## Experiment No.5.

**Aim:** Design and analyze architecture of Aneka and identify different entities to Understand the structure of it.

**Theory:**

Aneka is a software platform for developing cloud computing applications. It allows harnessing of disparate computing resources and managing them into a unique virtual domain

Aneka is a pure PaaS solution for cloud computing.

**Figure provides a complete overview of the components of the Aneka framework.**

-----------------------------------------------------------------------------------------------------

The core infrastructure of the system provides a uniform layer that allows the framework to be deployed over different platforms and operating systems. The physical and virtual resources representing the bare metal of the cloud are managed by the Aneka container, which is installed on each node and constitutes the basic building block of the middleware. A collection of interconnected containers constitute the Aneka Cloud: a single domain in which services are made available to users and developers.

The PAL is responsible for detecting the supported hosting environment and providing the Corresponding implementation to interact with it to support the activity of the container. The PAL pro- vides the following features:
• Uniform and platform-independent implementation interface for accessing the hosting platform
• Uniform access to extended and additional properties of the hosting platform
• Uniform and platform-independent access to remote nodes
• Uniform and platform-independent management interfaces

The PAL is a small layer of software that comprises a detection engine, which automatically configures the container at boot time, with the platform-specific component to access the above information and an implementation of the abstraction layer for the Windows, Linux, and Mac OS X operating systems. The collectible data that are exposed by the PAL are the following:
• Number of cores, frequency, and CPU usage
• Memory size and usage
• Aggregate available disk space
• Network addresses and devices attached to the node

-----------------------------------------------------------------------------------------------------

**Anatomy of the Aneka container**

The Aneka container constitutes the building blocks of Aneka Clouds and represents the runtime machinery available to services and applications. The container, the unit of deployment in Aneka Clouds, is a lightweight software layer designed to host services and interact with the underlying operating system and hardware. The main role of the container is to provide a lightweight environment in which to deploy services and some basic capabilities such as communication

channels through which it interacts with other nodes in the Aneka Cloud. Almost all operations performed within Aneka are carried out by the services managed by the container.

The services installed in the Aneka container can be classified into three major categories:

• Fabric Services

• Foundation Services

• Application Services

The container features three different classes of services: Fabric Services, Foundation Services, and Execution Services. These take care of infrastructure management, supporting services for the Aneka Cloud, and application management and execution, respectively. These services are made available to developers and administrators by means of the application management and development layer, which includes interfaces and APIs for developing cloud applications and the management tools and interfaces for controlling Aneka Clouds.

The services stack resides on top of the Platform Abstraction Layer (PAL), representing the interface to the underlying operating system and hardware. It provides a uniform view of the software and hardware environment in which the container is running. Persistence and security traverse all the services stack to provide a secure and reliable infrastructure. In the following sections we discuss the components of these layers in more detail.

**Fabric services**
Fabric Services define the lowest level of the software stack representing the Aneka Container. They provide access to the resource-provisioning sub system and to the monitoring facilities implemented in Aneka. Resource-provisioning services are in charge of dynamically providing new nodes on demand by relying on virtualization technologies, while monitoring services allow for hardware profiling and implement a basic monitoring infrastructure that can be used by all the services installed in the container.

**Foundation services**

Fabric Services are fundamental services of the Aneka Cloud and define the basic infrastructure management features of the system. Foundation Services are related to the logical management of the distributed system built on top of the infrastructure and provide supporting services for the execution of distributed applications. All the supported programming model scan integrate with and leverage these services to provide advanced and comprehensive application management. These services cover:

• Storage management for applications

• Accounting, billing, and resource pricing

• Resource reservation

**Application services**

Application Services manage the execution of applications and constitute a layer that differentiates according to the specific programming model used for developing distributed applications on top of Aneka. The types and the number of services that compose this layer for each of the programming models may vary according to the specific needs or features of the selected model. It is possible to identify two major types of activities that are common across all the supported models: scheduling and execution. Aneka define s a reference model for implementing the runtime support for programming models that abstracts these two activities in corresponding services: the Scheduling Service and the Execution Service. Moreover, it also defines base implementations that can be extended in order to integrate new models.

