Program : **B.E**

Subject Name: **Cloud Computing**

Subject Code: **CS-8002**

Semester: **8th**

Subject Notes
CS 8002 - Cloud Computing

Unit 5

Market Based Management of Clouds

Cloud marketing is the process of an organizations efforts to market their goods and services online through integrated digital experiences, by which they are specialized for every single end user. The aim is to use advertising methods to target customers via online applications through social media websites such as Facebook, Twitter and various online portals to target consumers. Cloud marketing platforms are supported with third parties who maintain the platform. Cloud Marketing was established by a company in Raleigh, North Carolina calling it SharedVue Marketing Technologies. The marketers targeting clients need to ensure their material is compatible with all electronic media devices.

Advantages of Cloud marketing for business

- **Cost Effectiveness**

Cloud marketing enables a business to decrease the cost of marketing distribution materials; these include sending catalogues and magazines to consumers as digital media, allowing a business to send promotional content through digital formats, which enables a faster and cheaper approach to target consumers. The cost reduces the printing costs and increases efficiency using online material accessible continuously.

- **Customization**

Customization allows a business to creatively use interactive means to create an relevant and effective advertising approach when targeting a consumer. Customization includes social media sites such as Facebook to customize pages to send to friends or the public over the internet. Marketers can combine data through third party data sources, including email and surveys, to visualize the consumer's experience.

- **Time**

Time is vital for targeting customers. Advertising using traditional methods, such as posters and surveys, have limited time before they often become invalid. Cloud marketing enables a business to produce advertising when required. The material can easily be removed and if a campaign or season is over, the material can be erased from the internet or adapted to enhance material to the end user, linking with the customization element to ensure the marketing material is fit for its purpose, and delivered at the correct time.

Disadvantages of Cloud Marketing for business

- **User Experience**

When a company markets their products and services to a consumer, the end consumer is not able to touch or physically manage the product or service. The experience could potentially lay of customers that have been targeted, if the businesses efforts have not satisfied the consumers decision to buy the merchandise. The material content would vary on the device, as compatibility and operating systems will affect the material content being delivered.

- **Fraudulent Material**

Internet fraud has grown rapidly globally, faster than the internet. More and more fraudulent criminals can send promotional pop ups in the form of online advertising on the World Wide Web to attract web traffic to display promotional content. The malware attacks can lay off customers responding to marketing material posted to their devices. Labor MP Chris Evans said: 'Copycat websites are a part of a growing industry which exists purely to trick the public out of their hard-earned money.

- **Digital Divide**

Digital divide is the partition between a given population within their use of information technology. This can be due to factors including:

- Geographic
- Cultural
- Economic growth
- Democracy
- Disabilities

This limits a business's performance to market their goods and services globally to new locations if there is limited access to information technology in certain locations. The segment of consumers would be unable to experience and view online marketing methods from a business or resources resulting in adopted a traditional method of leaflets and bill boards known as direct marketing.

- **Cloud Marketing Plan Strategy**

Strategy is the direction of action which will achieve a goal or objective. The strategy for cloud marketing is broken down into 4 key elements.

- **Establishing the goals**

The first steps into cloud marketing include finding the objective or goal for the marketing project. The proposer would need to clearly state the objectives, which can be retained in quantitative or qualitative data. By establishing the goal and objectives of the marketing campaign, this limits the plan being deployed haphazardly.

- **Development**

The development stage is where the marketing team creates the graphics and media material. The web development team find a method to post the material onto the world wide web or online source. The marketing ad would need to meet its main objective and purpose, the development team will need to develop and plan to make the material visually appealing.

- **Maintenance**

The maintenance step will require updating whilst the material is online. it will require continuous upkeep. Cloud marketing techniques include regular updating to ensure they are reaching their end user and have a valid subject. Marketing members are responsible for moderating any discussion boards and keeping content updated increasing the validity.

- **Evaluation**

Throughout the duration of the marketing material, the message would need to be evaluated to determine how successful it has been to the end user. The outcome should be established in the strategy allowing the marketer to adapt and increase the overall efficiency of the cloud marketing method.

