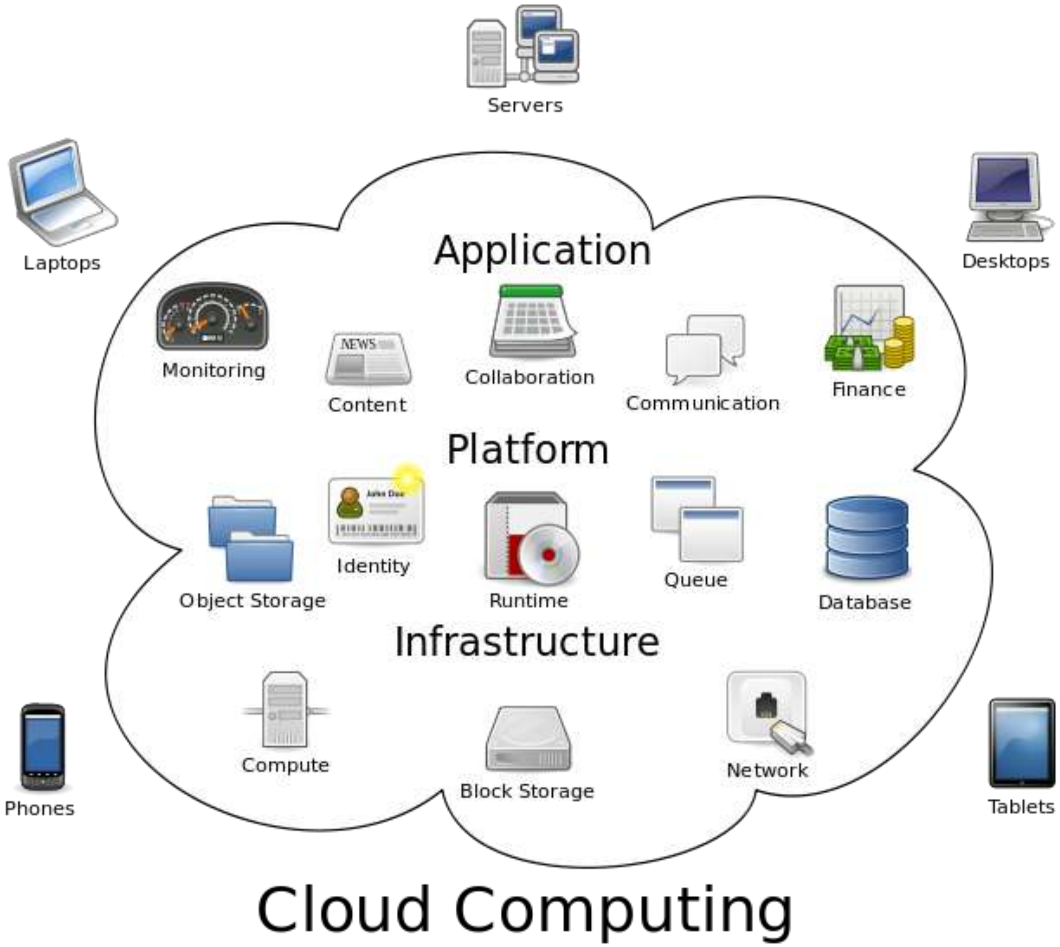
Agenda

- Cloud Computing Introduction
- Big Data
- RDBMS and Cloud Databases
- Scalability, Elasticity, Availability – New attributes for databases
  - In RDBMS
  - In Cloud Databases
- Challenges in Cloud Databases
- Big Data Analytics in Cloud
- Conclusion

Cloud Computing Introduction

- What is in the Cloud?
  - Application as a service
  - Hardware and system software
- Public and Private Cloud
- Cloud Computing attributes
  - Virtually infinite computing resources
  - Start small and grow as needed
  - Pay-per-use scheme

Cloud Computing Introduction



Source: http://en.wikipedia.org/wiki/Cloud_computing

Big Data

- Large and complex data sets
  - Exponential growth
  - Structured, semi-structured, unstructured
  - Hard to process in traditional database system
- Challenges with Big Data
  - Capture, scrutinization, storage, search, sharing, analysis…
- Big Data sources
  - Mobile devices
  - Sensors
  - Software/server logs
  - Cameras and so on…

