VIRTUALIZATION & CLOUD COMPUTING

Lecture

CSE 423

- *Introduction to Cloud Computing*

What is Cluster Computing?

- A cluster is a type of parallel or distributed computer system, which consists of a collection of inter-connected stand-alone computers working together as a single integrated computing resource .
- Key components of a cluster include multiple standalone computers (PCs, Workstations, or SMPs), operating systems, high-performance interconnects, middleware, parallel programming environments, and applications.

“Utility” Computing ?

- Utility Computing is purely a concept which cloud computing practically implements.
- Utility computing is a service provisioning model in which a service provider makes computing resources and infrastructure management available to the customer as needed, and charges them for specific usage rather than a flat rate.
- This model has the advantage of a low or no initial cost to acquire computer resources; instead, computational resources are essentially rented.
- The word *utility* is used to make an analogy to other services, such as electrical power, that seek to meet fluctuating customer needs, and charge for the resources based on usage rather than on a flat-rate basis. This approach, sometimes known as *pay-per-use*

Cloud Computing in a nutshell

- Analogy to electricity use
- Technologies such as cluster, grid, and now cloud computing, have all aimed at allowing access to large amounts of computing power in a fully virtualized manner, by aggregating resources and offering a single system view
- Utility computing describes a business model for on-demand delivery of computing power; consumers pay providers based on usage.
- It denotes a model on which a computing infrastructure is viewed as a "cloud," from which businesses and individuals access applications from anywhere in the world on demand

Cloud Computing in a nutshell

- BUYYA
- “Cloud is a parallel and distributed computing system consisting of a collection of inter-connected and virtualized computers that are dynamically provisioned and presented as one or more unified computing resources based on service-level agreements (SLA) established through negotiation between the service provider and consumers.”
- NIST
- a pay-per-use model for enabling available, convenient, on-demand network access to a shared pool of configurable computing resources (e.g. networks, servers, storage, applications, services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.”

Cloud Computing in a nutshell

- While there are countless other definitions, there seems to be common characteristics between the most notable ones listed above, which a cloud should have: (

(i) pay-per-use (no ongoing commitment, utility prices);

(ii) elastic capacity and the illusion of infinite resources;

(iii) self-service interface

(iv) resources that are abstracted or virtualised.

