Solutions to Homework Problems in Chapter 1

Hwang, Fox and Dongarra: Distrubuted and Cloud Computing,
Morgan Kaufmann Publishers, copyrighted 2012

Note by Hwang: The solutions of Chapter 1 problems were partially contributed by Siddharth Razdan, Lizhong Chen and VarunPalivela, who took my EE 657 class at Univ. of Southern California .

Problem 1.1:

(a) High-performance computing (HPC) system

HPC system emphasizes the raw speed performance, and is often measured in term of FLOPS as the floating-point computing capability.

(b) High-throughput computing (HTC) system

HTC concerns more about high throughput. This means that it is interested in how many tasks can be completed per unit time instead of how fast an individual task can complete.

(c) Peer-to-peer (P2P) networks

P2P network is a distributed application architecture that partitions tasks among peers. In this system, every node acts as both client and server, and the system is self-organizing with distributed control. The peer IDs form an overlay network at the logical level, which can be either structured or unstructured.

(d) Computer Clusters vs. Computational Grids

Computer clusters are built by a collection of interconnected stand-alone computers, which work cooperatively together as a single integrated computing resource. Computing grids on the other hand offers an infrastructure that couples computers, software, and sensors etc. together, and is often constructed across certain networks. One big difference between clusters and grids is that grids tend to be more loosely coupled, heterogeneous, and geographically dispersed.

(e) Service oriented architecture (SOA)

SOA originated from the idea of building systems in terms of services. A system that is based on SOA groups functionality as a suite of interoperable services that can be used within multiple separate systems. Some typical examples include Jini, CORBA and REST.

(f) Pervasive computing vs. Internet computing

Pervasive computing means the growing trend towards embedding microprocessors in everyday objects so they can communicate information everywhere. It is also sometimes called ubiquitous computing. Internet computing on the other hand is able to distribute the computing task among different computing facilities across the internet.

(g) Virtual machines vs. Virtual Infrastructure

Virtual machine is an abstraction provided by a middleware layer that exports an illusion of identical resources (CPU, memory, disk, etc.) as the physical machine. Virtual infrastructure is

what connects resources (such as compute, storage and networking) to the distributed application. It is a dynamic mapping of the system resources to specific applications.

(h) Public clouds vs. Private Clouds

In public cloud, service provider makes resources, such as applications and storage, available to the general public over the Internet, while private cloud is a proprietary network or data center that uses cloud computing technologies, and is managed by the organization it serves.

(i) Radio frequency identifier (RFID)

RFID uses radio waves to identify and track an object by attaching an electronic tag to it and communicating with the reader. It can be used in the internet of things.

(j) Global positioning system (GPS)

GPS is a global positioning system that provides reliable location and time information of the GPS-equipped objects through the communication with satellites.

(k) Sensor networks

Sensor Network consists of distributed autonomous sensors that monitor physical or environmental conditions, such as temperature, sound, vibration, pressure and to cooperatively pass their data through the network to a main location for further processing.

(l) Internet of things (IOT)

Internet of things is a wireless network of sensors to interconnect all things in our daily life. It can connect any thins at any time and place with low cost. It is getting popular as the advent of IPv6 protocol.

(m) Cyber-physical systems (CPS)

CPS merges computation, communication and control into an intelligent closed feedback system between the physical world and information world. It explores the virtual reality applications in the physical world.

Problem 1.2:

(1). c. (2). c

Problem 1.3:

(a). Main characteristics of cloud computing systems

Dynamics, efficiency, scalability, service oriented, high availability and reliability, low cost

(b). Key enabling technologies in cloud computing systems

Fast platform deployment, virtual clusters on demand, multi-tenant techniques, massive data

processing, web-scale communication, distributed storage, licensing and billing services.

(c). Different ways for cloud service providers to maximize their revenue

Cloud services are used in a pay-as-you-go manner. Cloud can support large number of users at the same time; one physical machine can support several virtual machines. All this makes the resources utilization increasing and because of the high utilization, service provider can make money. Moreover, Service providers enjoy greatly simplified software installation and maintenance and centralized control over versioning. In addition, cloud service providers can build the cloud platform at the places where the electricity, network resources are cheaper, and cooling is easier.

