UNIT 4 RESOURCE POOLING, SHARING AND PROVISIONING

4.1 INTRODUCTION

Resource pooling is the one of the essential attributes of Cloud Computing technology which separates cloud computing approach from the traditional IT approach. Resource pooling along with virtualization and sharing of resources, leads to dynamic behavior of the cloud. Instead of allocating resources permanently to users, they are dynamically provisioned on a need basis. This leads to efficient utilization of resources as load or demand changes over a period of time. Multi-tenancy allows a single instance of an application software along with its supporting infrastructure to be used to serve multiple customers. It is not only economical and efficient to the providers, but may also reduce the charges for the consumers.

4.2 OBJECTIVES

After going through this unit, you should be able to:

- Know about Resources pooling –Compute, Storage and Network pools
- Know about Resources pooling architectures
- Know about Resources sharing techniques
- ~~Describe~~ Know about various provisioning approaches
- ~~Describe how VM resizing is performed~~
- Know on the Resource Pricing

4.3 RESOURCE POOLING

Resource pool is a collection of resources available for allocation to users. All types of resources – compute, network or storage, can be pooled. It creates a layer of abstraction for consumption and presentation of resources in a consistent manner. A large pool of physical resources is maintained in cloud data centers and presented to users as virtual services. Any resource from this pool may be allocated to serve a single user or application, or can be even shared among multiple users or applications. Also, instead of allocating resources permanently to users, they are dynamically provisioned on a need basis. This leads to efficient utilization of resources as load or demand changes over a period of time.

For creating resource pools, providers need to set up strategies for categorizing and management of resources. The consumers have no control or knowledge of the actual locations where the physical resources are located. Although some service providers may provide choice for geographic location at higher abstraction level like- region, country, from where customer can get resources. This is generally possible with large service providers who have multiple data centers across the world.

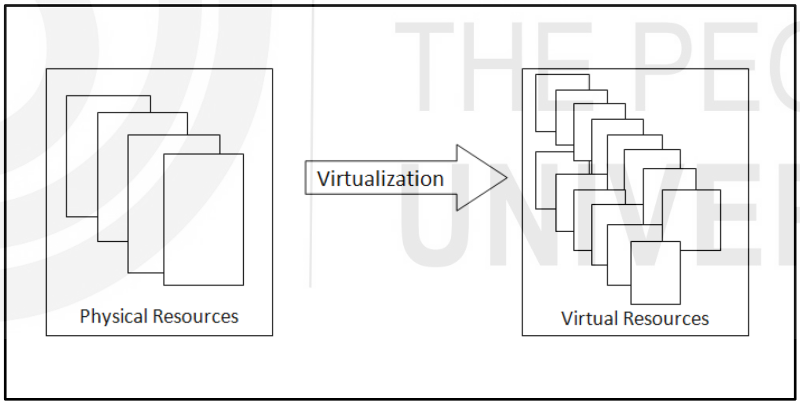

Fig 4.1 Pooling of Physical and Virtual Resources

4.4 RESOURCE POOLING ARCHITECTURE

Each pool of resources is made by grouping multiple identical resources for example – storage pools, network pools, server pools etc. A resource pooling architecture is then built by combining these pools of resources. An automated system is needed to be established in order to ensure efficient utilization and synchronization of pools.

Computation resources are majorly divided into three categories – Server , Storage and Network. Sufficient quantities of physical resources of all three types are hence maintained in a data center.

4.4.1 Server Pools

Server pools are composed of multiple physical servers along with operating system, networking capabilities and other necessary software installed on it. Virtual machines are then configured over these servers and then combined to create virtual server pools. Customers can choose virtual machine configurations from the available templates (provided by cloud service provider) during provisioning. Also, dedicated processor and memory pools are created from processors and memory devices and maintained separately. These processor and memory components from their respective pools can then be linked to virtual servers when demand for increased capacity arises. They can further be returned to the pool of free resources when load on virtual servers decreases.

4.4.2 Storage Pools

Storage resources are one of the essential components needed for improving performance, data management and protection. It is frequently accessed by users or applications as well as needed to meet growing requirements, maintaining backups, migrating data, etc.

Storage pools are composed of file based, block based or object based storage made up of storage devices like- disk or tapes and available to users in virtualized mode.

1. File based storage – it is needed for applications that require file system or shared file access. It can be used to maintain repositories, development, user home directories, etc.