Roots of Cloud Computing

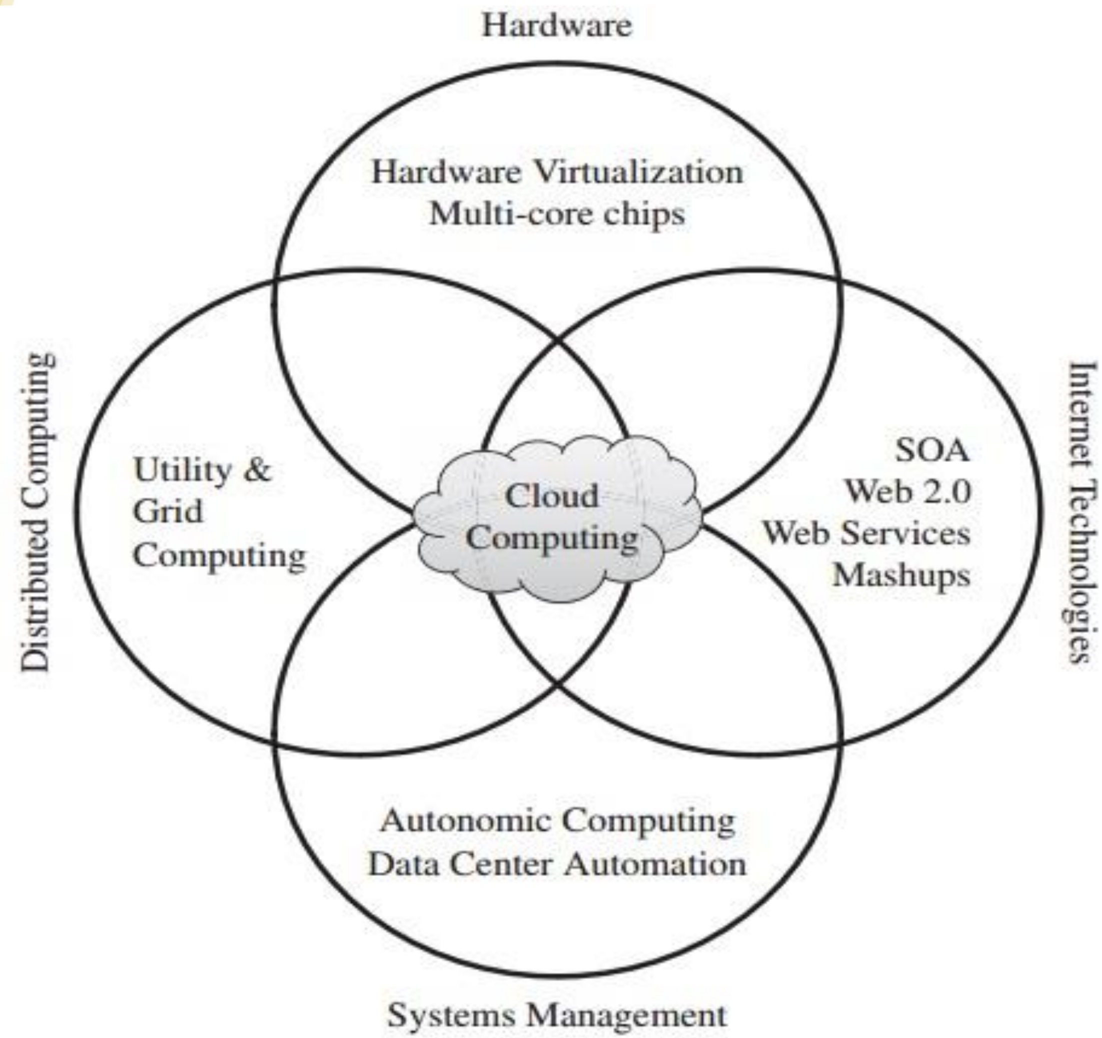

FIGURE 1.1. Convergence of various advances leading to the advent of cloud computing.

Roots of Cloud Computing

(i) Mainframe to cloud

(ii) SOA, Web Services, Web 2.0 and Mashups

(iii) Grid Computing

(iv) Utility Computing

(v)Hardware Virtualization

(vi)Virtual Appliance and OVF

(vii) Autonomic Computing

From Mainframe to cloud

- currently experiencing a switch in the IT world, from in-house generated computing power into utility-supplied computing resources delivered over the Internet as Web services
- Computing delivered as a utility can be defined as “on demand delivery of infrastructure, applications, and business processes in a security-rich, shared, scalability based computer environment over the Internet for a fee”
- Advantage to both consumer and providers
- Earlier provided timeshared mainframes , declined due to advent of fast and inexpensive microprocessors

SOA, Web Services, Web 2.0 and Mashups

- Web services can glue together applications running on different messaging product platforms, enabling information from one application to be made available to others, and enabling internal applications to be made available over the Internet.
- The purpose of a SOA is to address requirements of loosely coupled, standards-based, and protocol-independent distributed computing
- Services such user authentication, e-mail, payroll management, and calendars are examples of building blocks that can be reused and combined in a business solution in case a single, ready-made system does not provide all those features

Grid Computing

- **Grid computing** is the collection of **computer** resources from multiple locations to reach a common goal. The **grid** can be thought of as a distributed system with non-interactive workloads that involve a large number of files.

- A key aspect of the grid vision realization has been building standard Web services-based protocols that allow distributed resources to be “discovered, accessed, allocated, monitored, accounted for, and billed for..

- Issues:
- QOS, Avaibility of resource with diverse software configuration
- Soln: virtualisation

Utility Computing

- **Utility computing** is a service provisioning model in which a service provider makes **computing** resources and infrastructure management available to the customer as needed, and charges them for specific usage rather than a flat rate.
- In utility computing environments, users assign a “utility” value to their jobs, where utility is a fixed or time-varying valuation that captures various QoS constraints (deadline, importance, satisfaction).
- The service providers then attempt to maximize their own utility, where said utility may directly correlate with their profit.