Problem 1.4:

Globus (d); BitTorrent (f); MapReduce (c); EC2(i); TeraGrid(h);

EGEE (g); Hadoop(a); SETI@home(j); Napster(b);Bigtable(e)

Problem 1.5

(a) The completion times for core A, B, C and D are 32/1, 128/2, 64/3, 32/1 respectively. Therefore, the total execution time is 128/2=64 time units

(b) The processor utilization rate is:

(32+64+64/3+32) / (64x4) = 149.33/256 = 58.3%

Problem 1.6

(a) According to Amdahl's law:

S = (0.25+0.75/4) / (0.25+0.75/256) = 1.73

(b) In this case, we have:

S = (0.25+0.75/256) / (0.25/2+0.75/256) = 1.9771

Problem 1.7 :

a) Speedup, $S = 1/[\alpha + (1 - \alpha) / n]$
Since, $\alpha = 0$ and $n = 64$, **Speedup, S = 64**

b) Efficiency, $E = S / n$ **= 1 = 100%**

c) Speedup, S' = Time taken by a single server / Time taken by the cluster
$= cN'^3 / [(cN'^3 / n) + (dN'^2 / n^{0.5})]$

$c = 0.8$, $d = 0.1$, $N' = n^{1/3}N$, $N = 15{,}000$, $n = 64$

$S' = 0.8 \times 64 \times N^3 / [0.2N^2(4N + 1)] = 256N / (4N + 1)$

Since, $4N >> 1$, $S' = 256N / 4N$ **= 64**

d) Efficiency, $E' = \alpha / n + (1 - \alpha)$, Since$\alpha = 0$, **E' = 1 = 100%**

e)

- For both cases i.e. fixed workload and scaled workload the speedup and efficiency is the same i.e. S = 64 and E = 100%
- The bottleneck for speedup even after using a large number of parallel cores is the part of sequential code α. So even if we have a large cluster we will not have maximum speedup due to α. Also the efficiency is very low when we use a fixed workload.
- We use scaled workload to improve the efficiency. However it is still not 100% due to α.
- In the above 2 cases since $\alpha = 0$, we are able to get the maximum speedup of 64 and efficiency of 100% for both fixed and scalable workloads.

Problem 1.8 :

(a) Hardware, software and networking support

As cloud platforms are built on top of datacenters, the hardware resources of cluster/grid could be part of a larger computing cloud. Cloud typically needs some kind of web service or software like a Web browser. Also, components of a cloud may be more sparsely distributed, which needs a wider network support.

(b) Resource allocation and provisioning methods

Cloud resources are dynamically provisioned by datacenters upon user demand while clusters are not. Additionally, cloud system provides computing power, storage space and flexible platforms for upgraded web-scale application services.

(c) Infrastructure management and protection

Cloud computing relies heavily on the virtualization of all sorts of resources, and it needs stronger protection than cluster/grid.

(d) Supporting of utility computing services

Cloud computing inherently supports utility computing services, just like electricity, while cluster/grid are usually deployed by individual enterprises and do not provide to public as utility.

(e) Operational and cost models applied

Cloud is more cost-effective than cluster/grid. Besides the advantages brought by virtualization, the customers of cloud do not have to buy infrastructure before using it. Also, QoS and SLA add extra requirement on the operational and cost models.

In the future, the two computing paradigms can be converged. For example, there could be no so-called cluster. Everything is based on a comprehensive cloud. In this comprehensive cloud, a customer could request compute, storage and network resources as much as a traditional cluster. Thus, clusters are included in the cloud.

Problem 1.9:

(a). The general computing trend has been to leverage more and more shared resources over the internet. On the HPC front we have seen that out of the desire to share computing resources,

the co-operative clusters have replaced homogenous supercomputers. On the PC side there is an interest of moving the desktop computing to service oriented computing using server clusters and huge databases and datacenters over the public internet through powerful network interfaces. So we see here that rather than revolutionary HPC and PC have taken an evolutionary path over the last 30 years.

(b). Disruptive and sudden changes in the processor architecture are not favorable because if the processor architecture changes drastically it will require the instruction set architecture to change and the software and hardware built for the previously used processor architecture may not be compatible with the new architecture so drastic changes in processor architecture hamper backward compatibility.