Federated Clouds/Inter Cloud

Cloud Federation refers to the unionization of software, infrastructure and platform services from disparate networks that can be accessed by a client via the internet. The federation of cloud resources is facilitated through network gateways that connect public or external clouds, private or internal clouds (owned by a single entity) and/or community clouds (owned by several cooperating entities); creating a hybrid cloud computing environment. It is important to note that federated cloud computing services still rely on the existence of physical data centers.

- **Cloud Federation Benefits**

The federation of cloud resources allows clients to optimize enterprise IT service delivery. The federation of cloud resources allows a client to choose the best cloud services provider, in terms of flexibility, cost and availability of services, to meet a particular business or technological need within their organization. Federation across different cloud resource pools allows applications to run in the most appropriate infrastructure environments. The federation of cloud resources also allows an enterprise to distribute workloads around the globe, move data between disparate networks and implement innovative security models for user access to cloud resources.

- **Cloud Federation Implementation**

One weakness that exists in the federation of cloud resources is the difficulty in brokering connectivity between a client and a given external cloud provider, as they each possess their own unique network addressing scheme. To resolve this issue, cloud providers must grant clients the permission to specify an addressing scheme for each server the cloud provider has extended to the internet. This provides customers with the ability to access cloud services without the need for reconfiguration when using resources from different service providers. Cloud federation can also be implemented behind a firewall, providing clients with a menu of cloud services provided by one or more trusted entities.

Cloud Federation Stack

With the aid of Cloud computing technology, businesses and institutions make compute resources available to customers and partners to create more capable, scalable, flexible, and cost effective environments for application development and hosting. Cloud computing continues the trend started with on-demand, strategic outsourcing, and grid computing, to provideIT resources as a standardized commodity, targeting real-time delivery of infrastructure and platform services. A next stepin this evolution is to have cooperating providers of Cloud services in which a customer request submitted to one Cloudprovider is fulfilled by another, under mediation of a brokering structure. This latter idea invokes a federation ofCloud domains providing a service analogous to that of interoperating grid resources created for a similar goal by researchinstitutions using grid brokers in the grid computing framework.

To illustrate how this works, consider a business providing a SaaS offering from a private or public Cloud. Users submit requests to the application layer which assesses if sufficient local resources are available to service the requests within a specified time. If the application layer cannot meet its service goals it can optionally fulfill the requests through an independent SaaS layer provider of the same service as indicated by the horizontal (federation. Results are returned to the user as if locally produced by the application executing in Cloud A. Federation at the SaaS layer is analogous to the use in traditional business of 'sub' or 'peer' contractors who supply equivalent final parts or services to the primary provider facilitating elasticity to support a dynamic market. While this approach is common in industry sectors that produce goods or services such as manufacturing or publishing, it is not as common in software due to lack of standard interfaces and insufficient market forces to motivate sharing at the service layer. An application layer under stress also has a second option to increase capacity through delegation. In this service abstraction, the application layer works together with its underlying layers to provide the required computing needs. In delegation, the application layer asks the PaaS layer in the local Cloud for additional resources. The request for more resources may befulfilled in multiple ways depending on availability in the current Cloud. The PaaS layer can delegate to the local IaaS layer a request for more raw virtual machines and then provision the necessary platform software. If sufficient resources are not available locally the PaaS layer can attempt to acquire them from another Cloud in the federation through brokering at the PaaS layer.

In a typical scenario, the PaaS layer represents executing middleware such as web application containers and other application execution platforms, or distributed data applications. Here a more general view of federation is needed in which these support programs and environments form the federations between the Clouds in a way that isolates them from the underlying infrastructure layer. Some current middleware products, such as web application servers (e.g., IBM WebSphere Application Server or Oracle Fusion Middleware), provide isolation or lightweight virtualization from the underlying hardware and allow applications to dynamically expand across machines increasing capacity.

While attractive from a business perspective, this federated Cloud model requires new technologies to work efficiently.Because it is a layered model, an important part of the design is to maintain isolation of concerns between layers. For example, the SaaS application delivers a result to the customer in a certain response time. It is aware of the aggregate processing and network transmissions necessary to meet the delivery time. But the application does not need to know the details of the underlying infrastructure. Thus, it is necessary to translate requirements at the application to those understood by the PaaS and IaaS layers. This is accomplished through empirical modeling and experiments that map metrics of application performance such as response time onto the middleware and compute resource requirements understood by the PaaS or IaaS layer.