Hardware Virtualisation

- Hardware virtualization allows running multiple operating systems and software stacks on a single physical platform

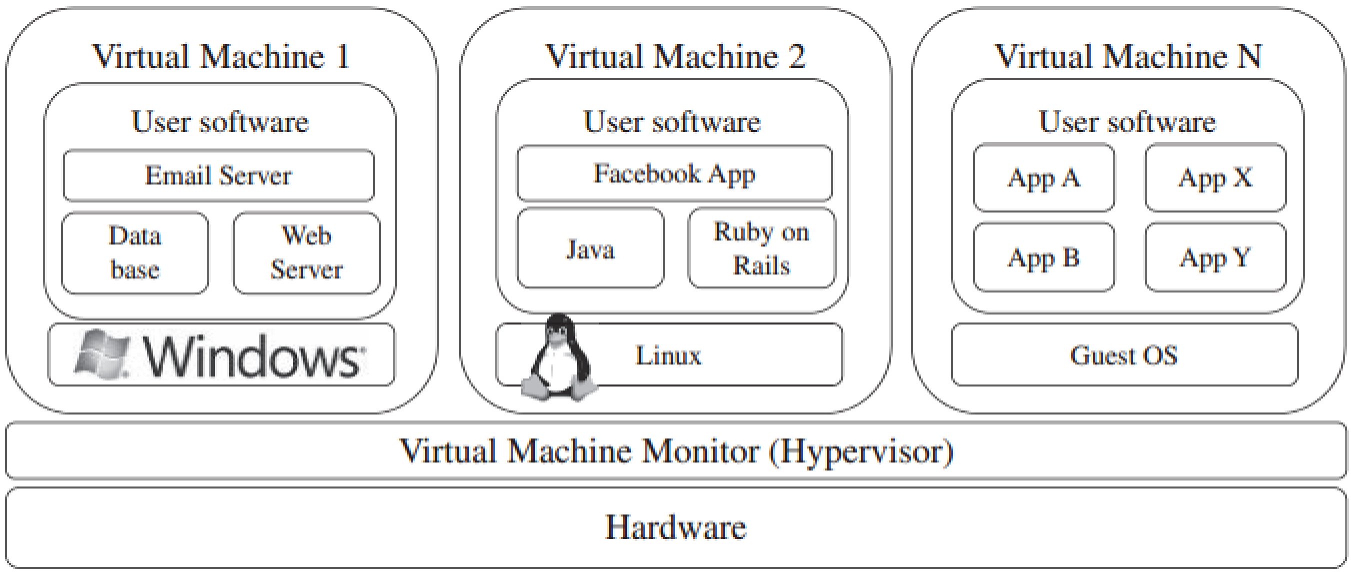

FIGURE 1.2. A hardware virtualized server hosting three virtual machines, each one running distinct operating system and user level software stack.

- 3 basic capabilities related to management of workload: isolation, Consolidation and Migration

- A number of VMM platforms exist that are the basis of many utility or cloud computing environments.

- **VMWare ESXi :**
- pioneer in virtualisation, bare metal hypervisor,
- provides advanced virtualization techniques of processor, memory, and I/O. Especially, through memory ballooning and page sharing, it can overcommit memory,

- **Xen:**
- open-source project
- It has pioneered the para-virtualization concept, on which the guest operating system, by means of a specialized kernel, can interact with the hypervisor, thus significantly improving performance

- **KVM:**
- kernel-based virtual machine (KVM) is a Linux virtualization subsystem
- Is has been part of the mainline Linux kernel since version 2.6.20, thus being natively supported by several distributions.
- In addition, activities such as memory management and scheduling are carried out by existing kernel
- KVM leverages hardware-assisted virtualization, which improves performance and allows it to support unmodified guest operating systems

Virtual Appliance and OVF(open virtual format)

- An application combined with the environment needed to run it (operating system, libraries, compilers, databases, application containers, and so forth) is referred to as a "virtual appliance."
- In a multitude of hypervisors, where each one supports a different VM image format and the formats are incompatible with one another, a great deal of interoperability issues arises.
- For instance, Amazon has its Amazon machine image (AMI) format, made popular on the Amazon EC2 public cloud. Other formats are used by Citrix XenServer, several Linux distributions that ship with KVM, Microsoft Hyper-V, and VMware ESX

Autonomic Computing

- The increasing complexity of computing systems has motivated research on autonomic computing, which seeks to improve systems by decreasing human involvement in their operation

- Autonomic, or self-managing, systems rely on monitoring probes and gauges (sensors), on an adaptation engine (autonomic manager) for computing optimizations based on monitoring data, and on effectors to carry out changes on the system.

