

ANEKA Is a Compatible Platform for Cloud Development Application

Prof.S.Y.Inamdar, Ms. Priyanka M.Kamble, Ms.RugvedaD.Kulkarni

Assistant Professor, Computer Science and Engineering, DACOE, Karad , India.

Student, Computer Science and Engineering, DACOE, Karad, India.

Student, Computer Science and Engineering, DACOE, Karad, India.

Abstract: Aneka is a platform used to develop and manage cloud computing. Aneka is used as PaaS. The feature of the Aneka is used for provisioning resources on public & private cloud. Such as Amazon EC2, Windows Azure etc. Aneka cloud computing use a larger number of resources (computers) to execute their application in distributed & parallel. Aneka is a market oriented cloud development & management platform with rapid application and workload distribution capabilities. Aneka is integrated middleware seamless package which allows you to build & manage interconnect network. It is market oriented; it allows you to creating, scheduling, monitoring & provisioning result using pricing & accounting in private & public cloud. Aneka is freely available with different version. Aneka is a specially designed to work with the popular environment like .net and provides a choice of programming models to ensure that, your application can get all of the benefits that cloud computing.

Keywords: Cloud computing, jobscheduling, scheduling algorithm, cloud services, Service Level Agreement ,task scheduling ,resource utilization.

I. INTRODUCTION

ANEKA is a patented Grid/Cloud computing technology. It has been developed over many years by a number of talented institutes and researchers within the University of Melbourne, led by Rajkumar Buyya[3].

This research is supported by the DIISR (Department of Innovation, industry, Science & Research). It has its roots in open source Gridbus technologies also arising from Dr. Buyya and his team's research since 2002.

ANEKA was created with the aim of providing a set of services that made easy development of cloud construction and applications without sacrificing flexibility, scalability and extensibility [3].

Cloud computing is the way the to built traditional software systems and running a utility-based model for IT infrastructure, platforms, applications, and services. When providers began using virtual private network (VPN) services for the data communications, the term cloud originates from the word of telecommunication [4].

The key features supported by ANEKA are:

- An Aneka container provides pluggable services, persistence solutions, and security and communication protocols.
- Programming models including File –based task model, object oriented thread model.
- Authentication mechanisms such as role based security and Windows domain-based authentication.
- It provides multiple options like RDBMS, SQL Express, MySQL and flat files[3].

Applications Of Aneka:

II. ANEKA ARCHITECTURE:

There are mainly three programming models in ANEKA.

1. Task model

The task model is an application as a collection of task. The task are independent work unit that are managed in any order by the scheduler. The task model included all the components require for its execution on a node.

The task model is the solution for when the distributed application consist of a collection of jobs that are executed on node whose result are submitted and composed together by end user.

In this scenario user creates number of task, it submitted to the Aneka, and wait for the result. The task model will be a specific unit called Aneka task, a task scheduler, a task Executor and Task Manager.

2. Map reduce

MapReduce is a processing technique and a program model based on java for distributed computing. In this model two important tasks, namely Map and Reduce used in algorithms. In this model Map takes a set of input and converts it into another set of data, elements are broken down into tuples (key/value pairs) by this model. The MapReduce library groups together all input values associated with the same key and passes them to the Reduce function. The Reduce function, also written by the user, accepts an intermediate key and generate the set of values for that key. It combines together these values to form a smaller set of values. Typically just zero or one output value is produced by the MapReduce. The average values are supplied to the user's reduce function via an iterator.

3. Thread model

The modern operating system supports the abstraction of process and thread for defining the runtime profile of a software application. A process is nothing but software infrastructure that is used by operating system to control the execution of application. A process generally contains more than one threads. A thread is a sequence of instructions that are executed in parallel with other instructions.

When an application is running that time operating system take care of modifying their execution on local machine. It is a responsibility of the developer to develop a persistent computation as result of thread execution.