One challenge to making the operation of delegation work is to introduce a standardized form of expressing inter-layermappings. Some work along this line is contained in the manifestapproach used by the Reservoir project. A related issue is how to choose between delegation and federation when both options are available. Selection criteria suchas the mapping of performance metrics may be combined with policies as discussed in Sections 2 and 5. Another challenge is defining the protocols and policies for the inter-Cloud brokering required to join each layer in a federation. Considers brokering at different Cloud service layers and then proceeds to the inner workings and policy issues by whichbrokers expose and share Cloud services and resources.

Third Party Cloud Services: Google App Engine

Google App Engine is an application hosting and development platform that powers everything from enterprise web applications to mobile games, using the same infrastructure that powers Google's global-scale web applications. Developers know that time-to-market is critical to success, and with Google App Engine's simple development, robust APIs and worry-free hosting, you can accelerate your application development and take advantage of simple scalability as the application grows. With support for Python, Java, and Go, you don't have to change the way you work. Your application can take advantage of powerful APIs, High Replication data storage, and a completely hands-free hosting environment that automatically scales to meet any demand, whether you're serving several users or several million.

Google App Engine makes it easy to take your app ideas to the next level.

- **Quick to start**

With no software or hardware to buy and maintain, you can prototype and deploy applications to your users in a matter of hours.

- **Simple to use**

Google App Engine includes the tools you need to create, test, launch, and update your apps.

- **Rich set of APIs**

Build feature-rich services faster with Google App Engine's easy-to-use APIs.

- **Immediate scalability**

There's almost no limit to how high or how quickly your app can scale.

- **Pay for what you use**

Get started without any upfront costs with App Engine's free tier and pay only for the resources you use as your application grows.

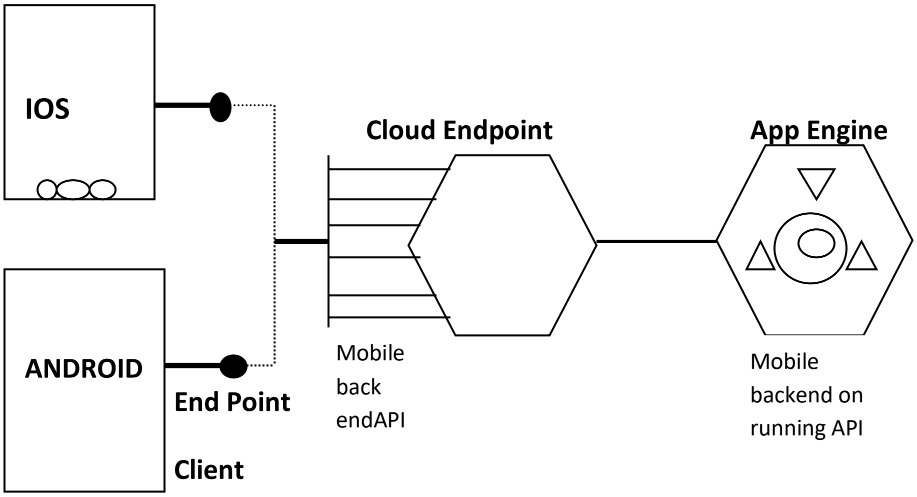

Figure: 5.1 Google App Engine Model

Microsoft Azure

Microsoft Azure is Microsoft's application platform for the public cloud. The goal of this article is to give you a foundation for understanding the fundamentals of Azure, even if you don't know anything about cloud computing.

- **The Components of Azure**

Azure groups services into categories in the Management Portal and on various visual aids like the What Is Azure Info graphic. The Management Portal is what you use to manage most (but not all) services in Azure.

This article will use a different organization to talk about services based on similar function, and to call out important sub-services that are part of larger ones.

- **Management Portal**

Azure has a web interface called the Management Portal that allows administrators to access and administer most, but not all Azure features. Microsoft typically releases the newer UI portal in beta before retiring an older one. The newer one is called the "Azure Preview Portal".