Migration

- when and how to migrate one's application into a cloud ?
- what part or component of the IT application to migrate into a cloud and what not to migrate into a cloud ?
- what kind of customers really benefit from migrating their IT into the cloud ?

The Seven-Step Model of Migration into a Cloud

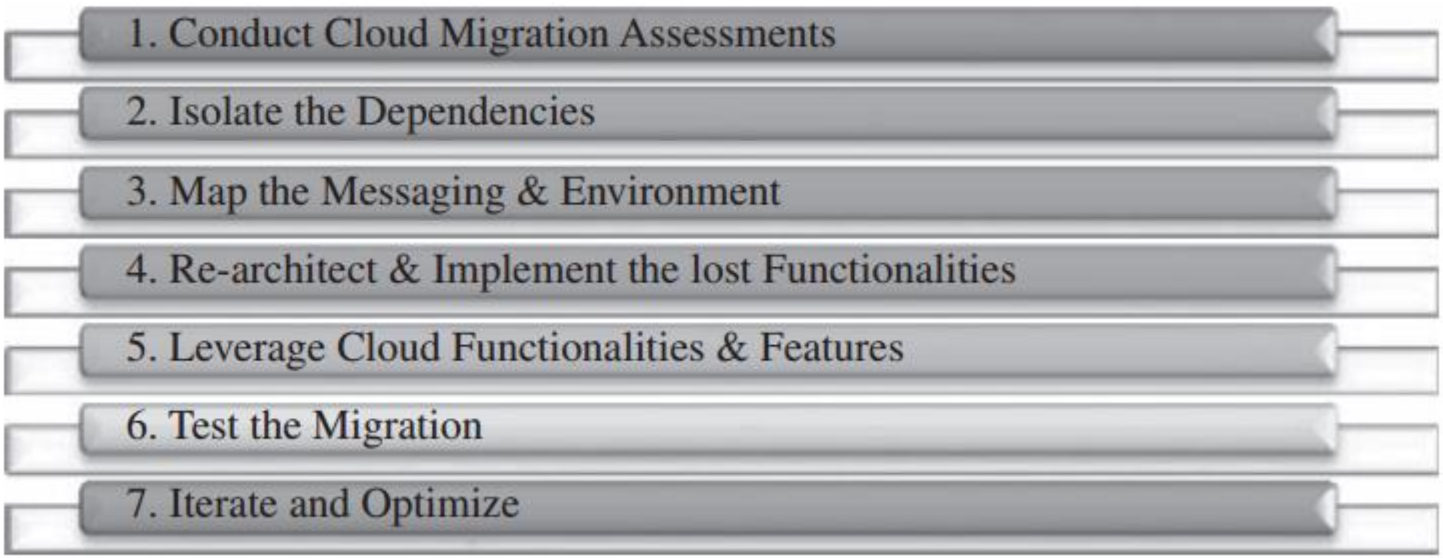

FIGURE 2.4. The Seven Step Model of Migration into the Cloud. (*Source:* Infosys Research.)

