

REVIEWING SOME PLATFORMS IN CLOUD COMPUTING

S.Nagaprasad^{1*} A.VinayaBabu,² K.Madhukar,³ D.Marlene G Verghese,⁴ V.Mallaiah⁵ and A.Sreelatha⁶

¹Research Scholar, Dept. of CSE,
Aacharya Nagarjuna University, Guntur,
Professor, Dept. of CSE,
JNTU College of Engineering, Hyderabad,
Professor, Nizam College, Osmania University, Hyderabad
Associate Professor
Bhimavaram Institute of Engineering & Technology
Pennada, West Godavari.
Research Scholar, Dept. of CSE,
Aacharya Nagarjuna University, Guntur.
Student, Dept. of CSE,
JITS College of Engineering, Karimnagar.

Abstract:

Cloud Computing is an emerging technology in which the research community and industries have recently embarked. However, the infrastructures of most cloud computing systems today are invisible to the research community, or are not explicitly designed to the researchers interested in cloud computing systems. In this paper, we depict the Cloud computing paradigm from a variety of aspects, such as definitions, features, and technologies. Moreover, some Cloud computing Systems are illustrated as well as their architectural trade-offs. It would contribute to better comprehension of the Cloud computing technology so as to enable some communities to design more efficient clouds, and the adoption of this novel computing approach is facilitated in scientific areas.

Keywords: cloud computing, virtualization, distributed paradigm

1. Introduction:

Presently, the Cloud computing has been emerging as a good hot topic since the late of 2007. Industry and academia are starting projects related to Cloud computing. For example, Microsoft has published its Cloud computing system- Windows Azure Platform [1]. Amazon Elastic Compute Cloud [2]; IBM's Blue Cloud [3]; HP, Intel Corporation and Yahoo! Inc. recently announced the creation of a global, multi-data center, open source Cloud computing test bed for industry, research and education [4]. In the last few years, virtualization has introduced some novel system techniques so that the cloud-provider can transparently satisfy its cloud customers' requirements without impacting their own system utilization. Cloud computing differs from grid computing in this regard, it can run in conjunction with the original business workloads. Moreover, novel virtualization technologies, e.g. live-migration and pause-resume give rapid and transparent solutions, interference may not occur between the original systems and the cloud workloads [6]. Consequently,

several business cloud models rapidly evolved to harness the technology by providing computing infrastructure, data-storage, software applications and programming platforms as services. However, the inter-relationships inside cloud computing system have been ambiguous and the enable feasibility of inter-operability has been debatable while referring to the core cloud computing services. Furthermore, every cloud computing service has an interface and employs a different access control protocol. A unified interface for each cloud computing service to support integrated access to cloud computing services is not existent, though portals and gateways can provide unified web-based user interface. So, the introduction of the cloud computing domain, its components and their inner relations are necessary to help the researchers achieve a better understanding of the novel technology. The rest of this paper is structured as follows. The motivation for this study is introduced in section II. Section III addresses the three layers in cloud computing, while section 4 illustrates some cloud computing platforms. We conclude our paper and give the further work in Section 5.

2. MOTIVATION

For the Cloud computing, there are not yet widely accepted definitions in spite of much attention has been given to the Cloud computing. Many researchers and practitioners have attempted to define Cloud computing in various ways. Here are some views from different perspectives.

2.1) Wikinvest View:

Cloud computing is a technology that uses the internet and central remote servers to maintain data and applications. Cloud computing allows consumers and businesses to use applications without installation and access their personal files at any computer with internet access. This technology allows for much more efficient

computing by centralizing storage, memory, processing and bandwidth.

2.2.) A User's View:

After using some services from a cloud platform, a common user may say, "No more servers, no more IT managers, no more licensing. All I need is a browser. I will find the service I need and pay it when I use it based on how much I used it", i.e., Software is as a Service [7].

2.3) A Developer's View:

Accelerate your software application development and test efforts. Eliminate provisioning delays; improve software utilization and lower deployment costs for your applications. In the opinion of a programmer with cloud computing, he concludes, "Programming an application will not be a 30 man/month job with a stupid and horrible project manager.

2.4) A Computer Scientist's View:

Scientists investigate cloud computing into "We are not programming a single machine, rather the World Wide Computer". It means infrastructure is as a Service.

2.5) An economist's view:

Economically, "The most interesting thing about the cloud computing is not the technology, but the new evolving social standards and business models, and the ramifications of egalitarianism on a global scale". Everything he thinks is as a Service. Based on our understanding of the essence of what Clouds are promising to be, we propose that Cloud computing is a usage model in which resources are delivered and, it means to provide resources, such as hardware, software and applications as scalable and "on demand" services via public network in a multi-tenant environment. The providing resource network is called 'Cloud'. All resources in the 'Cloud' are used whenever as utility and scalable infinitely.