Memory Wall: The issue of memory wall has risen because the rate of improvement of the microprocessor speed exceeds that of the rate of the growth of improvement in DRAM memory speed. An important reason for this disparity is the limited communication bandwidth beyond chip boundaries. Studies show that from 1986 to 2000, CPU speed improved at an annual rate of 55% while memory speed only improved at 10.
The memory wall is stopping factor from achieving scalable performance because the advantages of higher clock speeds are negated by memory latency, since memory access times have not been able to keep pace with increasing clock frequencies.

Moreover, power consumption, cooling and packaging also become an issue as power consumption increases linearly with respect to clock frequency. The clock rate has increased from 12 MHz to 4 GHz. Another problem is the issue of memory wall. The growing disparity between CPU and memory is mainly due to the limited communication bandwidth between the CPU and memory and is becoming the main bottleneck in development of large scalable systems. Increasing memory latency with the growing disparity between CPU and memory is the major cause of concern in achieving scalable performance.

(c). X86 is the name of a processor instruction set, or collection of operations that a processor is able to perform. These instructions include mathematics and logic calculations, among other types of tasks. Nearly every processor in use today maintains compatibility with the x86 instruction set, which is now more than 30 years old.X86 is the name of a processor instruction set, or collection of operations that a processor is able to perform. These instructions include mathematics and logic calculations, among other types of tasks. Nearly every processor in use today maintains compatibility with the x86 instruction set, which is now more than 30 years old.

Over the years, many processor manufacturers have entered the x 86 markets to compete head on with Intel. Replication of the x86 instruction set was accomplished through reverse engineering, a process by which a chip's capabilities are reproduced by engineers who have no experience with the chip itself, and thus cannot steal its technology. The best known x 86 processor manufacturers aside from Intel is AMD, which competes with Intel in the server, desktop and notebook processor markets.

In summary, x-86 processors are still dominating the PC and HPC markets because of the

following advantages :

- Backward compatibility with previous architectures and hence even older applications made for older architectures are compatible.
- The x-86 processors are cheap and readily available.
- The x-86 architecture can be virtualized.
- Latest x-86 processors offer 8 cores.
- Processor speeds go up to 4 GHz.

Problem 1.10:

(a) Both multi-core and GPU are parallel architectures. However, GPUs usually have a highly parallel structure that makes them more effective than general-purpose CPUs for a range of complex algorithms used for video processing. Also, due to the parallel nature of graph, GPU has much high usage than multi-core CPU.

(b) Processor technology follows Moore's law and advances rapidly. On the other hand, not all programs can be parallelized. Moreover, most programs have both a serial and a parallel part, which determines the maximum degree of parallelism. Lastly, there could be extra communication overhead among the parallel components of a program. Therefore, parallel programming cannot match the progress of processor technology.

(c) One way is to exploit thread-level parallelism. With core scaling, we would have hundreds or thousands of cores on a chip, which could support a large number of concurrent threads. By recognizing possible data or other structural parallelism, we could improve the effectiveness of parallel programming. Also, if we cannot increase the core usage, we may use DVFS to save its energy.

(d) SSD uses solid-state memory to store persistent data, while traditional hard disk drive uses electromechanical devices that contains spinning disks and movable read/write heads. Therefore, SSD which uses microchips to retain data in non-volatile memory chips and contain no moving parts would be much faster. HPC and HTC systems that use SSD would deliver better speedups.

(e) For current technology, InfiniBand and Ethernet can both achieve a bandwidth of 10 Gbps, and are both cost-effective. These two attributes satisfy the current requirements of HPC system. In the near future, InfiniBand and Ethernet will support even higher speed (Ethernet already has IEEE standard for 40/100 Gbps). This would be sufficient for the interconnection in HPC for at least several years. Therefore, both of them will continue to dominating the HPC market.

Problem 1.11:

Single – threaded superscalar

a. Architecture characteristics
   - Instruction level parallelism
   - Single core but multiple functional units
   - Instructions are issued from a sequential instruction stream

- Data dependencies resolved by hardware

b. Advantages
   - Instructions per cycle, IPC > 1

c. Shortcomings
   - Limited amount of instruction level parallelism
   - Complexity of the dispatcher and the dependency checker

d. Examples: P5 Pentium, AMD K5

Fine - Grain Multithreading

a. Architecture characteristics
   - Instruction level parallelism.
   - However instructions are issued from different threads after every clock cycle.
   - Single core, single functional unit.
   - Instructions per cycle, IPC = 1(maximum).
   - Presence of enough registers so that its contents need not be saved and restored during every context switch every cycle.

