VICTORIA UNIVERSITY OF WELLINGTON
Te Whare Wananga o te Upoko o te Ika a Maui

Introduction to Cloud Databases

Lecturer* : *Dr. Pavle Mogin

SWEN 432
Advanced Database Design and Implementation

Plan for Intro to Cloud Databases

- An overview of Cloud Computing
  - What is Cloud Computing,
  - The Architecture of Cloud Computing
  - Cloud Computing Services
- Database as a Service (DaaS)
  - The basic features of DaaS
    - Distribution,
    - Replication
    - Scalability
- Inter node communication
  - Gossip protocol

    - ***Readings:*** Have a look *at Useful Links at the Course Home Page*

What is Cloud Computing (1)

- A major new paradigm rapidly shifting the way IT services and tools are used
- The main idea:

  Provide cheep infrastructure to IT online services
- How has it been achieved:
  - Using network connections to remote applications and services where users need to possess only a minimum of hardware and software
  - Paying for applications and services as per their usage, only
- Cloud Computing is based on a subscription model that is very similar to utility services like electricity, gas, or water

What is Cloud Computing (2)

- Users do not have to invest in any major hardware, or any major software
- They access them and use them on the cloud and pay only for the resources they use
- Users also do not need to be burdened by hardware and software installation and maintenance
- Applications and databases are stored in large server farms or data centers owned by Google, IBM, Microsoft, Amazon, …

Cloud Computing Services (1)

- Software as a Service (SaaS)
  - Software applications are provided to users by cloud providers where users do not have to install them at their sites
- Platform as a Service (PaaS)
  - A hardware or software platform (a computer, an operating system, a programming environment, run-time libraries,…) is provided to users
- Infrastructure as a Service (IaaS)
  - A service where users can use expensive hardware like array processor servers, and network processors
- Database as a Service
  - Cloud storage service where users hire storage facilities, including a DBMS and pay only for storage space they use

Cloud Computing Services (2)

- To attain scalable performance and robust availability of services, cloud computing vendors use:
  - Hardware, software, and data redundancy and
  - Complex techniques for managing networked platforms and data replication
- *Scalability* means that performance of a service does not depend on the number of users and clients requesting the service
- *Availability* of a service relates to the fact that the service is always ready for use
  - A service is highly available if it has a small latency

Cloud Databases

- Cloud databases have been developed to serve the massive growth of digital data and consumer services in the last 10 to 15 years:
  - Social networks (Facebook, Twitter),
  - Data storage and sync services (Dropbox, iCloud),
  - Desktop replacement (Google Documents)
- Cloud databases are distributed database systems that are accessed via cloud services
  - Database data are replicated and stored on multiple independent servers (nodes) to achieve scalability and availability
  - End users and client application use data via API's that hide details of the exact location where an data object might be stored

Common Features of CDBs (1)

- Most database services offer web-based consoles, which the end user can use to provision and configure database instances
- Database services consist of a database manager (DBMS) component, which controls the underlying database instances using a service API
  - The service API is exposed to the end users, and permits users to perform maintenance and scaling operations on their database instances
    - Creating a database instance,
    - Modifying resources available to the instance,
    - Deleting a database instance,
    - Creating a database backup,
    - Restoring a database from its backup

Common Features of CDBs (2)

- Database services make the underlying software stack transparent to the user
  - The stack includes an OS, DBMS and third-party software used by the database,
  - The service provider is responsible for installing, patching and updating the underlying software stack
- Database services take care of scalability and high availability of the database
  - Some vendors offer auto-scaling, others enable users to scale up using an API
  - There is typically a commitment for a certain level of high availability (e.g. 99.9% or 99.99% of the time).

Network Architectures of Cloud DBs

- Large cloud databases distributed over several node machines fall under one of the following two categories:
  - ***Shared nothing***, where each node contains a database partition and whole responsibility for data it holds,
  - ***Shared disk***, where all nodes have access to shared disks containing all database data and all nodes share responsibility for the sole copy of the database

Shared Nothing and Shared Disk

- In a shared nothing architecture:
  - A middleware is required to route incoming data requests to the node(s) responsible for data requested,
  - Reads and Writes involving data on a single node are efficient,
  - Reads and Writes involving data on multiple nodes are inefficient, since joins and constraints must be validated over multiple nodes,
  - Nodes can be added, or removed easily without affecting other nodes, making the architecture scalable
  - Very suitable for using commodity machines
- In a shared disk architecture:
  - All nodes have access to the entire database (no middleware required),
  - Allow for high database consistency needed by OLTP
  - Locking (*for consistency*) and logging (*for recovery*) introduce overheads, so hard to scale