3. Three Layers of Cloud Computing:

Cloud Computing is a very broad term used to describe a multitude of aspects in computing, it makes the architecture classification complicated. Architecture can be divided to different service levels according to the level of virtualization. These service levels are normally described as. The most common division is into **Platform-as-a-Service (PaaS)**, **Infrastructure-as-a-Service (IaaS)** and **Software-as-a-Service (SaaS)**. The offered services can be accessed anytime, anywhere in the world over the Internet. Figure 1, shows the layered architecture of Cloud Computing.

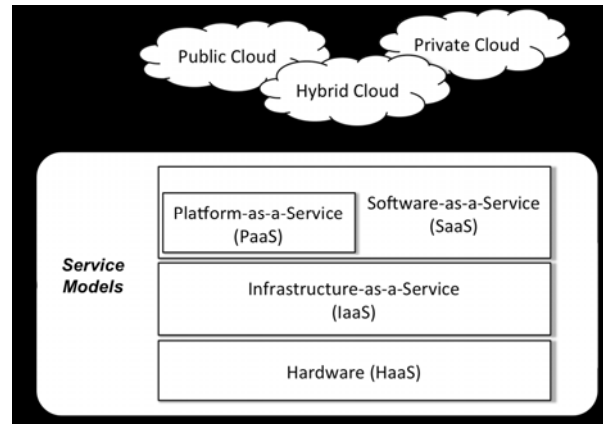


Figure 1: Cloud Architecture.

3.1) Platform-as-a-Service (PaaS):

Platform layer provides another layer of abstraction on top of the virtualized infrastructure. It establishes an integrated software platform over the Web for developing, testing, deploying and hosting of user-friendly web applications for application developers. It eliminates the need for direct contact between supplier and consumer as well as hides the complexity of the underlying hardware layer. Traditionally, a strong vendor lock-in for the applications has been prevalent as each PaaS provider uses their own Application Programming Interfaces (APIs), which are not compatible with each other.

3.2) Infrastructure-as-a-Service (IaaS)

IaaS provides the core of the Cloud service offerings. It sits directly on top of the hardware layer and provides low-level services. Separating Cloud IaaS services from traditional hardware hosting services, is the virtualization layer, which allows dynamic resizing, splitting and assigning of computational resources according to user needs without the user's knowledge. These virtual machines are independent of the underlying hardware and can be moved and copied to allow flexibility and stability. The IaaS layer is available to the customers through underlying networks.

3.3) Software-as-a-Service (SaaS):

SaaS is perhaps the most widely recognized of the Cloud Computing layers. It is normally built on top of either own or foreign IaaS and PaaS environments, utilizing the offered flexibility. SaaS provides an alternative to locally run applications and therefore offers the widest appeal to the end users. The pricing in SaaS is usually based on access basis allowing the users to pay only for what they consume and giving the service providers a steady revenue stream. This differs from the traditional software licensing, where the user has to pay high licensing costs up-front.

In addition to the architectural levels, Cloud Computing services can be classified according to their availability. These modes can be divided into Private, Public and

Hybrid Clouds. As we can see from Figure 1, these are not restricted to any one architectural level [8] [9] [10].

3.3.1) Private Cloud:

Private Clouds are evolution from internal datacenters. They are managed within the organization and are not available to the general public. This offers several benefits to the organization, including better control over security, hardware restrictions and legal requirements making them ideal for mission critical systems. However, they might not offer the same scalability and flexibility benefits as Public Clouds do, but still have cost saving benefits. As the definitions are still somewhat unambiguous, some disagreement is evident in the literature on whether Private Clouds should be included in the Cloud Computing definition or not [12]. Virtual Private Clouds can be used to describe the grey zone between Private and External Clouds provided by third-party providers, but with datacenter sharing restricted to the parts of a single organization [11].

3.3.2) Public Cloud:

As opposed to Private Clouds, Public Clouds are available to the public using the utility model. They offer Cloud Computing services in the mainstream sense. The services are offered by a third-party provider over the Internet on pay-as-you-go, on-demand basis. However, they face many concerns regarding security, privacy, integrity, lock-in and availability that are associated with giving sensitive information to a third-party.

3.3.3) Hybrid Cloud:

The reality is that neither Private nor Public Clouds on their own can fully support the requirements of enterprise IT services. Therefore, most enterprise IT environments are forms of Hybrid Clouds, consisting of internal and external providers. Mission critical systems can be kept in control in internal datacenters while scalability and elasticity issues can be addressed by Public Clouds.

4. Some Cloud Computing Platforms:

Industry analysts and Software Developers have made confident projections on how Cloud computing will transform the entire computing industry.