b. Advantages
   - For single pipelined cores, this is the only way to improve the IPC.
   - Data dependencies are less since the consecutive instructions are from different threads. Hence, delays due to long latency events such as page faults are low.
   - Context switching overhead is low compared to coarse grained multithreading.

c. Shortcomings
   - Since there is context switching after every cycle, a single thread may take a long time to complete.
   - Not all workloads contain sufficient parallelism to supply an active switch to thread on every cycle.
   - High hardware requirements.

d. Examples: SunUltraSPARC T1, Cray MTA

Coarse - grain Multithreading

a. Architecture characteristics
   - Instruction level parallelism
   - Context switching occurs only when large latency events are encountered such as page faults.
   - State must be saved and restored on every context switch.
   - Single core, single functional unit.
   - Instructions per cycle, IPC = 1(maximum).

b. Advantages

- Lower hardware requirements compared to fine grain multithreading.
- Single threads run until completion unless a large latency event occurs.

c. Shortcomings
- High overhead for context switching.
- Wastage of clock cycles whenever a high latency event occurs because the remaining instructions already in the pipeline need to be flushed before new instructions from another thread can be issued.

d. Examples: Intel's Montecito, IBM RS64-II

e.

Simultaneous Multithreading (SMT)

a. Architecture characteristics
- Instructions are issued from multiple threads in a single clock cycle.
- Must be superscalar in order to implement.
- Both thread level parallelism and instruction level parallelism.

b. Advantages
- Instructions per cycle, IPC > 1 and is higher than single threaded superscalar architecture.

c. Shortcomings
- Burden on the application developers as they have to check if SMT will increase or decrease the performance.
- Disadvantage if you want 100% CPU performance to solve a single problem.

d. Examples: Intel Atom, IBM Power7

Multicore Chip Multiprocessor(CMP)

a. Architecture characteristics
- Multiple cores
- Instructions per cycle, IPC > 1
- Uses both thread level parallelism and instruction level parallelism
- Cores in the CMP may implement superscalar or multithreading architectures.

b. Advantages
- Performance is much greater than the earlier architectures.
- Multicore requires less energy than a single core for the same performance.

c. Shortcomings
- Programmers have to write code that can be parallelized.
- Integration of multi core chips is harder than single core chips.

d. Examples :AMD FX-series, Nvidia GeForce 9

Summary in a Table

Processor Micro-architectures	Architecture Characteristics	Advantages/ Shortcomings	Representative Processors
Single-threaded Superscalar	Only instructions from the same thread are executed	Low processor utilization; cannot tolerate long-latency event	Pentium P5 AMD K5
Fine-grain multithreading	Switches the execution of instructions from different threads per cycle	Take advantage of small latency / Need hardware support of many threads	Sun Niagrara T1 Denelcor's HEP
Coarse-grain multithreading	Executes instructions from same thread for severalbefore switching	tolerate L1 miss / Large cost due to flush pipeline stages when switching	IBM iSERIESSStar; Intel Montecito
Simultaneous Multithreading	Simultaneously scheduling instructions from different threads at the same cycle	Take advantage of small latency in OoO cores / Complex hardware	IBM Power 7 Intel Atom
Multicore Chip Multiprocessor	Different cores execute instructions from different threads completely	Energy-efficient, and is able to explore thread-level parallelism	IBM Power7 Niagara 3

Problem 1.12:

(a) This is because virtual machines and virtual clusters offer novel solutions to several serious issues in cloud computing system, such as underutilized resources, application inflexibility, software manageability and security concerns. Also, with resources virtualized, it would be more cost-effective.

(b) These areas include the virtualization of low-cost machines to provider high overall performance, virtualization techniques to better deal with data deluge, and virtualization that could provide better security.

(c) Cloud platforms can have a significant impact on how HPC and HTC designs, creates, and delivers applications and systems to the customers. For example, with SLA, HPC and HTC systems would face bigger challenges to design the system with minimal delay and maximum throughput. Larger storage space and better reliability are also desired.