2. Block based storage – it is a low latency storage needed for applications requiring frequent access like databases. It uses block level access hence needs to be partitioned and formatted before use.

3. Object based storage – it is needed for applications that require scalability, unstructured data and metadata support. It can be used for storing large amounts of data for analytics, archiving or backups.

4.4.3 Network Pools

Resources in pools can be connected to each other, or to resources from other pools, by network facility. They can further be used for load balancing, link aggregation, etc.

Network pools are composed of different networking devices like- gateways, switches, routers, etc. Virtual networks are then created from these physical networking devices and offered to customers. Customers can further build their own networks using these virtual networks.

Generally, dedicated pools of resources of different types are maintained by data centers. They may also be created specific to applications or consumers. With the increasing number of resources and pools, it becomes very complex to manage and organize pools. Hierarchical structure can be used to form parent-child, sibling, or nested pools to facilitate diverse resource pooling requirements.

Check Your Progress 1

1. What is a Resource pool ?
2. Explain Respource pooling architecture.
3. What are the various types of storage pools available. Explain.

4.5 RESOURCE SHARING

Cloud computing technology makes use of resource sharing in order to increase resource utilization. At a time, a huge number of applications can be running over a pool. But they may not attain peak demands at the same time. Hence, sharing them among applications can increase average utilization of these resources.

Although resource sharing offers multiple benefits like – increasing utilization, reduces cost and expenditure, but also introduces challenges like – assuring quality of service (QoS) and performance. Different applications competing for the same set of resources may affect run time behavior of applications. Also, the performance parameters like- response and turnaround time are difficult to predict. Hence, sharing of resources requires proper management strategies in order to maintain performance standards.

4.5.1 Multi-tenancy

Multi-tenancy is one of the important characteristics found in public clouds. Unlike traditional single tenancy architecture which allocates dedicated resources to users, multi-tenancy is an architecture in which a single resource is used by multiple tenants (customers) who are isolated from each other. Tenants in this architecture are logically separated but physically connected. In other words, a single instance of a software can run on a single server but can server multiple tenants. Here, data of each tenant is kept separately and securely from each other. Fig 1 shows single tenancy and multi-tenancy scenarios.

Multi-tenancy leads to sharing of resources by multiple users without the user being aware of it. It is not only economical and efficient to the providers, but may also reduce the charges for the consumers. Multi-tenancy is a feature enabled by various other features like- virtualization, resource sharing, dynamic allocation from resource pools.

In this model, physical resources cannot be pre-occupied by a particular user. Neither the resources are allocated to an application dedicatedly. They can be utilized on a temporary basis by multiple users or applications as and when needed. The resources are released and returned to a pool of free resources when demand is fulfilled which can further be used to serve other requirements. This increases the utilization and decreases investment.

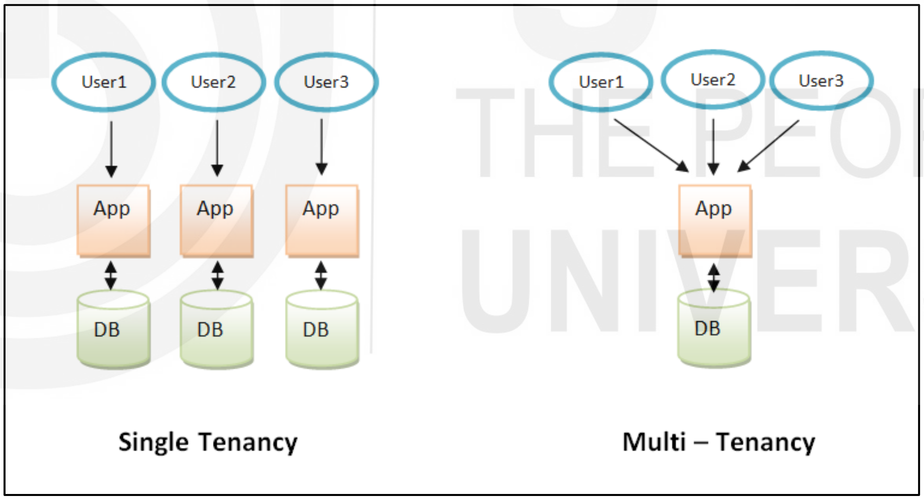

Fig 1: Single tenancy Vs Multi-tenancy

4.5.2 Types of Tenancy

There are two types of tenancy – Single tenancy and multi-tenancy.

