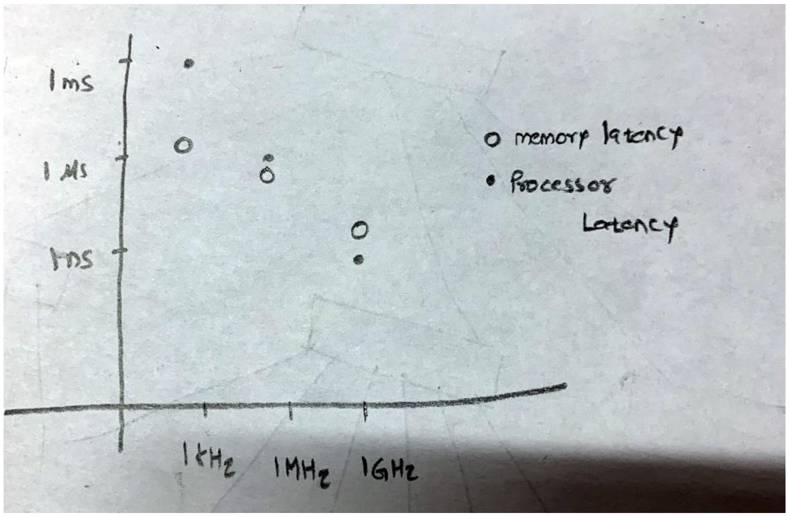
CS-553 Cloud Computing
Homework#05

Rohit Madhukar Lakde A20434998

1. Processors

a. Why did processors from the 1980s not need cache-coherent processors?

Ans: Below is the relationship between processor latency and memory latency. If we consider the era of 1980’s where memory was 1000 times faster than the processor and now today processor are faster than memory so to dominate that speed we use cache and cache coherence techniques to match with processor speed but in 1980’s processor speed was too slow which entirely dominated the speed of memory so there was no need of cache memory or cache coherence processors.



b. Assume that you have a x86 processor that has the following characteristics: 100 cores, 10 hardware threads per core, 1GHz speed for AVX640 instructions (vector of 640 bits) with 5 Fused Multiply-Add (FMA), 10GHz speed for all other instructions with 1 FMA and 2 instructions per cycle, 1MB L1 cache, 10MB L2 cache, and 100MB L3 cache, with a power envelope of 100 watts. What is the theoretical double precision (DP) TeraFlops per second?

Ans:

performance in GFlops = (CPU speed in GHz) x (number of CPU cores) x (CPU instruction per cycle) x (number of CPUs per node).

= 10 * 100 * 640/64 * 5 * 10^9 * 2

= 10^14 GFlops

= 100 TeraFlops/sec

2. Threading

a. Would it make sense to limit the number of threads in a server process?

Yes, it would make sense to limit number of threads in server process & it is mainly because of two reasons:

1. Threads have their own memory that is register, program counter and stack which is not shared with any other thread and processes. Increasing several threads unnecessary will take a too much of memory which might degrade the system performance.
2. If our task is small and we are sharing given task between number of threads, then majority of the time will be spent on spawning and terminating the threads.

b. Constructing a concurrent server by spawning a process has some advantages and disadvantages compared to multithreaded servers. Mention a few.

Ans:

Advantages:

Main advantage is that processes don't share memory between each other, and all are protected against each other which is very useful in handling independent tasks.

Disadvantages:

Disadvantage is that if task we are handling is dependent on some other task then internal communication is needed then it is easy to do it with threads instead of processes.

Creating new process is a very costly operation if we consider it with multithreaded servers.

3. Resilience

a. What components in a computer system do we know how to make resilient, and what technique is used? Name 3 such components.

Ans: Below are the components which can make system resilient

1. Disks: Adding more disks to the system can make system resilient to data recovery. RAID techniques are used for that. Data or resources can be replicated on different machines which will reduce the dependency of application and in case of failure another machine will lead the processing.
2. Processor: Extra processors can be added to system which remains idle while normal processing and in case of failure system activates that processor which makes it processor resilient.
3. Power: Power backup makes system power resilient, in case of power failure entire system can crash so maintaining extra power sources make system power resilient.

b. RAID (redundant array of inexpensive disks) is an example what type of recovery mechanism? Briefly describe 3 different levels of RAID.

RAID is a type of data recovery mechanism, In RAID data is backed up in another disk so in case of failure we can retrieve data from another disk and complete our operation.

RAID 0- It's a process of dividing data into blocks and spreading the data blocks over multiple disks. Strips are used for data reference in which data is divided into set of hard disks. It is used where speed and performance matters. One major drawback of RAID 0 level is it does not use data redundancy so failure in any drive results in data loss.