Problem 1.13:

- ***Infrastructure as a Service (IaaS):*** This model put together infrastructures demanded by users, namely servers, storage, networks, and datacenter fabric. The user can deploy and run on multiple VMs running guest OSes on specific applications. The user does not manage or control the underlying cloud infrastructure, but can specify when to request and release the needed resources. The best examples are AWS, GoGrid, and Rackspce.
- ***Platform as a Service (PaaS):***This model provides the user to deploy user-built applications onto a virtualized cloud platform. PaaS include middleware, database, development tools,

and some runtime supports like Web 2.0 and Java, etc. The platform include both hardware and software integrated with specific programming interfaces. The provider supplies the API and software tools (e.g., Java, python, Web2.0, .Net). The user is freed from managing the cloud infrastructure. The best examples are Google AppEngine, Windows Azure, and IBM BlueCloud.

- ***Software as a Service (SaaS):*** This refers to browser-initiated application software over thousands of paid cloud customers. The SaaS model applies to business processes, industry applications, CRM (*consumer relationship management)*, ERP (*enterprise resources planning*), HR (*human resources*) and collaborative applications. On the customer side, there is no upfront investment in servers or software licensing. On the provider side, costs are rather low, compared with conventional hosting of user applications. The best examples are the Salesforce.com, Google Docs, and Microsoft Dynamix CRM services.

Problem 1.14:

a) Application cloud services

- Also known as Software as a Service(SaaS).
- Delivery of software from the cloud to the desktop.
- This is browser initiated application software.
- Eliminates the need to install and run the application from the users computer resulting in simpler maintenance and support.
- It applies to business processes, industry applications, consumer relationship management, enterprise resources planning, human resources and collaborative applications.
- On the consumer side there is no upfront investment in servers or software licensing.
- On the provider side costs are low compared to traditional hosting of user applications.
- Examples :GoogleDocs and salesforce.com

b) Platform cloud services

- Also known as Platform as a Service(PaaS).
- Delivers a computing platform as a service.
- It facilitates deployment of applications without the cost and complexity of buying and managing the underlying hardware and software layers.
- Users can write applications that can run on the cloud, using the virtualized platform and services that they provide.
- The provider supplies the API and software tools(eg., Java, Python, Web 2.0)
- Examples : Google App Engine, Microsoft Azure

c) Compute and Storage services

- Also known as Infrastructure as a Service(IaaS).
- Rather than buying servers and storage devices users can buy those resources as a fully outsourced service.
- The user does not manage the cloud infrastructure but can specify which resources he wants to access or release.
- Provides a data center and a way to host client Virtual Machines and data.

- Examples : Amazon Web Services, IBM Reservoir

d) Collocation cloud services

- Also known as Location as a Service(LaaS).
- Provides a collocation service to house, power and secure all the physical hardware and network resources.
- It requires multiple cloud providers to work together to support supple chains in manufacturing.
- Examples :Savvis, Internap

e) Network cloud services

- Also known as Network as a Service(NaaS).
- Includes Virtual LAN's for interconnecting all the hardware components.
- Provides network services such as firewall.
- Network resources required to support new services such as virtual private cloud.
- Examples : AT&T, Qwest

Problem 1.15:

(a) Denial of Service(DoS):

It is an attempt to make a resource usually an Internet Site from working properly or even stop it from working completely for a varied amount of time. DoS attacks are carried out mainly on major web servers like financial institutions and also root nameservers. DoS attacks are usually done by using excessive number of external communication requests to saturate the target machine which makes the target machine unavailable or too slow to respond to legitimate traffic.

DoS attacks can compromise the entire network rather than the host they are carried out on by consuming the bandwidth.The symptoms of DoS attack are Unavailability of website or inability to access it, slow network performance, Increase in number of spam messages.

(b) Trojan Horse:

Trojan horse is non self replicating malware that appears to perform a desirable function for the user but instead facilitates unauthorized access to the user's computer system. If a Trojan horse is installed on a system it gives the hacker the remote access to that computer and then depending on the type of the Trojan horse and the security and user privileges on the user computer hacker can perform various operations like Data theft, Modifying or Deleting files, logging Keystrokes, installing software, downloading and uploading files, using the host machine for spamming or DoS attacksAnti Virus Software are used to detect and remove Trojan horse.

(c) Network Worms:

Network worm is a self-replicating Malware computer program. It uses a computer network to send copies of itself to other computers on the network. Worms unlike Viruses do not need to attach themselves to an existing program and hamper the network performance by

hogging the bandwidth and can lead to attacks on a host computer like deletion of files, creation of a backdoor to allow the worm author to control that computer.