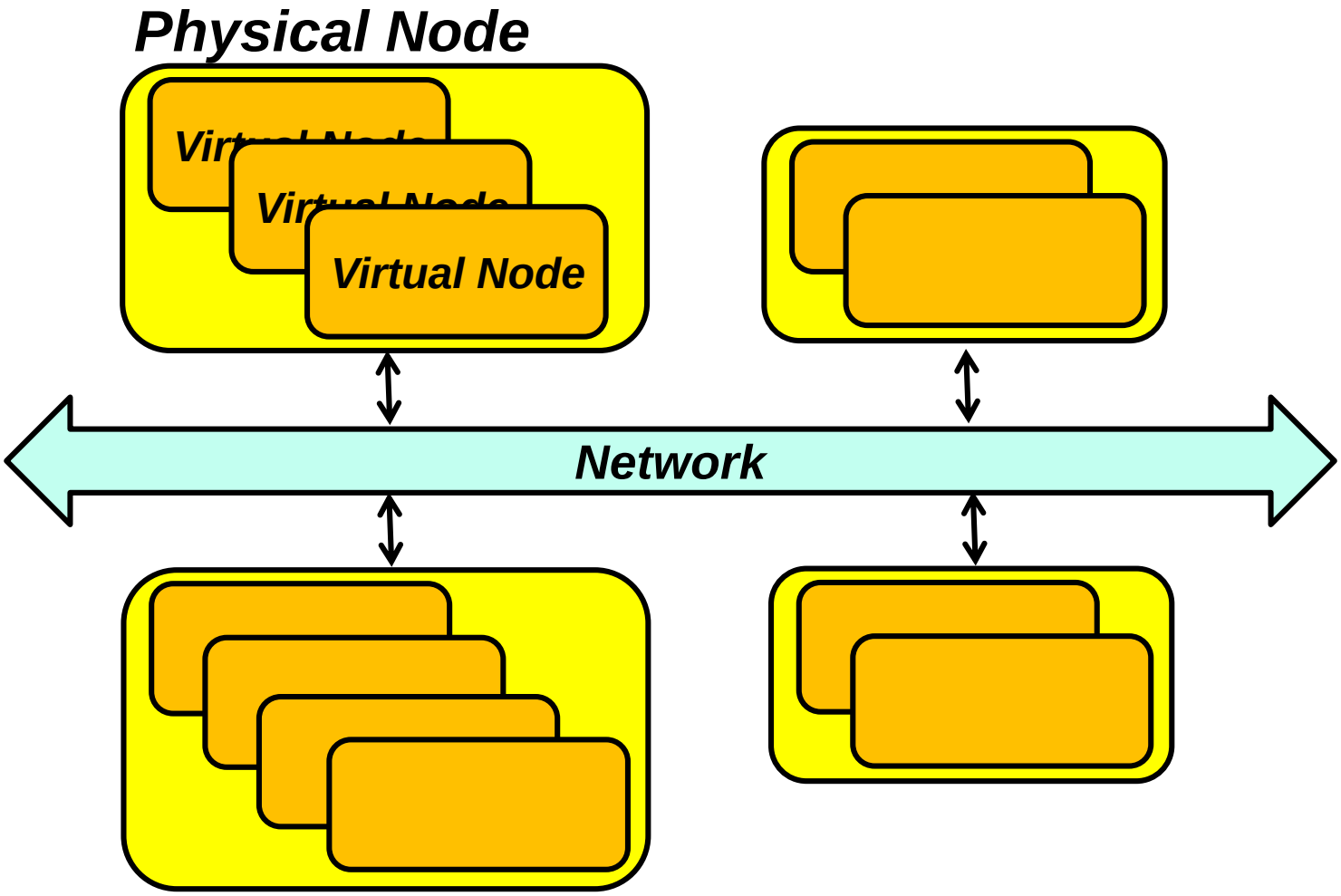
Data Replication and Distribution

- Practically all CDBMSs replicate data on several machines
  - Several copies of the same data are available to many users accessing the database simultaneously
  - Data availability has been enhanced in the case of: a great number of users accessing same data, or in the case of a server or network failure
- Machines, used to store replicas of a database may belong to several data centers
- A data center contains:
  - Servers, telecommunication infrastructure, buck-up and security facilities
  - Users are guided to their databases by cloud DBMS APIs and remain unaware of the exact location of their data