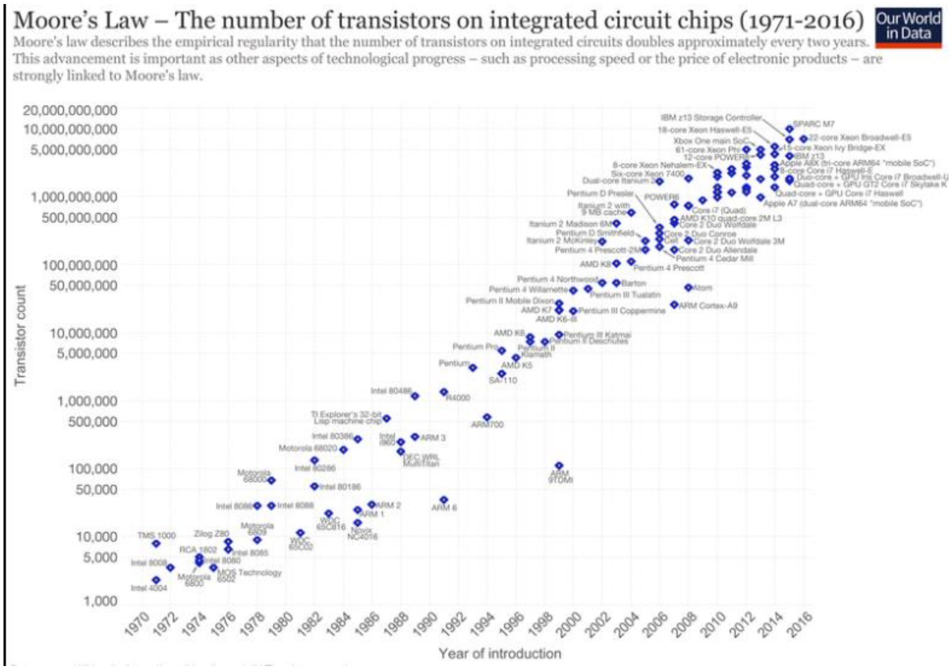
RAID 1- In RAID level 1 data is replicated on one or more disks, it is good for applications which requires high availability and high performance such as transactions and operating system. As data is present on multiple disks so data can be read from multiple disks simultaneously which makes read operation very fast. Write operations are slower as we need to write data in multiple disks at the same time.

RAID 4- It uses block level data striping and dedicated disks storing parity bits, which means it can read from any single disk which allows to use overlapped input output for read operations. While considering write operation we must update parity drive and no input output overlapping is possible for that. If any disk fails, then parity bit is used to create replacement disk. Disadvantages of RAID 4 is that it can create bottleneck to write operation.

4. Laws

a. Describe Moore's Law.

Moore’s law states that number of transistors in an integrated circuit doubles in every two years. It is a more of prediction given by Gordon Moore in 1965. As per the graph given below the prediction stated by Moore is well maintained over period of decade.



But today we have reached at the extreme end of Moore’s law and it is because of below reasons:

- **Electrical Leakage:** Transistors are getting smaller and more energy efficient, however they are getting smaller as small as 10nm and the channels which are carrying current cannot always contains it, which results in heat and that can damage transistors and its performance.
- **Increased cooling cost:** As the number of transistors increases heat generated by them also increases which results in increased cost of cooling, so to minimize the cost we need to minimize the number of transistors.

b. Describe Amdahl's Law.

Amdahl’s law states that if we are doing a task on a parallel platform then only part of the task can be optimized with a greater number of cores.

speedup = 1 / (s + p / N)

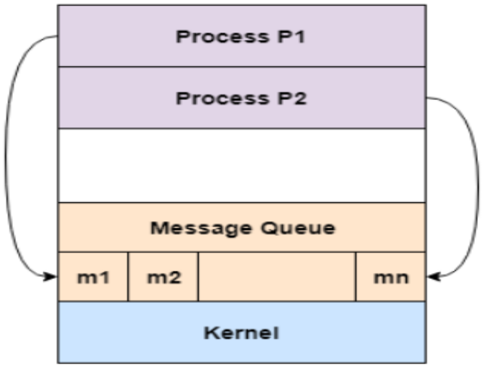
Consider a program that takes 20 hours to run using a single processor core. If a particular part of the program, which takes one hour to execute, cannot be parallelized (s = 1/20 = 0.05), and if the code that takes up the remaining 19 hours of execution time can be parallelized (p = 1 − s = 0.95), then regardless of how many processors are devoted to a parallelized execution of this program, the minimum execution time cannot be less than that critical one hour. Hence, the theoretical speedup is limited to at most 20 times (when N = ∞, speedup = 1/s = 20).