There is typically a long overlap when both portals are active. While core services will appear in both portals, not all functionality may be available in both. Newer services may show up in the newer portal first and older services and functionality may only exist in the older one. The message here is that if you don't find something in the older portal, check the newer one and vice-versa.

- **Compute**

One of the most basic things a cloud platform does is execute applications. Each of the Azure compute models has its own role to play.

You can use these technologies separately or combine them as needed to create the right foundation for your application. The approach you choose depends on what problems you're trying to solve.

The ability to create a virtual machine on demand, whether from a standard image or from one you supply, can be very useful. This approach, commonly known as Infrastructure as a Service (IaaS), is what Azure Virtual Machines provides.

To create a VM, you specify which VHD to use and the VM's size. You then pay for the time that the VM is running. You pay by the minute and only while it's running, though there is a minimal storage charge for keeping the VHD available. Azure offers a gallery of stock VHDs (called "images") that contain a bootable operating system to start from. These include Microsoft and partner options, such as Windows Server and Linux, SQL Server, Oracle and many more. You're free to create VHDs and images, and then upload them yourself. You can even upload VHDs that contain only data and then access them from your running VMs.

Wherever the VHD comes from, you can persistently store any changes made while a VM is running. The next time you create a VM from that VHD, things pick up where you left off. The VHDs that back the Virtual Machines are stored in Azure Storage blobs, which we talk about later. That means you get redundancy to ensure your VMs won’t disappear due to hardware and disk failures. It's also possible to copy the changed VHD out of Azure, then run it locally.

Your application runs within one or more Virtual Machines, depending on how you created it before or decide to create it from scratch now.

This quite general approach to cloud computing can be used to address many different problems

Hadoop Introduction

Hadoop is an Apache open source framework written in java that allows distributed processing of large datasets across clusters of computers using simple programming models. A Hadoop frame-worked application works in an environment that provides distributed storage and computation across clusters of computers. Hadoop is designed to scale up from single server to thousands of machines, each offering local computation and storage.

Hadoop Architecture

Hadoop framework includes following four modules:

- Hadoop Common: These are Java libraries and utilities required by other Hadoop modules. These libraries provide filesystem and OS level abstractions and contains the necessary Java files and scripts required to start Hadoop.
- Hadoop YARN: This is a framework for job scheduling and cluster resource management.
- Hadoop Distributed File System (HDFS™): A distributed file system that provides high-throughput access to application data.
- Hadoop MapReduce: This is YARN-based system for parallel processing of large data sets.

We can use following diagram to depict these four components available in Hadoop framework figure 5.2.

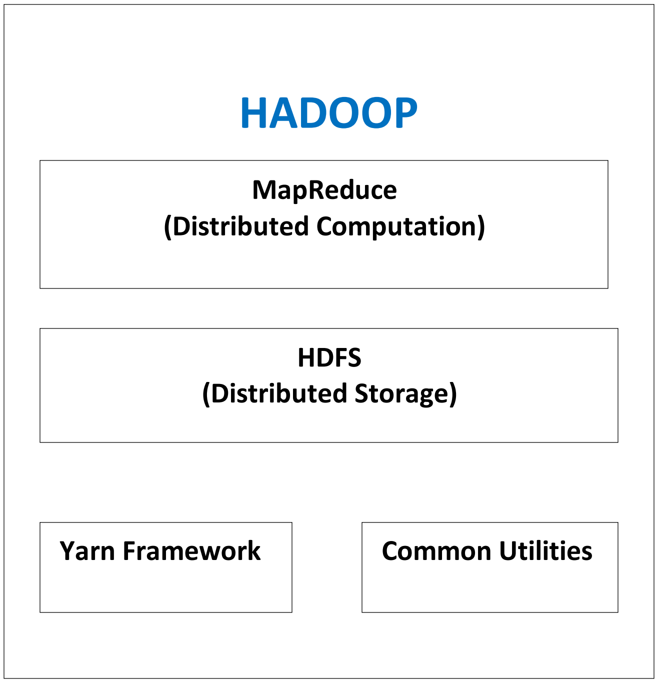

Figure: 5.2 Hadoop Model

Since 2012, the term "Hadoop" often refers not just to the base modules mentioned above but also to the collection of additional software packages that can be installed on top of or alongside Hadoop, such as Apache Pig, Apache Hive, Apache HBase, Apache Spark etc.