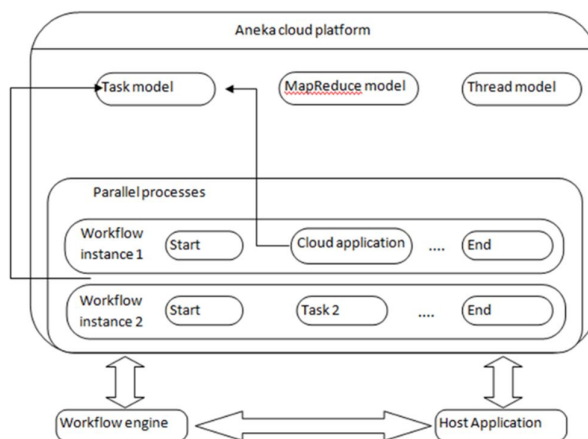


Fig. ANEKA Framework

III. ANEKA CLOUD APPLICATION PLATFORM

Aneka is a platform for establishing resource-intensive and elastic applications as well as their deployment on Clouds. It can trap a vast variety of physical and virtual resources, ranging from desktops, clusters, to virtual Data centres, to provide a single logical “application execution layer”.

The key constituent of the platform are depicted in Aneka Architecture, which provides an overall view of Aneka from its foundations to the applications and the end user services.

The platform is positioned on an extensible Service Oriented Architecture (SOA), which makes the assimilation of new components, incremental development of new visages, and infrastructure deployment and configuration coherent tasks.

1. Middleware.

The platform features a Compatible distributed runtime environment for applications. Such environment is made by aggregating together physical and virtual nodes hosting the Aneka container. The container is lightweight layer that come together with the hosting environment and manages the services deployed on a node. Services constitute the core logic of Aneka Clouds and each container hosts three different classes of services:

• Fabric Services.

Fabric services implement the primitive operations of the infrastructure of the Cloud. These services include: high-availability as well as Failover for improved reliability, node membership and directory, resourceprovisioning, performance monitoring.

• Foundation Services.

Foundation services develop the core functionalities of the Aneka middleware. They prepare a basic set of capabilities that enhance application execution in the Cloud. These services accommodate the infrastructure with added value and are both of use for system administrators and developers.

Within this category we can list: storage management, resource reservation, reporting, accounting, billing, services monitoring, and licensing.

• Application Programming Services.

Application services deal directly with the execution of applications. Applications services are in charge of allow appropriating runtime environment for each application model. At this level Aneka explicates its true potential in supporting number of application models and distributed programming patterns.

Aneka maintains support for the most known application programming patterns such as distributed threads, bag of tasks, and MapReduce.

2. Application Development and Management.

Aneka renditions advanced features for developing and managing applications on the Cloud. The Software Development Kit (SDK) as well as the Management Kit are the two components manifesting such capabilities. They hand over for interacting with the middleware along with managing it with advanced user interfaces and bindings for applications.

With the help of Aneka SDK, developers can instantaneous develop distributed applications, integrate the scaling capabilities of Aneka into existing applications, or implement new services to extend the potential of Aneka. The Management Kit allows deploying, managing, and tuning Aneka-based Clouds. By using a visual approach, it provides means to access and control every aspect of the middleware and also offers advanced features such as application reporting, accounting, billing, user management, and performance monitoring. The SDK and the Management Kit are the tools that enrich the user experience of developers and administrators respectively.

IV. BACKGROUND

Overview of Aneka Cloud Application Development Platform.

Following figure shows the basic architecture of Aneka. The system includes four key components, including Aneka Master, Aneka Worker, Aneka Management Console, and Aneka Client Libraries.

The Aneka Master and Worker are both Aneka Containers which represents the basic deployment unit of Aneka

based Clouds. Aneka Containers host different four kinds of services depending on their role.

The Master is responsible for the Scheduling, Accounting, Reporting, Reservation, Provisioning, and Storage services, and the Workers run execution services [1].

For example, consider the number of workers which are handling by the single master. Master Container divide task between the different workers. The worker is responsible for executing the task and entire task is submitted to the master. The Master Container is responsible for managing the entire Aneka Cloud, coordinating the execution of applications by dispatching the collection of work units to the compute nodes, whilst the Worker Container is take charge of executing the work units, monitoring the execution, and collecting as well as forwarding the results to the master container.