The Seven-Step Model of Migration into a Cloud

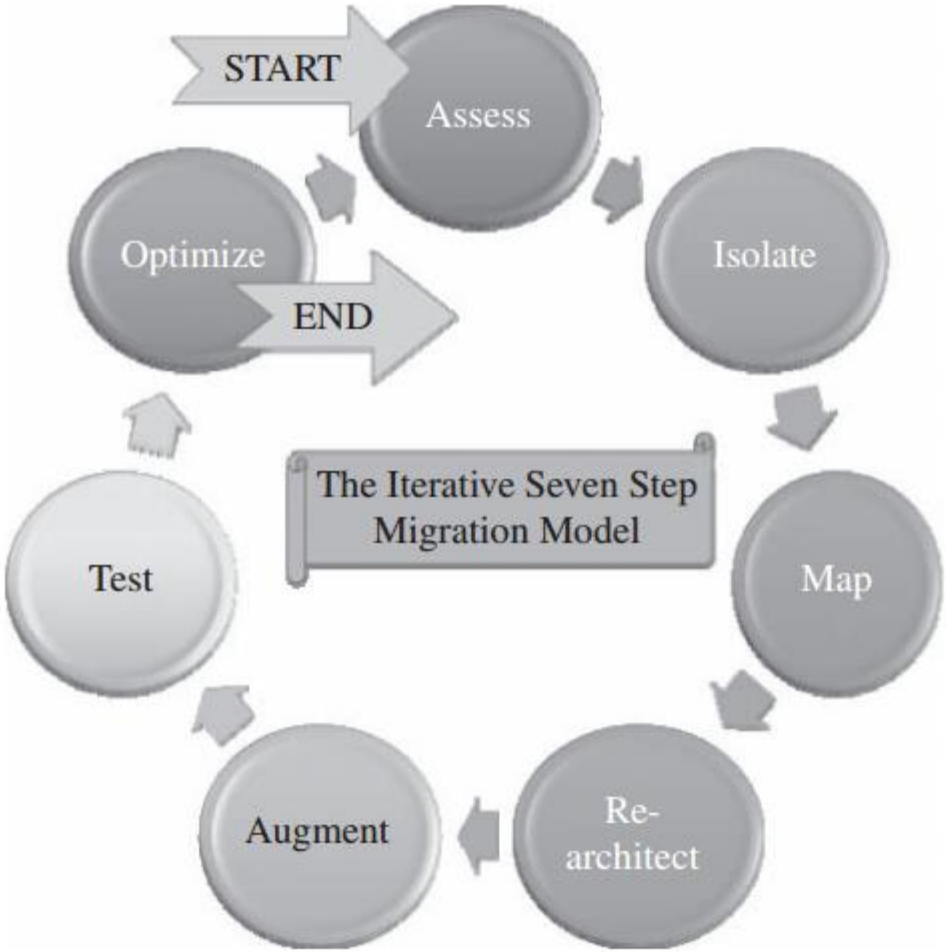

FIGURE 2.5. The iterative Seven step Model of Migration into the Cloud. (*Source:* Infosys Research.)

The Seven-Step Model of Migration into a Cloud

Step 1

- Cloud migration assessments comprise assessments to understand the issues involved in the specific case of migration at the application level or the code, the design, the architecture, or usage levels.

- These assessments are about the cost of migration as well as about the ROI that can be achieved in the case of production version.

Step 2

- isolating all systemic and environmental dependencies of the enterprise application components within the captive data center

Step 3

- generating the mapping constructs between what shall possibly remain in the local captive data center and what goes onto the cloud.

The Seven-Step Model of Migration into a Cloud

Step 4

- substantial part of the enterprise application needs to be rearchitected, redesigned, and reimplemented on the cloud.

Step 5

- We leverage the intrinsic features of the cloud computing service to augment our enterprise application in its own small ways.

Step 6

- we validate and test the new form of the enterprise application with an extensive test suite that comprises testing the components of the enterprise application on the cloud as well

Step 7

- Test results could be positive or mixed.
- In the latter case, we iterate and optimize as appropriate. After several such optimizing iterations, the migration is deemed successful

The Seven-Step Model of Migration into a Cloud

Assess	Isolate	Map	Re-Architect	Augment	Test	Optimize
• Cloudonomics • Migration Costs • Recurring Costs • Database data segmentation • Database Migration • Functionality migration • NFR Support	• Runtime Environment • Licensing • Libraries Dependency • Applications Dependency • Latencies Bottlenecks • Performance bottlenecks • Architectural Dependencies	• Messages mapping: marshalling & de-marshalling • Mapping Environments • Mapping libraries & runtime approximations	• Approximate lost functionality using cloud runtime support API • New Usecases • Analysis • Design	• Exploit additional cloud features • Seek Low-cost augmentations • Autoscaling • Storage • Bandwidth • Security	• Augment Test Cases and Test Automation • Run Proof-of-Concepts • Test Migration strategy • Test new testcases due to cloud augmentation • Test for Production Loads	• Optimize– rework and iterate • Significantly satisfy cloudonomics of migration • Optimize compliance with standards and governance • Deliver best migration ROI • Develop roadmap for leveraging new cloud features

FIGURE 2.6. Some details of the iterative Seven Step Model of Migration into the Cloud.