Big Data Attributes

- Volume
  - Factors contribute to the increase of data, for example, text streams from social networks
  - Hidden relationships in data
  - Data storage cost decreased but data analysis issues increased!
- Variety
  - Data can be of all possible formats
  - Structured/semi-structured data from RDBMS
  - Unstructured data from documents, emails, video, audio, sensors
- Velocity
  - Keep up the data processing speed with data production speed
  - Streams of real time data from sensors and social media
  - Reacting quickly to the increase of data velocity

Relational Database

- A relational database is
  - Collection of tables (entities)
    - Multiple columns
    - Multiple rows (tuples)
- Accessed by SQL
- Join multiple tables to get related data
- Normalization is used to minimize redundancy and dependency
- Referential Integrity is used to ensure data consistency
- Managed by Relational Database Management System (RDBMS)
- Oracle, MS SQL Server, MySQL, etc.

RDBMS - A misfit for cloud?

- RDBMS has
  - Simplicity
  - Robustness
  - Flexibility
  - Performance
  - Compatibility
  - (Limited) Scalability
- Cloud databases require
  - Scalability
  - Elasticity
  - Availability

Cloud Databases

- Key/value store – a new kind of database management system
  - Store data as key/value pair
  - Targeted for specialized applications where a RDBMS is not suitable
- Also known as
  - Document-oriented database
  - Internet-facing database
  - Attribute-oriented database
  - Distributed database, etc.

Cloud Databases - Advantages

- Stores data in format of items
  - Customer items, Order items in an e-commerce system
  - A single item contains all the relevant data
- Relationships are not deprecated, just simplified
  - Order items contains the keys of associated Customer item and Product items
- Able to scale easily and dynamically
  - Allows the user to pay only for used resources
  - Allows the vendor to scale their infrastructure depending on their entire platform size

Cloud Databases - Advantages

- Reduce the development time
  - By decreasing developing time with object relational data mapping
  - Easier to map application object to key/value database items

Cloud Databases - Disadvantages

- Relationships are not defined in data models
  - DBMS cannot enforce data integrity
  - Deleting item from a set of related items will make data inconsistent
- No shared standard
  - Totally different set of APIs
  - Application developed for one cloud vendor is hard to port to another cloud vendor

Scalability

- Desired property of a system to accommodate growing amounts of work
  - By adding more hardware in single machine
  - By adding more machines (a.k.a. node)
- Two ways to scale a system
  - Vertically or Scale Up
    - New hardware is added to a single node in a system
    - Adding more processors or memory to a single machine
  - Horizontally or Scale Out
    - Add more nodes to a system
    - Scaling out from one web-server system to a three web-server system

Elasticity

- Ability to spread the workloads dynamically over the available resources
  - Automatically adds more resources when workload increases
  - Automatically shrinks back and removes the unneeded resources when workload decreases
- Very important for cloud environment
  - Pay-per-scheme

Availability

- Allows the user read and write data at any time without blocking them
- Response time is virtually constant and does not depend on
  - Number of concurrent users
  - Database size
  - Any other system parameter
- Automatic data backups and failover management

Scalability in RDBMS

- RDBMS provides limited scalability
  - Scale up on a single node
  - Scale out with relatively small numbers of nodes
- Scale up is not infinite but increase in workload can be virtually infinite
- Scale out is overwhelming in system with hundreds or thousands of nodes

Elasticity in RDBMS

- RDBMS allows very limited elasticity at storage and web/application server layers
  - Add a web server when the workload increases and adjust the throughput to dissipate the loads to the new server
  - When workload decreases, detach the server from the system, use it for different purposes
  - At storage layers, more disks can be added
- Adding a bigger machine and replace the overloaded database server
  - Expensive investment
  - Unnecessary investment for a seasonal hype

Availability in RDBMS

- Employs storage redundancy by performing data replication
  - Also ensure improved performance for concurrent users
  - Provides resiliency in case of a failure
- Data replication is not so easy process
  - Synchronization
  - Replicate the whole database to make synchronization easier

Scalability, Elasticity and Availability in Cloud Databases

- New breed of databases focusing on scalability, elasticity and availability
  - Key/value store supports nearly limitless scalability
  - In the expense of other benefits come with RDBMS
- Data accessed by a single key
  - Provides the basis for scalability
  - Data item is contained in a single object and handled by a single node
- Some modern applications need multiple key/value pair access atomically
  - Online multi-player games, Google Drive
  - Hence required multi-key atomicity