Machine Layout

- The underlying infrastructure is usually composed of a large number of networked commodity machines
- Each machine is called a ***physical node*** (PN)
- Each PN has the same software configuration but may have varying hardware performance
  - Processor speed,
  - Memory and disk capacity
- Depending on its performance, each PN contains a number of ***virtual nodes***

Physical and Virtual Nodes



Scalability (1)

- ***Scalability*** is the ability of a system:
  - To handle a growing amount of work in a capable manner or
  - The ability to be enlarged to accommodate the workload growth
    - This can refer to the capability of a system to increase total throughput under an increased load when resources (typically hardware) are added
  - A system whose performance improves after adding hardware proportionally to the capacity added, is said to be a **scalable system**
  - If the system fails when the workload or the quantity of hardware added increase, it **does not scale**
- One of the primary goals of cloud database systems is achieving ***cost effective*** scalability

Scalability (2)

- In principle, scaling of cloud databases can be achieved by:
  - Dedicating more nodes to the database system (referred to as horizontal scaling), and
  - Adding more resources (memory, CPU) to individual existing nodes
- Horizontal scaling is important to cloud databases because of its cost effectiveness
  - A horizontally scalable cloud database can be run on cheaper commodity hardware
  - As the number of users and data grow and more performance is required, more cheap nodes are added, and data and work load are distributed to the new nodes

Node Communication in a Network

- In a networked and replicated database system, where a client may issue a read or update request to any node, nodes have to communicate in order to:
  - Dispatch the client's request to a corresponding replica and
  - Propagate the client's updates to all replicas
- Very often, nodes communicate using a ***gossip protocol***
  - Gossip protocols are inspired by the form of gossip seen in social networks
- The term ***epidemic protocol*** is sometimes used as a synonym for a gossip protocol, because gossip spreads information in a manner similar to the spread of a virus in biological communities

Gossip Protocol

- The gossip protocol is an inter node communication protocol that satisfies the following conditions:
  1. The core of the protocol involves periodic, pair wise inter node interactions
  2. The information exchange during these interactions is of a limited size
  3. When nodes interact, the state of at least one of them changes to reflect the state of the other one
  4. Reliable communication is not assumed
  5. The frequency of the interaction is low compared to typical message latency, so that the cost of the protocol is negligible
  6. There is some form of randomness in the peer selection
     - Peers may be selected from the full set of nodes or from a smaller set of nodes (nodes hosting a replica of the same data)

Efficiency of a Gossip Protocol

- This is an approximate calculation
  - Assume, in the first round of gossiping, a node gets an information that it needs to share with others
  - So, the node picks another node and after the second round of gossiping two nodes know for the information
  - In the third round each node knowing the information shares it with two new nodes and that results in four nodes knowing the information
  - After the round *i* there are $2^{(i-1)}$ nodes knowing the information
  - Assume after the round *h* all *n* nodes in the system know the information

$$n = \sum_{i=1}^{h} 2^{i-1} = 2^{h}$$

  - The number of rounds to disseminate information to all *n* nodes is

$$h = \log_2 n$$

Gossip Protocol (Example)

- Assume:
  - A network contains $n = 25\,000$ nodes
  - A gossip occurs every *100* ms
- Then:
  - There are $h = 15$ rounds needed to spread information through the network and it takes *1.5* seconds

Summary (1)

- Cloud Computing:
  - Storing and accessing applications and computer data through a Web browser rather than running installed software on your computer
  - Internet-based computing whereby information, IT resources, and software applications are provided to computers and mobile devices on demand
- Cloud Computing Services leverage:
  - Commodity hardware,
  - Data redundancy, and
  - Robust availability

  over a collection of networked computers and reduce complexity of managing such systems by abstracting implementation details to a user

Summary (2)

- Cloud databases have been developed to serve the massive growth of digital data and consumer services in last 10 to 15 years
- The database service provider takes responsibility for installing and maintaining the database, and application owners pay according to their usage.
- Many cloud databases use the shared nothing architecture and replicate data on several network nodes
- Scalability is the ability of a system to increase total throughput under an increased load when resources (typically hardware) are added

Summary (3)

- Gossip inter node communication protocols are very popular among cloud database management systems
- They spread information and exchange states in a very reliable way, since there is no single point of failure
- There are $\log_2 n$ rounds of gossiping needed to spread an information through a network of n nodes