In single tenancy architecture, a single instance of an application software along with its supporting infrastructure, is used to serve a single customer. Customers have their own independent instances and databases which are dedicated to them. Since there is no sharing with this type of tenancy, it provides better security but costs more to the customers.

In multi-tenancy architecture, a single instance of an application software along with its supporting infrastructure, can be used to serve multiple customers. Customers share a single instance and database. Customer's data is isolated from each other and remains invisible to others. Since users are sharing the resources, it costs less to them as well as is efficient for the providers.

Multi-tenancy can be implemented in three ways –

1. **Single multi-tenant database** - It is the simplest form where a single application instance and a database instance is used to host the tenants. It is a highly scalable architecture where more tenants can be added to the. It also reduces cost due to sharing of resources but increases operational complexity.

2. **One database per tenant** – It is another form where a single application instance and separate database instances are used for each tenant. Its scalability is low and costs higher as compared to a single multi-tenant database due to overhead included by adding each database. Due to separate database instances, its operational complexity is less.

3. **One app instance and one database per tenant** - It is the architecture where the whole application is installed separately for each tenant. Each tenant has its own separate app and database instance. This allows a high degree of data isolation but increases the cost.

4.5.3 Tenancy at Different Level of Cloud Services

Multi-tenancy can be applied not only in public clouds but also in private or community deployment models. Also, it can be applied to all three service models – Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS). Multi-tenancy when performed at infrastructure level, makes other levels also multi-tenant to certain extent.

Multi-tenancy at IaaS level can be done by virtualization of resources and customers sharing the same set of resources virtually without affecting others. In this, customers can share infrastructure resources like- servers, storage and network.

Multi-tenancy at PaaS level can be done by running multiple applications from different vendors over the same operating system. This removes the need for separate virtual machine allocation and leads to customers sharing operating systems. It increases utilization and ease maintenance.

Multi-tenancy at SaaS level can be done by sharing a single application instance along with a database instance. Hence a single application serves multiple customers. Customers may be allowed to customize some of the functionalities like- change view of interface but they are not allowed to edit applications since it is serving other customers also.

Check Your Progress 2

1. What is a Single tenancy and Multi-tenancy?
2. Explain teanancy at different service levels of cloud.

4.6 RESOURCE PROVISIONING AND APPROACHES

Resource provisioning is the process of allocating resources to applications or the customers. When a customer demands resources, they must be provisioned automatically from a shared pool of configurable resources. Virtualization technology makes the allocation of resources faster. It allows creation of virtual machines in minutes, where customers can choose configurations of their own. Proper management of resources is needed for rapid provisioning.

Resource provisioning is required to be done efficiently. Physical resources are not allocated to users directly. Instead, they are made available to virtual machines, which in turn are allocated to users and applications. Resources can be assigned to virtual machines using various provisioning approaches. There can be three types of resources provisioning approaches– static, dynamic and hybrid.

4.6.1 Static Approach

In static resource provisioning, resources are allocated to virtual machines only once, at the beginning according to user's or application's requirement. It is not expected to change further. Hence, it is suitable for applications that have predictable and static workloads. Once a virtual machine is created, it is expected to run without any further allocations.

Although there is no runtime overhead associated with this type of provisioning, it has several limitations. For any application, it may be very difficult to predict future workloads. It may lead to over-provisioning or under-provisioning of resources. Under-provisioning is the scenario when actual demand for resources exceeds the available resources. It may lead to service downtime or application degradation. This problem may be avoided by reserving sufficient resources in the beginning. But reserving large amounts of resources may lead to another problem called Over-provisioning. It is a scenario in which the majority of the resources remain un-utilized. It may lead to inefficiency to the service provided and incurs unnecessary cost to the consumers. Fig 2 shows the under-provisioning and Fig 3 shows over-provisioning scenarios.