5. Programming models

a. Describe what shared address space and message passing is, and the difference between them? In what environments would one be used over the other?

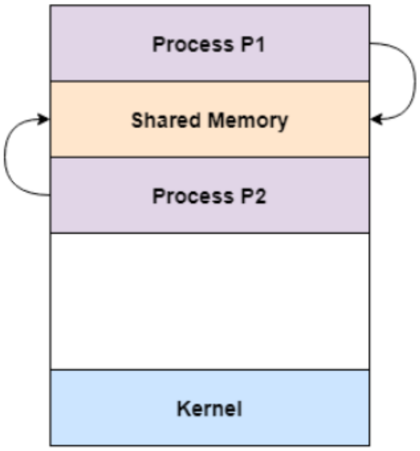
Ans: Shared address space and message passing are used for inter-process communication.

Message passing: Message passing is used by operating systems for inter-process communication. Message passing queues are used to store and pass messages and are stored until another processes retrieves it. As shown in below diagram both processes P1 and P2 can access the message queue and store and retrieve data. By using message passing it is easier to build parallel hardware.



Message Passing Model

Shared memory process: It is nothing, but a part of memory reserved for process communication and can be accessed by multiple processes. In the below diagram both processes P1 and P2 can access shared memory. Advantages of shared memory over message passing is, it is faster than message passing connection setup is not required. In some cases problems like mutual exclusion can occur where we need to protect part of memory to avoid deadlock.



Shared Memory Model

Difference between message passing and shared memory

Message Passing	Shared Memory
Memory communication is slower as compared to shared memory	Memory communication is faster as compared to message passing
Message queue is maintained for inter-process communication	Memory is reserved for inter process communication
Initial setup is required for message passing	Initial setup is not required for shared memory communication
Concurrency issues are not there	It faces lot of concurrency issue which might leads to deadlock
It is more time consuming than shared memory	Time required is less as compared with message passing

6. Distributed Systems

a. Name some advantages of distributed systems over centralized systems.

Ans: Advantages are as follows:

Economics: After some threshold distributed systems are cheaper as compared with centralized systems.
Speed: Speed and computing power of distributed systems is more as compared to centralized systems.
Inherent distribution: No need to put all hardware at the same place as centralized systems
Reliability: In distributed systems even if one system crashes operations wont fail.
Incremental growth: Computing power can be added in small increments.
Extensibility: After some limit, it is not possible to extend centralized system, but it is possible with distributed systems.

b. Define a cluster of computers. Define a supercomputer. What is the difference between clusters and supercomputers?

Ans: Cluster of computers: Computer cluster is a group of computers that are connected to perform given work, it is also known as number of computers that are connected by LAN.

Supercomputer: Computer with high level performance which can be measured in floating point operations per second is known as supercomputer.

Computer Cluster	Supercomputer
In cluster each machine is independent on other machine in term of memory and disk	Only one machine is present with thousands of processors
Systems communicate with each other by normal networking	Protocols like message passing protocols are used to communicate
Bunch of machines	Single machine
Building computer cluster is easier	After some extent building supercomputer is difficult or impossible
Cost of implementing computer cluster having same speed as that of supercomputer is cheaper	Cost of implementing supercomputer is very costly as compared to computer clusters with same specifications

c. Define grid computing. Define cloud computing. What is the difference between grids and clouds?

Grid computing: It is a process architecture that combines computer resources from various domains to reach main objective. It is designed to solve problems which are very big for supercomputers.

Cloud Computing: On demand availability of computer system resources like computing power, memory and storage.

Grid Computing	**Cloud Computing**
It operates within a corporate network	It can be accessed anywhere from the internet
It is for application oriented	It is for service oriented
Resources are distributed among different computing units to perform a given task	Computer resources are managed centrally and placed over multiple servers
It is a collection of interconnected computers	More than one computer coordinates to resolve the problem
It provides shared pool of computing resources	It involves dealing with common problems using varying number of computer resources

7. Energy

a. Why power consumption is critical to datacenter operations?

There are mainly two reasons for that which are as follows:

1. ***To reduce cost:*** 20% of the total cost required to build on data center is spent on power and cooling infrastructure. It's a very big amount so it is very necessary to reduce overall power consumption and cost of cooling at minimal level.
2. ***Prevent hardware components from damage:*** Even if we are not considering the cost, suppose we are using a lot of power to run our datacenter. We know power is directly proportional to

work done and work done is proportional to heat produced. As we raise our power supply heat generated in system increases which results in system damage.

b. What is dynamic voltage frequency scaling (DVFS) technique?