-------------------------------------------------------------------------------------------------------------

These services are made available to developers and administrators by means of the application management and development layer, which includes interfaces and APIs for developing cloud applications and the management tools and interfaces for controlling Aneka Clouds.

The framework includes the basic services for infrastructure and node management, application execution, accounting, and system monitoring; existing services can be extended and new features can be added to the cloud by dynamically plugging new ones into the container. Such extensible and flexible infrastructure enables Aneka Clouds to support different programming and execution models for applications. A programming model represents a collection of abstractions that developers can use to express distributed applications; the runtime support for a programming model is constituted by a collection of execution and foundation services interacting together to carry out application execution. Thus, the implementation of a new model requires the development of the specific programming abstractions used by application developers and the services, providing runtime support for them. Programming models are just one aspect of application management and execution. Within an Aneka Cloud environment, there are different aspects involved in providing a scalable and elastic infrastructure and distributed runtime for applications.
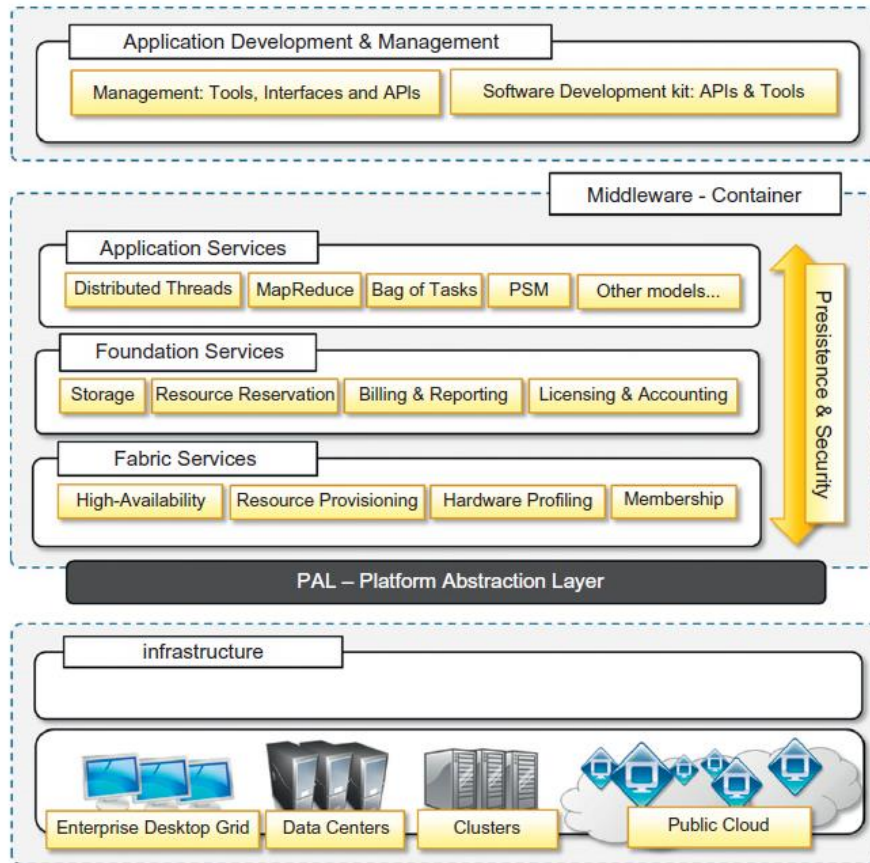
**Figure: Aneka Framework Overview**

The services stack resides on top of the Platform Abstraction Layer (PAL), representing the interface to the underlying operating system and hardware. It provides a uniform view of the software and hardware environment in which the container is running. Persistence and security traverse all the service stack to provide a secure and reliable infrastructure. In the following sections we discuss the components of these layers in more detail.

**Conclusion:** Hence I, have studied and analyzed the architecture of Aneka and identified different entities to understand its structure.