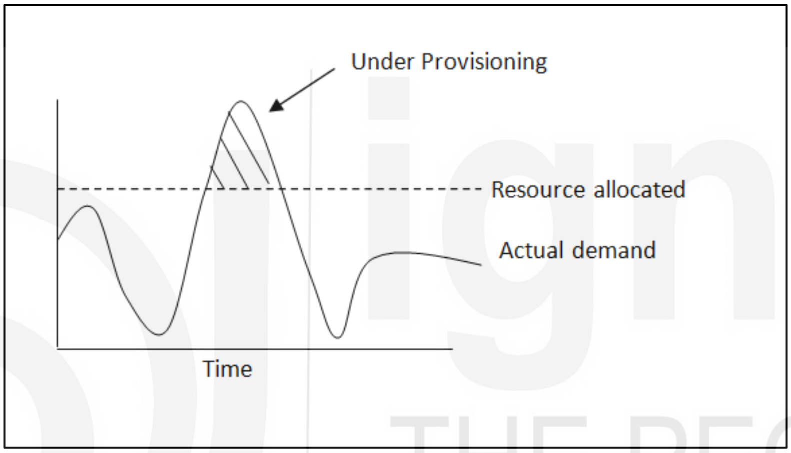

Fig 2: Problem of Resource Under-provisioning

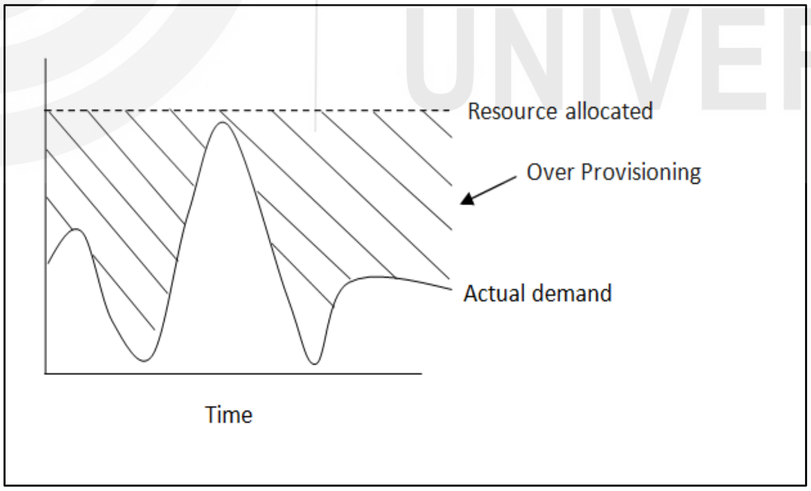

Fig 3: Problem of Resource Over-provisioning

4.6.2 Dynamic Approach

In dynamic provisioning, as per the requirement, resources can be allocated or de-allocated during run-time. Customers in this case don't need to predict resource requirements. Resources are allocated from the pool when required and removed from the virtual machine and returned back to the pool of free resources when no more are required. This makes the system elastic. This approach allows customers to be charged per usage basis.

Dynamic provisioning is suited for applications where demands for resources are un-predictable or frequently varies during run-time. It is best suited for scalable applications. It can adapt to changing needs at the cost of overheads associated with run-time allocations. This may lead to a small amount of delay but solves the problem of over-provisioning and under-provisioning.

4.6.3 Hybrid Approach

Dynamic provisioning although solves the problems associated with static approach but may lead to overheads at run-time. Hybrid approach solves the problem by combining the capabilities of static and dynamic provisioning. Static provisioning can be done in the beginning when creating a virtual machine in order to limit the complexity of provisioning. Dynamic provisioning can be done later for re-provisioning when the workload changes during run-time. This approach can be efficient for real-time applications.

4.7 VM SIZING

Virtual machine (VM) sizing is the process of estimating the amount of resources that a VM should be allocated. Its objective is to make sure that VM capacity is kept proportionate to the workload. This estimation is based upon various parameters specified by the customer. VM sizing is done at the beginning in case of static provisioning. In dynamic provisioning, VM size can be changed depending upon the application workload.

There are two ways to do VM sizing –

1. Individual VM based – In this case, depending upon the previous workload patterns, resources are allocated VM-by-VM initially. Resources can be later allocated from the pool when load reaches beyond expectations.
2. Joint-VM based – In this case, allocation to VMs are done in a combined way. Resources assigned to a VM initially can be reassigned to another VM hosted on the same physical machine. Hence it leads to overall efficient utilization.

Check Your Progress 3

1. What is a Resource Provisioning ?
2. Explain various resource provisioning approaches.
3. Explain the problems of Over-provisioning and Under-provisioning.