Scalability, Elasticity and Availability in Cloud Databases

- Different database implementations
  - Google's MegaStore
  - G-Store
  - Relational Cloud
  - ElasTras

MegaStore

- Uses Bigtable as the underlying system
- Provides multi-key atomicity
  - Data Fusion
  - Group multiple key/value pair as single collection
  - Write/ahead logging
  - Two-phase commit to support ACID transactions on a collection

MegaStore

- Advantages
  - Allows entities to be arbitrarily distributed over multiple nodes
  - Better performance when entity group co-located in a single node
- Disadvantages
  - Exhibits performance issues when entity group is distributed across multiple nodes

G-Store

- Provides transactional multi-key access over dynamic, non-overlapping groups of keys
- Created groups are transient in nature
- Creates abstract group for on-demand transaction access
  - Leader key, follower keys
  - Ownership of read/write access transfers to the node hosting the key group
  - No key should not be claimed by multiple group, no key should be without a owner

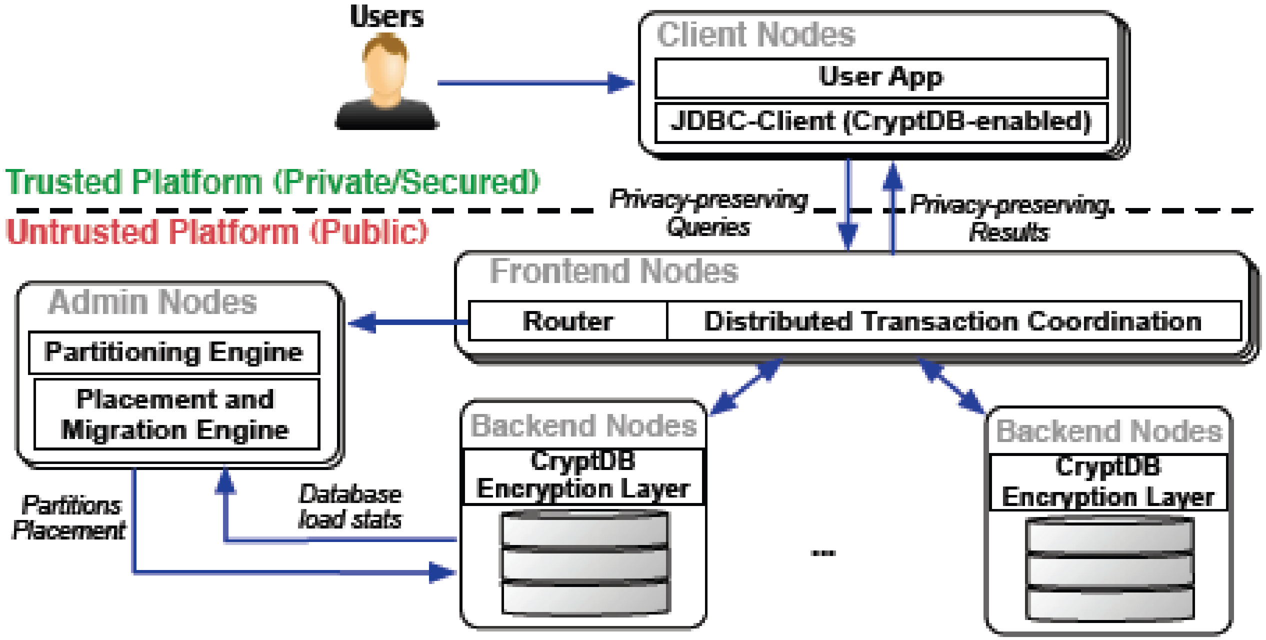
G-Store

- Advantages
  - Transactions are efficient for key group resides on single node
- Disadvantages
  - A group must be small enough to reside on a single node

Relational Cloud

- Works on Elasticity extensively
- Uses a graph-based partitioning method to split large databases across multiple machines
  - Workload aware partitioning strategy
- Frontend transaction trace component
  - keeps track of transactions
  - Analyze the transactions to determine the set of tuples accessed together
  - Creates a graph of transactions
  - Weight is given to the edges to denote how often a transactions are executed

