Quiz 3

Essential characteristics of cloud workload characterization

Workloads are grouped into four classes:

1. Rabbit - sensitive apps that perform better when not sharing cache
2. Turtle - Apps that do not really use their cache
3. Sheep - apps not perturbed by other apps
4. Devil - apps that do not like occupying cache and detrimentaly effect other apps

Workload categories can be split in two ways

1. Static architecture - implementation of solution architecture (i.e.parallel computational setup/big data storage)
2. Dynamic behavior - how resources are used in real time/stress that a workload places on the computational resources

Cloud Workload Categories

- **Slow communication** - minuscule amounts of info w/o a delivery time limit
- **Real-time local tasks** - hardware measurements given to a computer system
- **Location aware** - utilizing supplementary location input data to miniscule amounts of info w/o a delivery time limit
- **Real-time geographicaly dispersed task**s - multitude of scattered hardware measurements systems giving data to a network
- **Access control -** requests initiated by users where the response is to a different server with more authorized activity
- **Voice or Video over IP -** requests initiated by users where the response is through a server to each other
- **Big Streaming data workload category** - an interactive initiation followed by long periods of huge amounts of data sent to an end customer
- **Big Data storage workload category** - grand data (big datasets) which are periodically updated (requires a large download from time to time)
- **In-memory database workload category** - large # of data that is frequently & rapidly accessed
- **Many tiny tasks workload category** - independently running miniscule tasks
- **Tightly coupled intensive calculation workload category** - issues needing teraflops of computing power.
- **Separable calculation-intensive HPC workload -** enormous time-consuming # of calculations

- **Highly interactive multi-person jobs -** connectivity of jobs (i.e. discussion/ collaborating chats)
- **Single computer intensive jobs** - high speed substantial single-user taks that have lots of user interaction

- **Private local task** - traditional single user tasks

Computing Resources

- Persistent storage-
    1. user approximates a need
    2. gets a valid Service Level Agreement
    3. utilizes until Ithe resource requires an increase
- Compute power/computational capability
    - measured by:
        - CPU time/cycles
        - # & type of computer nodes available
        - # of cores available
        - the types and capabilities of CPUs assigned
- Network Bandwidth - examples that depend on network bandwidth
    - Xbox Live
    - Netflix
    - Playstation Network
- Broadcast transmission receivers -
    - need a special device added to the computer (i.e. GPS)
- Data busses within a server
    - CPU to memory
    - cache to main memory
    - memory to disk

Temporal Variability of Workloads

- two distinct cases in which the workload category would change.
    - When the job is incorrectly categorized
    - When the next step or phase of a job is a diff category than the current category

Essential characteristics of cloud management and monitoring

Cloud Management Terms

- Regions
    - Compromised of >= 2 availability zones
- Availability zones
    - A distinct location (inside a region) that won't be impacted by failures in other availability zones.
- Elastic load balancing
    - Automates process that distributes incoming app traffic across several EC2 instances

- Load balancers
    - Automatic scaling
    - Robust security
- Instance
    - Copy of an amazon machine image (AMI)
- Instance type
    - Specification that defines the cpu,memory,hourly cost, and storage capacity for a single instance
- Application performance

Cloud Management Requirements

- In-band
    - Agent that usually runs in a VM/OS
- Out of band
    - Monitoring tools that usually uses a baseboard management controller (has its own memory system + processor)
        - observing the main server's health metrics

Examples of Monitoring Tools

- Amazon Cloud Watch
    - a monitoring service for AWS resources and apps running on AWS
- New relic
    - App performance monitoring solution
        - uses agents placed in a VM - monitor how app is acting
- Nagios
    - a free & open source tool to monitor computer systems, networks, and infrastructure

Follow-ME cloud

- In reaction to the physical movement of a user's equipment : ability to seamlessly migrate a mobile user from 1 data center to another
    - NO disruption in the service
- Security Concerns
    - info leak issue during the transmission of the info/process after being stopped and before being freed up

Tradeoffs of freq vs infrequent monitoring of a server's performance

- frequent monitoring
    - can detect any variation of performance (even if it is a small one on CPU, network ,memory)
    - Frequent monitoring will however take more computational power
- infrequently monitoring
    - less computation intensive
    - User less likely to detect small variations in performance.

How to build a fail safe strategy :

- Always provide back up services
    - Back up data regularly
    - Host microservices independently
    - Improve intercloud compatibility
    - Eliminate single points of failure
    - Having hot swappable hard drives & backup power supplies

Metric to be monitored to ensure health of DC

- CPU, memory, and disk usage
    - Helps detect limited hard drive space, bandwidth , bottlenecks, high CPU utilization, and insufficient RAM
    - Efficient DC metrics : PUE (Power Usage Effectiveness), LEED (Leadership in Energy & Environmental Design) , PAR4, ASHRAE (The American Society of Heating, Refridgerating and Air Conditioning Engineers)

Key difference of peer to peer solutions

- difference between client server and peer to peer network:
    - Client - server : there is a dedicated server and specific clients in the client server network model
    - Peer to peer : in a peer to peer each node can act as both server and client vs in the client-server model , the server provides services to the client.

CAP Theorem

- CAP - consistency, availability, partition tolerance
    - Consistency - all clients simultaneously see same data.
    - Availability - system continues to operate (even when node failures are present)
    - Partition tolerance - system continues to operate (even when network failures are present)

Reliability and consequences of an outage

- cloud outage - a period of time in which a cloud providers services are unavailable to users.
- Reliability in cloud computing
    - Repetitive Redundant resources kick in automatically when the system experiences a issue/fault.

        - No downtime
- How to build a fail safe strategy
    - Always provide back up services
        - Back up data regularly
        - Host microservices independently
        - Improve intercloud compatibility
        - Eliminate single points of failure
        - Having hot swappable hard drives & backup power supplies

Miscellanious :

- Essential characteristics of cloud computing
    - On demand self service
    - Multi tenancy
    - Resource pooling
    - Broad network access
    - Rapid elasticity
    - measured service