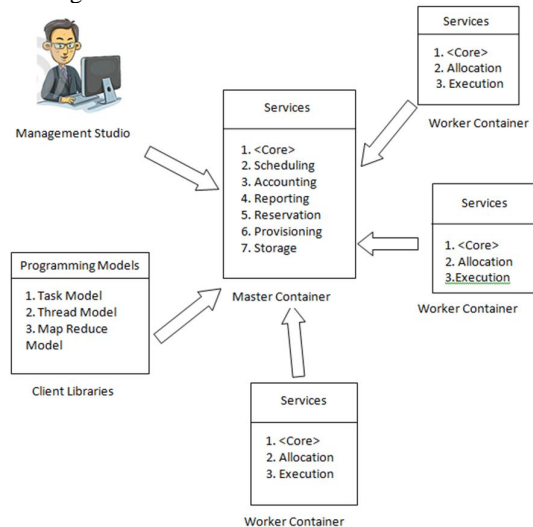


Fig. Basic Architecture of Aneka

V. CONCLUSION AND FUTURE WORK

In this paper, we have introduced the Aneka Cloud Development Platform (Aneka PaaS), presented and discussed the background, design and implementation of the integration of the Aneka.

The Aneka Platform is built on a .NET service oriented architecture allowing seamless integration of public Clouds applications.

Feature of this platform to enhance the development of applications by the software developer allowing to rapidly prototype elastically scalable applications.

There are several applications such as engineering, education; health as well as life sciences and several others have been proven to be appropriate to the Aneka PaaS.

In the first stage, we create the Aneka Worker Container for support of dynamic provisioning of Aneka. In the second step, we deployed the Aneka Master Container on Windows Azure, without requiring any local infrastructure; users can allow Aneka Cloud applications.

The message transferring overhead and message transfer cost will also decrease dramatically.

This is useful to both Service Providers who uses Aneka platform to deliver their services and the final users who expend the services.

VI. REFERENCES:

- [1]Aneka cloud application platform & its integration with windows azure
- Yi Wei1, Karthik Sukumar1, Christian Vecchiola2, Dileban Karunamoorthy2 and Rajkumar Buyya1, 2 Manjrasoft Pty. Ltd., Melbourne, Victoria, Australia
- Cloud Computing and Distributed Systems (CLOUDS) Laboratory, Department of Computer Science and Software Engineering, The University of Melbourne, Australia.
- [2]Manjrasoft Market Oriented Cloud Computing Platform (brochure)
- [3]Manjrasoft Pty Ltd (Aneka Flyer)
- [4]CloudComputing-Concepts,Architecture and Challenge
- [5]Christian Vecchiola, Xingchen Chu, and RajkumarBuyya, Aneka: A Software Platform for .NET-based Cloud Computing, High Speed and Large Scale Scientific Computing, 267-295pp, W. Gentsch, L. Grandinetti, G. Joubert (Eds.), ISBN: 978 -1-60750-073-5, IOS Press, Amsterdam, Netherlands, 2009.
- [6]RajkumarBuyya, Chee Shin Yeo, SrikumarVenugopal, James Broberg, and IvonaBrandic, Cloud Computing and Emerging IT Platforms: Vision, Hype, and Reality for Delivering Computing as the 5th Utility, Future Generation Computer Systems, 25(6):599-616, Elsevier Science, Amsterdam, The Netherlands, June 2009.
- [7]Chao Jin, RajkumarBuyya, MapReduce Programming Model for .NET-Based Cloud Computing, Euro-Par 2009: 417-428.

VII.ACKNOWLEDGMENT

It is our privilege to acknowledge with deep sense of gratitude to our project guide Prof. Inamdar S.Y. and our H.O.D Prof. Patil Ashish N. whose supervision, inspiration and valuable discussion has helped us tremendously to complete our project. Their guidance proved to be the most valuable to overcome all the hurdles in the fulfilment of thisproject on “ANEKA Is a Compatible Platform for Cloud Development Application”.

We are grateful to Principal Dr. Jalindar R. Patil for direct or indirect help in the completion of this project. Last but not least, this acknowledgement would be incomplete without rendering my sincere gratitude to all those who have helped us in the completion of this project.



BIOGRAPHY



Mr. S. Y. Inamdar
Assistant professor of Computer Science &
Engineering, DACOE, Karad.



Ms. Rugveda D. Kulkarni
Student of BE Computer Science &
Engineering, DACOE, Karad.



Ms. Priyanka M. Kamble.
Student of BE Computer Science &
Engineering, DACOE, Karad.