Relational Cloud


Users
Client Nodes
User App
JDBC-Client (CryptDB-enabled)
Trusted Platform (Private/Secured)
Untrusted Platform (Public)
Privacy-preserving Queries
Privacy-preserving Results
Frontend Nodes
Router
Distributed Transaction Coordination
Admin Nodes
Partitioning Engine
Placement and Migration Engine
Backend Nodes
CryptDB Encryption Layer
...
Backend Nodes
CryptDB Encryption Layer
Partitions Placement
Database load stats

Relational Cloud

- Advantages
  - Uses the MySQL, Postgre-SQL as backend databases
  - Migrate the database partitions without causing downtime
  - Replicate the data for availability
- Disadvantages
  - Scaling the graph representation is difficult as it leads to a graph with N nodes and up to N^2 edges for an N-tuple database

ElasTras

- A cloud database under research providing better scalability and elasticity with transactional data access

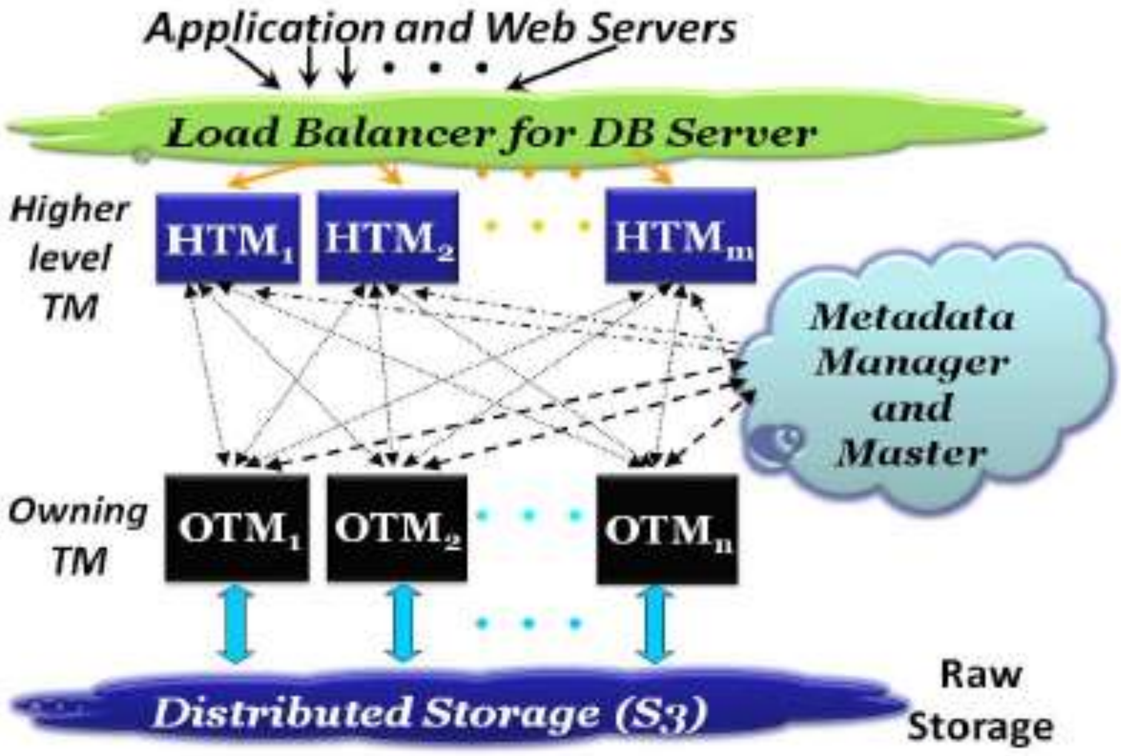


Figure: System overview of ElasTras

ElasTras

- Two level Transaction Manager (TM)
  - Higher level TM (HTM)
  - Owning TM (OTM)
- When any transaction request arrives
  - Load balancer uses some load balancing policy and forward the request to appropriate HTM
  - HTM decides whether to execute the transaction locally or forward to OTM
  - OTM has exclusive access rights to the data accessed by a single transaction
- System state information and database metadata managed by Metadata Manager

ElasTras

- Two approach to partition the database
  - Static Partitioning
  - Dynamic Partitioning
- Static Partitioning
  - Database designer defines the partitioning
  - ElasTras is responsible for mapping the partitions to their specific OTMs
  - Also reassigns partitions if workload increases
  - Application has the knowledge of partitions
  - ElasTras can provide ACID transactional guarantees as transactions executed locally to a partition