Ans: It's a power saving technique in computer architecture by adjusting the value of frequency or voltage of CPU or attached peripherals. It consists of two techniques Dynamic voltage scaling and Dynamic frequency scaling.

1. ***Dynamic voltage scaling:*** In this case voltage is adjusted to reduce the power consumption, there are two types of DVS knows as 'undervolting' and 'overvolting'. Increasing in voltage is known as overvolting and decrease in voltage is known as undervolting. Undervolting is to conserve the power which is mainly used in devices having batteries like laptops, mobiles. Overvolting is for increasing computer performance.
2. ***Dynamic frequency scaling:*** It's a technique where frequency of the computer is adjusted to reduce power consumption, it is also known as CPU Throttling. It helps to preserve the batteries which reduces the cooling cost. Software's put overclocked frequencies into dynamic frequency scaling algorithms to avoid chip degradation.

c. If you were to build a large $1B data center, which would require $50M/year in power costs to run the data center and $50M/year in power costs to cool the data center with traditional A/C and fans. Name 2 things that the data center designer could do to significantly reduce the cost of cooling the data center? Is there any way to reduce the cost of cooling to virtually $0? Explain why or why not?

Ans: There are two techniques which we can use to reduce the cost of cooling of data center.

1. Immersion cooling: It is a technique in which hardware components are directly immersed into non-conducting liquid. It reduces the overall size of datacenter by 10 times, talking about power consumption cost it saves around 95% of the cost, so it's very efficient technique to save power. In above case we can save **$47.5M** per year.
2. By building datacenter at the top of hill or nearby river so that we can pass cold water through tubes in hardware will also save cost of cooling, this technique is known as liquid cooling.

8. Benchmarking

a. Throughput can be used to measure processors, memory, disk, and networks. What are the basic units of measurement for each of these?

Ans: Below are the units to measure throughput of processors, memory, disk and networks

Processor: GigaOPS (Giga operations per second)

Below is the formula to calculate processor throughput

FLOPS = sockets * (cores/socket) * (cycles/second) * (Flops/cycle)

Memory: Bytes per second

Below is the formula to calculate memory throughput

Memory throughput = Base DRAM clock frequency * Number of data transfers per clock

* Memory bus (interface) width * Number of interfaces

Disk: IOPS (Input Output operations per second)

Below is the formula to calculate Disk throughput

Disk throughput = Throughput/IO (all value in kb)

Networks: bps (bits per second)

Below is the formula to calculate network throughput

Network throughput = Number of bits/second

9. Networking

a. Name two network technologies you would use in building a large-scale computing system? One network should be used to optimize cost, while the other should be used to optimize performance.

Ans: Below are network technologies we can use in building large scale computing system

1. Load balancing: Load balancing distributes network or application traffic across a bunch of servers. It increases responsiveness and availability of applications. It is placed between client and server accepting incoming network and application traffic and distributing it across multiple backend servers. By balancing application requests across multiple servers, a load balancer reduces individual server load and prevents any one application server from becoming a single point of failure, thus improving overall application availability and responsiveness.
2. Fault tolerance and caching is used in distributed systems
3. In terms of cost optimization, WAN optimization techniques can be used. They involve traffic shaping, bandwidth management, data compression and caching. This improves performance, high data availability and increased efficiency. It drastically reduces the cost of equipment and the need for additional bandwidth.

b. Assume you have a cluster with 30 nodes. You have 1 network card per node with 1Gb/sec Ethernet Full Duplex and have access to 6-port switches (also 1Gb/sec Ethernet Full Duplex) in order to build a Fat Tree network architecture. Draw a picture of the Fat Tree topology for your 30-node cluster (clearly show the switches, cables, and nodes).