MapReduce

Hadoop MapReduce is a software framework for easily writing applications which process big amounts of data in-parallel on large clusters (thousands of nodes) of commodity hardware in a reliable, fault-tolerant manner.

The term MapReduce actually refers to the following two different tasks that Hadoop programs perform:

- The Map Task: This is the first task, which takes input data and converts it into a set of data, where individual elements are broken down into tuples (key/value pairs).
- The Reduce Task: This task takes the output from a map task as input and combines those data tuples into a smaller set of tuples. The reduce task is always performed after the map task.

Typically both the input and the output are stored in a file-system. The framework takes care of scheduling tasks, monitoring them and re-executes the failed tasks.

The MapReduce framework consists of a single master JobTracker and one slave TaskTracker per cluster-node. The master is responsible for resource management, tracking resource consumption/availability and scheduling the jobs component tasks on the slaves, monitoring them and re-executing the failed tasks. The slaves TaskTracker execute the tasks as directed by the master and provide task-status information to the master periodically.

The JobTracker is a single point of failure for the Hadoop MapReduce service which means if JobTracker goes down, all running jobs are halted.

Hadoop Distributed File System

Hadoop can work directly with any mountable distributed file system such as Local FS, HFTP FS, S3 FS, and others, but the most common file system used by Hadoop is the Hadoop Distributed File System (HDFS).

The Hadoop Distributed File System (HDFS) is based on the Google File System (GFS) and provides a distributed file system that is designed to run on large clusters (thousands of computers) of small computer machines in a reliable, fault-tolerant manner.

HDFS uses a master/slave architecture where master consists of a single **NameNode** that manages the file system metadata and one or more slave **DataNodes** that store the actual data.

A file in an HDFS namespace is split into several blocks and those blocks are stored in a set of DataNodes. The NameNode determines the mapping of blocks to the DataNodes. The DataNodes takes care of read and write operation with the file system. They also take care of block creation, deletion and replication based on instruction given by NameNode.

HDFS provides a shell like any other file system and a list of commands are available to interact with the file system. These shell commands will be covered in a separate chapter along with appropriate examples.

How Does Hadoop Work?

- **Stage 1**

A user/application can submit a job to the Hadoop (a hadoop job client) for required process by specifying the following items:

1. The location of the input and output files in the distributed file system.
2. The java classes in the form of jar file containing the implementation of map and reduce functions.
3. The job configuration by setting different parameters specific to the job.

- **Stage 2**

The Hadoop job client then submits the job (jar/executable etc) and configuration to the JobTracker which then assumes the responsibility of distributing the software/configuration to the slaves, scheduling tasks and monitoring them, providing status and diagnostic information to the job-client.

- **Stage 3**

The TaskTrackers on different nodes execute the task as per MapReduce implementation and output of the reduce function is stored into the output files on the file system.

Advantages of Hadoop

- Hadoop framework allows the user to quickly write and test distributed systems. It is efficient, and it automatic distributes the data and work across the machines and in turn, utilizes the underlying parallelism of the CPU cores.
- Hadoop does not rely on hardware to provide fault-tolerance and high availability (FTHA), rather Hadoop library itself has been designed to detect and handle failures at the application layer.
- Servers can be added or removed from the cluster dynamically and Hadoop continues to operate without interruption.
- Another big advantage of Hadoop is that apart from being open source, it is compatible on all the platforms since it is Java based.

Amazon EC2

Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides secure, resizable compute capacity in the cloud. It is designed to make web-scale cloud computing easier for developers.

Amazon EC2's simple web service interface allows you to obtain and configure capacity with minimal friction. It provides you with complete control of your computing resources and lets you run on Amazon's proven computing environment. Amazon EC2 reduces the time required to obtain and boot new server instances to minutes, allowing you to quickly scale capacity, both up and down, as your computing requirements change. Amazon EC2 changes the economics of computing by allowing you to pay only for capacity that you actually use. Amazon EC2 provides developers the tools to build failure resilient applications and isolate them from common failure scenarios.