ElasTras

- Dynamic Partitioning
  - Basis for the elasticity of the data store
  - Uses range or hash based partitioning scheme
  - Applications are not aware of the partitions
  - Transactions are not guaranteed to be limited to a single partition
  - Provides mini transactions with restricted transactional semantics to ensure scalability and avoid distributed transactions
  - Mini transactions ensures recovery but no global synchronization

ElasTras

- Advantages
  - Provides transactional guarantees in scalable manner
  - OTM's reassigning partitions capability with changing workload ensures elasticity and scalability
  - Provides ACID transactions when transactions are limited to a single partition
- Disadvantages
  - In dynamic partitioning, ElasTras only supports mini transactions with restricted transactional semantics to avoid distributed transactions
  - Mini transactions only ensure recovery but no global synchronization

Challenges in Cloud Databases

- Importing data
  - Data transport are complex and may incur huge cost
- Auto failover management
  - Server crashes, hardware malfunction
  - Database must be replicated, automatically replace and start working if any failure occurs
- Auto scalability and elasticity management
  - Scale instantly and automatically both throughput and size
  - Very granular increases and shrinking back in resources

Big Data Analytics

- Big Data Analytics
  - Process of analyzing huge amount of structured, semi-structured and unstructured data of variety types
  - Discover the hidden patterns and unknown correlations in data
- Companies are interested in big data analytics to achieve competitive advantages over rival companies
  - Through effective marketing
  - Propose new innovative services

Big Data Analytics

- Big data analytics help companies make better business decisions
- Traditional analytic software are available for data analysis
  - Advanced technologies such as predictive analysis, data mining, etc.
- But, traditional analytics software
  - is not suitable for big data with semi-structured and/or unstructured data
  - is not able to handle the demand of processing power needs to analyze those big data
- New class of big data analytics environment has emerged
  - NoSQL databases
  - Hadoop
  - MapReduce

Big Data Analytics in Cloud

- Available database as a service in Cloud
  - Amazon SimpleDB
  - Google AppEngine
  - Microsoft SQL Azure
  - so on…
- Limitations in Cloud
  - Limitations over query execution time, for example, Amazon SimpleDB restricts any query which takes more than 5 sec
  - Limitations over result dataset size, for example, Google AppEngine does not allow users to retrieve more than 1000 items for any query
- Impractical for big data analytics

Big Data Analytics in Cloud

- Specialized solution for big data analytics in cloud
  - Google BigQuery
  - Amazon Elastic MapReduce (EMR)

Google BigQuery

- Cloud based interactive query service for big data
- Implementation of Dremel, a parallel query engine
- Query executes on a small number of very large append-only tables
- Two core technology
  - Columnar storage
    - Records are separated in column values
    - Put all single column values in different storage volume forming a tree
  - Tree architecture
    - Query pushing down to the branches of the tree
    - Results are aggregated from the leaves

Amazon Elastic MapReduce (EMR)

- A hosted Hadoop framework
- Provides a web service to process huge amounts of data
- Contains a MapReduce framework
  - Sub divides the data in smaller chunks and process them in parallel (the "map" function)
  - Recombines them into final solution (the "reduce" function)

Google BigQuery vs. Amazon EMR Head to Head

Google BigQuery	Amazon EMR
Interactive data analysis tool for large data set	A programming framework to process big data.
Comparable to Hive but claims to be faster than that	Accessible by data analysis application developed in Pig, Hive or other programming languages using Amazon's SDK
Designed to run faster query and user friendly even for non-programmers with built-in GUI	Supports implementing complex data processing logic
Good for ad-hoc and trial-and-error interactive query on large dataset for quick analysis and troubleshooting	Good for batch processing of large dataset doing time consuming data conversion and aggregation
Provides a regular expression engine to structure the unstructured data	Structuring data fully dependent on application logic
Does not support large result set neither joining of large tables	Supports both large result set and joining of table
Does not support updating existing data, only append of data is possible	Supports updating existing data

Conclusion

- End of RDBMS?
- Cloud databases for big data
  - Finding relationships in data
  - Solving the problem for scalability, elasticity and availability
- More rising issues
  - Efficient multi tenancy
  - Data privacy

Thanks for your attention