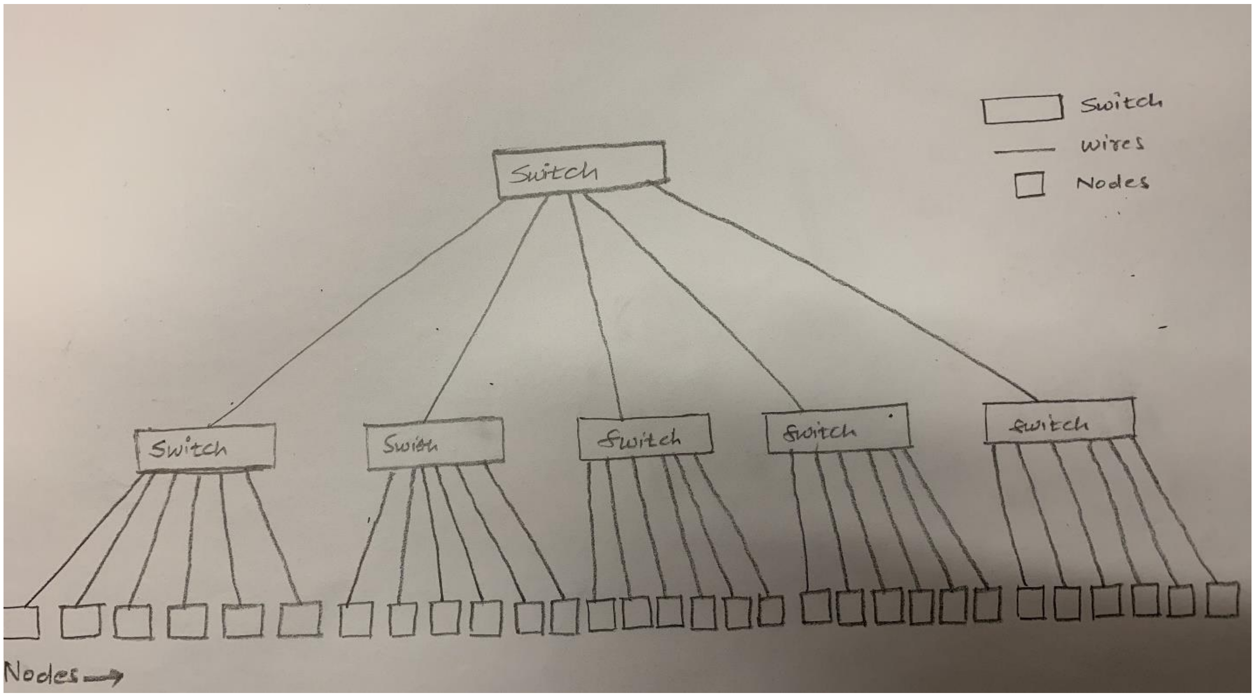
(a) How many switches in total do you need?

(b) What is the bi-section bandwidth of your network?

(c) What is the bi-section bandwidth of your network in Gb/sec (round to nearest Gb/sec)?

(d) Assuming each switch incurs a 100-microsecond forwarding delay, and networking stack requires 40microseconds to process network messages (e.g. TCP/IP) on each side (e.g. sender and receiver), what is the best-case and worst-case latency you can expect from this network topology?

Ans:



(a) How many switches in total do you need? → 6 switches

(b) What is the bi-section bandwidth of your network? → 0.33Gb/sec

(c) What is the bi-section bandwidth of your network in Gb/sec (round to nearest Gb/sec)? -->0.33Gb/sec

(d) Assuming each switch incurs a 100-microsecond forwarding delay, and networking stack requires 40microseconds to process network messages (e.g. TCP/IP) on each side (e.g. sender and receiver), what is the best-case and worst-case latency you can expect from this network topology?

Best case latency: 40 + 100 + 40 = 180 micro-seconds for the machines under same switch.

Worst case latency: 100 +100+100+40 +40 = 380 micro-seconds communication between machines present at extreme end switches.

10. Scalability

a. What does it mean for a system to be scalable?

Ans: The system which is scalable with number of components is known as scalable system. System is known as scalable only when adding or removing the components doesn't make the user sense a difference. The entire system feels like one coherent, logical system. Let's understand it by taking example suppose we have a system (comprising of many servers and probably many networks) and if gives response to the requests immaterial of the traffic coming in, which means working of the system is same even if 1k, 1M, 1B users are requesting the things. There are two types of scaling experiments vertical and horizontal scaling. Vertical scaling is when system includes more resources which are handled by making changes to the programming code or algorithms and horizontal scaling means adding more devices to the existing system.

b. Describe the difference between strong scaling and weak scaling experiments.

Ans: Difference between strong and weak scaling is as follows:

Strong Scaling	Weak Scaling
It spreads the same problem size across more nodes	It keeps the problem per size constant
Problem size stays fixed while the number of processors increased	Problem size stays assigned to each processor is fixed
Compute a fixed-size problem N times faster	Compute a problem N times bigger in the same amount of time
It doesn’t assume anything about communication between processors	It assumes amount of communication between processors remains constant
Speedup S = T1 / TN	Speedup depends on the amount of serial work remaining constant