EC2 Benefits

- **Elastic Web-Scale Computing**

Amazon EC2 enables you to increase or decrease capacity within minutes, not hours or days. You can commission one, hundreds, or even thousands of server instances simultaneously. Because this is all controlled with web service APIs, your application can automatically scale itself up and down depending on its needs.

- **Completely Controlled**

You have complete control of your instances including root access and the ability to interact with them as you would any machine. You can stop any instance while retaining the data on the boot partition, and then subsequently restart the same instance using web service APIs. Instances can be rebooted remotely using web service APIs, and you also have access to their console output.

- **Flexible Cloud Hosting Services**

You have the choice of multiple instance types, operating systems, and software packages. Amazon EC2 allows you to select a configuration of memory, CPU, instance storage, and the boot partition size that is optimal for your choice of operating system and application. For example, choice of operating systems includes numerous Linux distributions and Microsoft Windows Server.

- **Integrated**

Amazon EC2 is integrated with most AWS services such as Amazon Simple Storage Service (Amazon S3), Amazon Relational Database Service (Amazon RDS), and Amazon Virtual Private Cloud (Amazon VPC) to provide a complete, secure solution for computing, query processing, and cloud storage across a wide range of applications.

- **Secure**

Cloud security at AWS is the highest priority. As an AWS customer, you will benefit from a data center and network architecture built to meet the requirements of the most security-sensitive organizations. Amazon EC2 works in conjunction with Amazon VPC to provide security and robust networking functionality for your computing resources.

- **Reliable**

Amazon EC2 offers a highly reliable environment where replacement instances can be rapidly and predictably commissioned. The service runs within Amazon's proven network infrastructure and data centers. The Amazon EC2 Service Level Agreement commitment is 99.95% availability for each Amazon EC2 Region.

Aneka

Aneka is a platform and a framework for developing distributed applications on the Cloud. It harnesses the spare CPU cycles of a heterogeneous network of desktop PCs and servers or datacenters on demand. Aneka provides developers with a rich set of APIs for transparently exploiting such resources and expressing the business logic of applications by using the preferred programming abstractions. System administrators can leverage on a collection of tools to monitor and control the deployed infrastructure. This can be a public cloud available to anyone through the Internet, or a private cloud constituted by a set of nodes with restricted access figure 5.3.

The Aneka based computing cloud is a collection of physical and virtualized resources connected through a network, which are either the Internet or a private intranet. Each of these resources hosts an instance of the Aneka Container representing the runtime environment where the distributed applications are executed. The container provides the basic management features of the single node and leverages all the other operations on the services that it is hosting. The services are broken up into fabric, foundation, and execution services. Fabric services directly interact with the node through the Platform Abstraction Layer (PAL) and perform hardware profiling and dynamic resource provisioning. Foundation services identify the core system of the Aneka middleware, providing a set of basic features to enable Aneka containers to perform specialized and specific sets of tasks. Execution services directly deal with the scheduling and execution of applications in the Cloud.

One of the key features of Aneka is the ability of providing different ways for expressing distributed applications by offering different programming models; execution services are mostly concerned with providing the middleware with an implementation for these models. Additional services such as persistence and security are transversal to the entire stack of services that are hosted by the Container. At the application level, a set of different components and tools are provided to: 1) simplify the development of applications (SDK); 2) porting existing applications to the Cloud; and 3) monitoring and managing the Aneka Cloud.

A common deployment of Aneka is presented at the side. An Aneka based Cloud is constituted by a set of interconnected resources that are dynamically modified according to the user needs by using resource virtualization or by harnessing the spare CPU cycles of desktop machines. If the deployment identifies a private Cloud all the resources are in house, for example within the enterprise. This deployment is extended by adding publicly available resources on demand or by interacting with other Aneka public clouds providing computing resources connected over the Internet.