4.8 SUMMARY

In this unit an important attribute of Cloud Computing technology called Resource pooling is discussed. It is a collection of resources available for allocation to users. A large pool of physical resources - storage, network and server pools are maintained in cloud data centers and presented to users as virtual services. Resources may be allocated to serve a single user or application, or can be even shared among multiple users or applications. Resources can be assigned to virtual machines using static, dynamic and hybrid provisioning approaches.

Answers to Check Your Progress 1

1. Resource pool is a collection of resources available for allocation to users. All types of resources – compute, network or storage, can be pooled. It creates a layer of abstraction for consumption and presentation of resources in a consistent manner. A large pool of physical resources is maintained in cloud data centers and presented to users as virtual services. Any resource from this pool may be allocated to serve a single user or application, or can be even shared among multiple users or applications. Also, instead of allocating resources permanently to users, they are dynamically provisioned on a need basis. This leads to efficient utilization of resources as load or demand changes over a period of time.

2. A resource pooling architecture is composed of Server, storage and network pools. An automated system is needed to be established in order to ensure efficient utilization and synchronization of pools.

a) Server pools - They are composed of multiple physical servers along with operating system, networking capabilities and other necessary software installed on it.
b) Storage pools – They are composed of file based, block based or object based storage made up of storage devices like- disk or tapes and available to users in virtualized mode.
c) Network pools - They are composed of different networking devices like- gateways, switches, routers, etc. Virtual networks are then created from these physical networking

devices and offered to customers. Customers can further build their own networks using these virtual networks.

3. Storage pools are composed of file based, block based or object based storage.

a) File based storage – it is needed for applications that require file system or shared file access. It can be used to maintain repositories, development, user home directories, etc.
b) Block based storage – it is a low latency storage needed for applications requiring frequent access like databases. It uses block level access hence needs to be partitioned and formatted before use.
c) Object based storage – it is needed for applications that require scalability, unstructured data and metadata support. It can be used for storing large amounts of data for analytics, archiving or backups.

Answers to Check Your Progress 2

1. In single tenancy architecture, a single instance of an application software along with its supporting infrastructure, is used to serve a single customer. Customers have their own independent instances and databases which are dedicated to them. Since there is no sharing with this type of tenancy, it provides better security but costs more to the customers.
In multi-tenancy architecture, a single instance of an application software along with its supporting infrastructure, can be used to serve multiple customers. Customers share a single instance and database. Customer's data is isolated from each other and remains invisible to others. Since users are sharing the resources, it costs less to them as well as is efficient for the providers.

2. Multi-tenancy can be implemented at all the service levels.

a) Multi-tenancy at IaaS level – It can be done by virtualization of resources and customers sharing the same set of resources virtually without affecting others. In this way, customers can share infrastructure resources.

b) Multi-tenancy at PaaS level- It can be done by running multiple applications from different vendors over the same operating system. This removes the need for separate virtual machine allocation and leads to customers sharing operating systems.

c) Multi-tenancy at SaaS level- It can be done by sharing a single application instance along with a database instance. Hence a single application serves multiple customers.

Answers to Check Your Progress 3

1. Resource provisioning is the process of allocating resources to applications or the customers. When a customer demands resources, they must be provisioned automatically from a shared pool of configurable resources.

2. There can be three types of resources provisioning approaches– static, dynamic and hybrid.

   a) In static resource provisioning, resources are allocated to virtual machines only once, at the beginning according to user’s or application’s requirement. It is not expected to change further. It is suitable for applications that have predictable and static workloads.

   b) In dynamic provisioning, as per the requirement, resources can be allocated or de-allocated during run-time. Customers in this case don’t need to predict resource requirements. It is suited for applications where demands for resources are un-predictable or frequently varies during run-time.

   c) Hybrid Provisioning combines the capabilities of static and dynamic provisioning. Static provisioning is done in the beginning when creating virtual machines in order to limit the complexity of provisioning. Dynamic provisioning is done later for re-provisioning when the workload changes during run-time. This approach can be efficient for real-time applications.

3. Under-provisioning is the scenario when actual demand for resources exceeds the available resources. It may lead to service downtime or application degradation. This problem may be avoided by reserving sufficient resources in the beginning.

Reserving large amounts of resources may lead to another problem called Over-provisioning. It is a scenario in which the majority of the resources remain un-utilized. It may lead to inefficiency to the service provided and incurs unnecessary cost to the consumers.