(d) Masquerade:

In computer networks Masquerade attacks are where the attacker pretends to be an authorized user of a system in order to gain access to it or to gain greater privileges than it is authorized for. Masquerade attacks can be done by using stolen IDs, passwords weak authentication, gaps in security programs or bypassing the authentication mechanism. Depending on the privilege level that the attacker gets on entry he will able to modify and delete software and data, and make changes to network configuration and routing information.

(e) Eavesdropping:

Network Eavesdropping is a network attack consisting of capturing packets from the network transmitted by others' computers and reading the data content in search of sensitive information like passwords, session tokens, or any kind of confidential information.

The attack could be done using tools called network sniffers. These tools collect packets on the network and, depending on the quality of the tool, analyze the collected data like protocol decoders or stream reassembling.

(f) Service Spoofing:

In Service Spoofing a hacker first finds an IP address of a trusted host and then modifies the packet headers so that it appears that the packets are coming from that trusted host.

The hacker then sends messages to the target computer indicating that the message is coming from a trusted host in order to gain unauthorized access to computers.

(g) Authorization:

Authorization is the mechanism by which a system determines what level of access a particular authenticated user should have to secure resources controlled by the system. Authorization systems depend on secure authentication systems to ensure that users are who they claim to be and thus prevent unauthorized users from gaining access to secured resources.

(h) Authentication:

Authentication in a transaction refers to letting one party prove the identity of the other party. The party whose identity needs to be proved is called the claimant and the party that tries to prove the identity of the claimant is the verifier. Authentication can be achieved by using some secure public keys.

(i) Data Integrity:

The data available is said to be corrupt if it has some errors in it which make it different from the data we were expecting it to be. Data integrity means that the data is accurate and complete and has not been corrupted in any way. Data is usually corrupted by fault in sensors,

transmission errors over internet and transcription errors. Some methods to try and achieve data integrity are by using cryptographic hash functions.

(j) Confidentiality:

Confidentiality means ensuring that information is accessible only to those authorized to have access. Confidentiality is breached in many ways like lack of proper security, theft or loss of confidential data holding memory disk or card, passing confidential data over the phone.

Problem 1.16:

a) Power consumption in data - center operations

- According to an IDC report, in 2009, about 7% of data center expenses are for power distribution, lighting and transformer costs. This is about $12Billion for 55 million servers. Therefore to run a data center a company has to spend a huge amount of money for energy per year.
- Earth simulator requires $1200 an hour whereas Petaflop requires $10,000 an hour just for energy. These high costs are unacceptable to many system operators.
- It is estimated that on an average 15% of the servers in a company are idle i.e. they are powered on without any use. This means that around 8.25 million servers are not doing any useful work.
- By the earlier estimate of $12Billion spending worldwide in energy, turning off the idle servers results in a saving of $1.8Billion !
- Consumption of so much energy has adverse environmental affects too. The amount of wasted energy is equal to about 11.8 million tons of carbon dioxide per year.
- We need to apply techniques such as Dynamic power management(DPM) and dynamic voltage – frequency scaling(DVFS) in order to utilize power more efficiently.

b) Dynamic voltage frequency scaling technique (DVFS)

- It is based on the fact that energy consumed in a CMOS circuit is directly proportional to the frequency and the square of the voltage supply as follows:

$$E = C_{eff} f v^2 t, \qquad f = K(v - v_t)^2 / v$$

  where v is the voltage, K is a technology dependent factor, C_{eff} is the circuit switching capacity, $v_t$is the threshold voltage and t is the execution time of the task under clock frequency f.

- Power consumption is controllable by switching between different frequencies and voltages.
- The idea is to reduce voltage and/or frequency during the work – load slack time. The transition latencies between the low power modes are very small. Thus energy is saved by switching between operational modes.
- DVFS can reduce only the active component of the system power dissipation under the condition that the active power is much larger than the standby component. If the inverse is

true then reducing the voltage or frequency using DVFS will only increase the power consumption.