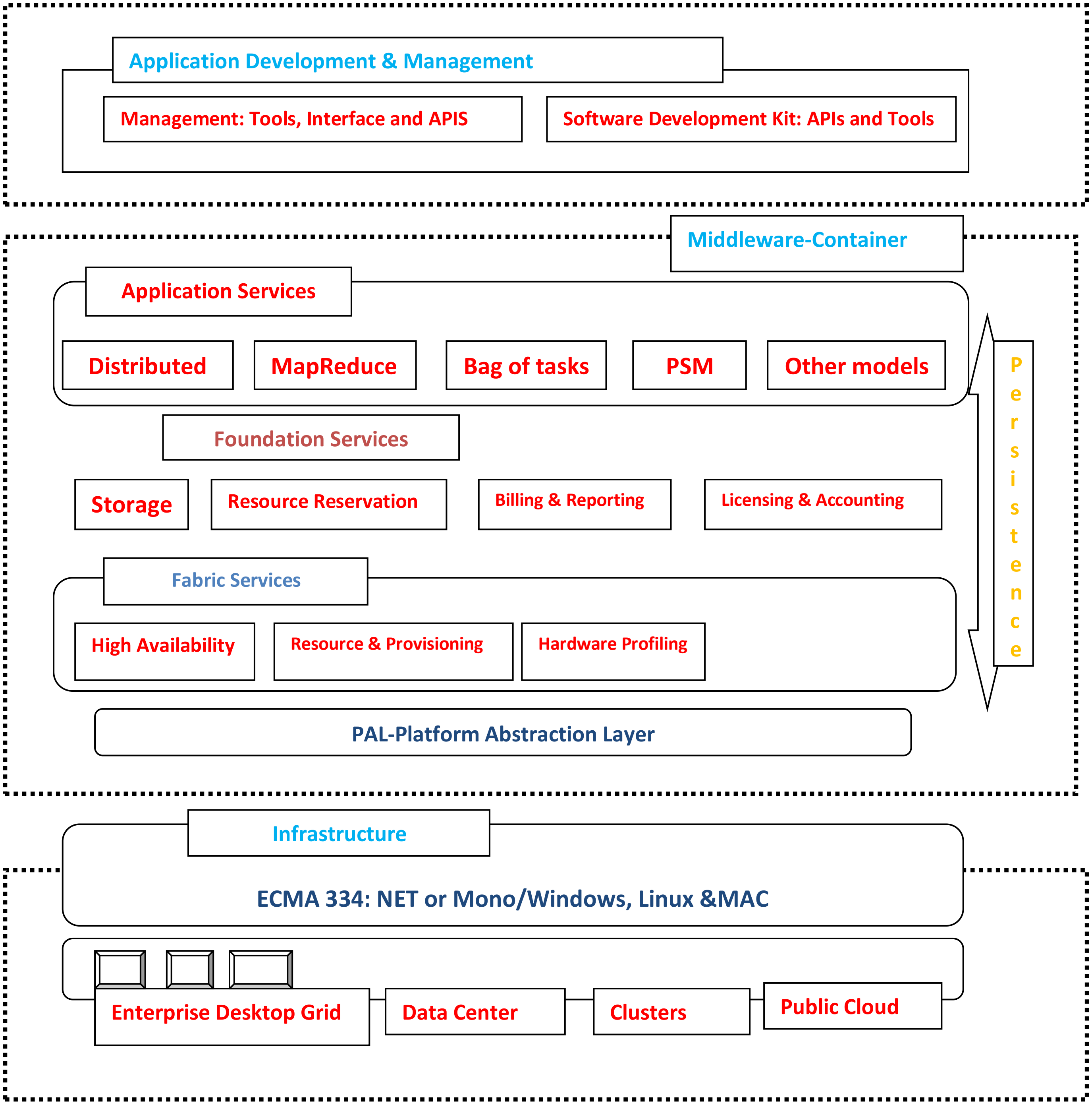

Figure: 5.3 Aneka Model

Web Resources:

- https://aws.amazon.com/security/introduction-to-cloud-security/
- https://www.cloudera.com/
- https://owncloud.org/
- https://www.fortinet.com/solutions/enterprise-midsize-business/cloud-security.html
- https://www.solutionary.com/managed-security-services/cloud-security/
- https://www.tibco.com/blog/2012/12/20/federated-cloud/