- Used in Intel's CPU throttling technology, SpeedStep, which is used in its mobile CPU line.
- However the downside is that it may reduce performance.

c) Green IT research and its application to datacenter design and cloud services

- Green IT is comprised of initiatives and strategies that reduce the environmental footprint of technology. This arises from reductions in energy use and consumables, including hardware, electricity, fuel and paper – among others. Because of these reductions, Green IT initiatives also produce cost savings in energy use, purchases, management and support, in addition to environmental benefits.
- For example, server virtualization. This allows users to reduce the capital cost of future server purchases and the operational costs of energy, maintenance and management.
- Electricity footprints and the amount of equipment needed are reduced, and often, the business realizes incentives for saving energy from the government.
- Decrease the overall number of devices running in the server room, along with the square footage needed to house these devices.
- Decrease the energy required to run servers and storage, along with the associated cost and greenhouse gas emissions.
- Decrease the cost of future investments in physical servers and storage devices. By operating server room assets at higher utilization rates, many IT shops will require fewer purchases in the future.
- Moving desktops to a virtual environment and employing thin-client machines reduces energy consumption and environmental impact of user infrastructure.
- Using software that centrally manages energy settings of PCs and monitors. Enforcing standardized power settings on all PCs before distributing to end-users. Procuring energy-efficient equipment, such as Energy Star certified devices.
- Server room upgrades and new server room builds in order to decrease cost and increase effectiveness of cooling and ventilation systems.
- Reducing travel costs by implementing remote conferencing and telecommuting strategies such as video conferencing, teleconferencing, virtual private network(VPN) and remote access.
- Implementing IT equipment recycling. This reduces the waste sent to landfills.

Green IT research on Datacenter Design and Cloud Service Applications

In the report to congress on "Server and Data Center Energy Efficiency" issued by U.S. Environmental Protection Agency ENERGY STAR Program in 2007, the energy used by the nation's servers and data centers is significant. It is estimated that this sector consumed about 61 billion kilowatt-hours (kWh) in 2006 (1.5 percent of total U.S. electricity consumption) for a total electricity cost of about $4.5 billion. This leads to the notion of Green IT.

San Murugesan defines the field of green IT as "the study and practice of designing, manufacturing, using, and disposing of computers, servers, and associated subsystems—such

as monitors, printers, storage devices, and networking and communications systems—efficiently and effectively with minimal or no impact on the environment." According to Wikipedia, the goals of green computing are similar to green chemistry, which include reducing the use of hazardous materials, maximizing energy efficiency during the product's lifetime, and promoting the recyclability or biodegradability of defunct products and factory waste.

Green IT has been paid great attentions by both government and industry, as it is an environmentally sustainable information technology.

Approaches towards Green IT

Many endeavors have been made into this field of research to build a green IT as much as possible. Below are some of the focus areas with this issue:

- Design for environmental sustainability;
- Energy-efficient computing;
- Power management;
- Data center design, layout, and location;
- Server virtualization;
- Responsible disposal and recycling;
- Regulatory compliance;
- Green metrics, assessment tools, and methodology;
- Environment-related risk mitigation;
- Use of renewable energy sources; and
- Eco-labeling of it products.

As a brief report, we summarize below some of the popular and recent research efforts to achieve green IT.

- *Enabling power management features:* the software places the PC into a lower-power consumption mode, such as shutdown, hibernation, or standby, and monitors into a sleep mode when they aren't being used.
- *Resource allocation:* algorithms can be used to route data to data centers where electricity is less expensive.
- *Virtualization:* it enables datacenters to consolidate their physical server infrastructure by hosting multiple virtual servers on a smaller number of more powerful servers, using less electricity and simplifying the data center.
- *Greening unwanted computers:* resue, refurbish and recycle the old computers, so call as Three R's.
- *Improving the efficiency of power supplies:* PSUs are generally 70–75% efficient, dissipating the remaining energy as heat. We could improve by adopting new Energy Star 4.0-certified desktop PSUs, which could be at least 80% efficient.
- *Reducing power consumption of storage:* smaller form factor hard disk drives often consume less power per gigabyte than physically larger drives, and research also shows that solid-state drives, which store data in flash memory with no moving parts, consumes less power.

In addition to above, there are also many research efforts in seeking the opportunity to reduce energy and power consumption by providing better operating system support, using terminal servers, implementing telecommuting and so on.

Impact on Datacenter Design and Cloud Service Applications

There are many applications of green IT to datacenter design and cloud service applications. Below are some of them:

- Cloud computing should continue using virtualization heavily for all sorts of resources, and more cost-effective virtualization techniques should be devised and used.
- We could build datacenters at some locations where electricity cost is lower to save electricity bill, or where the environment temperature is cooler to reduce cooling electricity.
- A comprehensive research and development program should be initiated to develop technologies and practices for data center energy efficiency, which may cover many topics including: computing software, IT hardware, power conversion, heat removal, controls and management, and cross-cutting activities.
- Examples of the R&D in the IT hardware of the above category could be: improve server platform power management capabilities; develop lower power states for use at lower utilization levels; improve power management for storage systems, to allow many disks to remain off most of the time with little impact on performance; investigate the reliability impacts of thermal cycles in a power-managed server; and investigate application of solid-state (non-mechanical) storage technologies to data centers, as mentioned the report to congress.

Problem 1.17:

A graphics processing unit (GPU) is well recognized as a graphics coprocessor or accelerator mounted on the graphics card or video card of a computer. A GPU offloads the CPU from tedious graphics tasks in video editing in a computer. Historically, the world's first *graphics processing unit* (GPU), GeForce 256, was marketed byNvidia in 1999. These GPU chips can process a minimum of 10 million polygons per second. They are used in nearly every computer. Some GPU features were also integrated into certain CPUs. The traditional CPUs are structured with only a few cores. For example, the Xeon X5670 CPU has 6 cores. However, a modern GPU chip could be built with hundreds of processing cores.

Unlike CPUs, GPUs have a throughput architecture that exploits massive parallelism by executing many concurrent threads slowly, instead of executing a single long thread in conventional microprocessor very quickly. Laterly, parallel GPUs or GPU clusters are catching a lot of attentions against the use of CPU with limited parallelism. *General-purpose computing on GPU, known as* GPGPU, has appeared as dubbed GPU computing in the HPC field. Nividia's CUDA model was used in HPC using GPGPUs.

Thousand-core GPUs may appear in Exa-scale (EFlops or 10^{18} Flops) systems in a few years. This reflects a trend to build future MPPs with a hybrid architecture of both types of processing chips. In a DARPA report in September 2008, four challenges are identified for Exascale computing: (1) energy and power, (2) memory and storage, (3) concurrency and locality and (4)

system resiliency. Here, we see the progress of GPUs along with CPU advances in power efficiency, performance and programmability.

Problem 1.18:

Feature Comparison of Three Distributed Operating Systems (DOS)

DOS Functionality	AMOEBA developed at VrijeUniversity	DCE as OSF/1 by Open Software Foundation	MOSIX for Linux Clusters at Hebrew U.
History and Current System Status	Written in C and tested in European Community, version 5.2 released in 1995	DEC was built as user extension on top of the UNIX, VMS, Windows, OS/2, etc.	Developed since 1977, now MOSIX2 used in HPC Linux clusters and GPU clusters
Distributed OS Architecture	Microkernel-based, location transparent, using many servers to handle files, directory, replication, run, boot, and TCP/IP services	Middleware-OS providing a platform for running distributed applications The system supports RPC, security, and Threads.	A distributed OS with resource discovery, process migration, run-time support, load balancing, and flood control, configuration, etc.
OS Kernel, Middleware and Virtualization Support	A special microkernel handles low-level process , memory, I/O, and communication functions	DCE packages handle file, time, directory, and security services, RPC, authentication at middlware or user space	MOSIX2 runs with Linux 2.6, extensions recent for use in multi-clusters and clouds with provisioned VMs
Communication Mechanisms	Use a network-layer FLIP protocol and RPC to implement point-to-point and group communications	DCE RPC supports authenticated communication and other security services in user programs	Using PVM, MPI in collective communicatiuons, priority process control, and queuing services

The DEC is a middleware-based system for distributed computing environment. Open Software Foundation (OSF) has pushed the use of DEC for distributed computing. However, all three systems are still research prototypes mainly for academic use. The conventional OS runs only on a centralized platform. With the distribution of OS services, the DOS design should take either a light-weight microkernel approach like the Amoeba or extend UNIX. The trend is to free up users from most resource management duties

MOSIX2 is a distributed OS, which runs with a virtualization layer in Linux environment. This layer provides partial *single-system image* to user applications. Both sequential and parallel applications are supported by MOSIX2, which discovers resources and migrate software processes among Linux nodes. MOSIX2 can manage a LINUX cluster or a grid of multiple clusters. Flexible management of a grid allows owners of clusters to share ther computational resources among multiple cluster owners.