4.1) Microsoft's Windows Azure platform:

The Windows Azure platform is a flexible cloud computing platform that lets you focus on solving business problems and addressing customer needs. This platform can be used both by applications running in the cloud and by applications running on local systems. It contains the following components and is shown in Figure 2.

Windows Azure:

A Windows based environment for running applications in Microsoft data centers.

SQL Azure:

SQL Server based data services.

.NET Services:

Distributed infrastructure services to cloud-based and local applications.

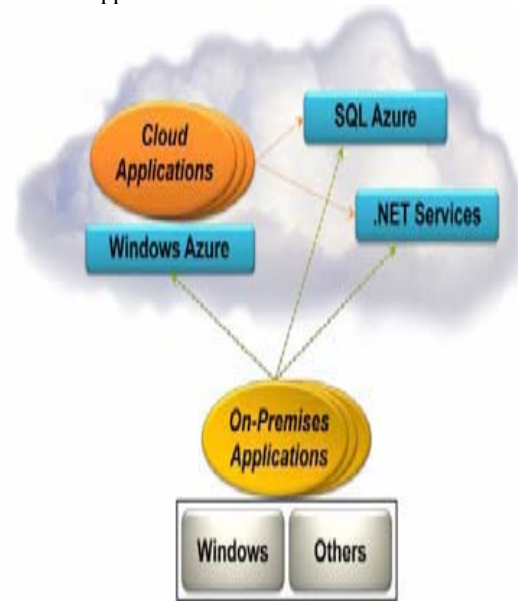


Figure 2: The Windows Azure platform.

4.2) Windows Azure:

Windows Azure is a cloud platform offering that "provides a wide range of Internet services that can be consumed from both on-premises environments and the Internet. Azure Services Platform is an application platform in the cloud that allows applications to be hosted and run at Microsoft datacenters. It provides a *cloud operating system* called **Windows Azure** that serves as a runtime for the applications and provides a set of services that allows development, management and hosting of applications off-premises.

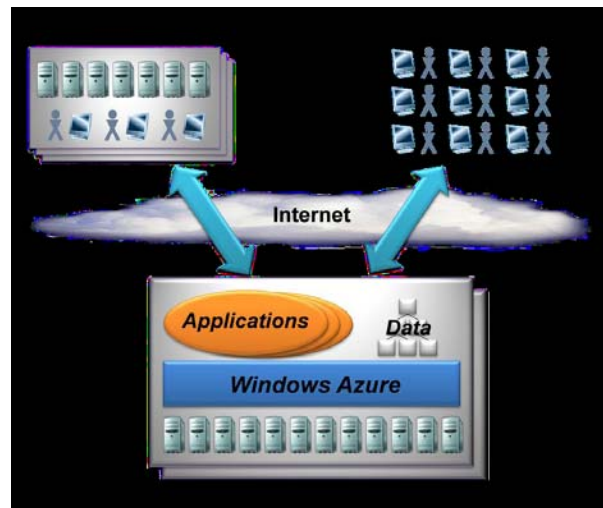


Figure 3: Windows Azure applications run in Microsoft data centers and are accessed via the Internet.

As the figure 3 shows, Windows Azure runs on machines in Microsoft data centers. Rather than providing

software that Microsoft customers can install and run themselves on their own computers, Windows Azure is a service, Customers use it to run applications and store data on Internet-accessible machines owned by Microsoft. Those applications might provide services to businesses, to consumers, or both. Here are some examples of the kinds of applications that might be built on Windows Azure.

4.3) SQL Azure:

Microsoft SQL Azure Database is a secure relational database service based on proven SQL Server technologies. The difference is that it is a fully managed cloud database, offered as a service, running in Microsoft datacenters around the globe. It is highly scalable, with built in high-availability and fault tolerance, giving you the ability to start small or serve a global customer base immediately.

- SQL Azure Database: Unlike the Windows Azure storage service, SQL Azure Database is built on Microsoft SQL Server.
- “Huron” Data Sync: “Huron” Data Sync. Technology synchronizes relational data across various on-premises DBMSs. The owners of that data can determine what should be synchronized, how conflicts should be handled and more.

4.4) .NET Services:

.NET Services provides cloud-based infrastructure services that can be used by either on-premises applications or cloud applications. It includes Access Control and Service Bus. Service Bus provides a mechanism to manage Web services exposed by applications.

5) Amazon Web Services for Cloud Competing:

The Amazon Web Services (AWS) are a collection of remote computing services also called web services that together make up a cloud computing platform, offered over the Internet. Amazon Web Services for Cloud Competing contains the following components.

5.1) Amazon Elastic Compute Cloud (EC2)

Amazon Elastic Compute Cloud (Amazon EC2) shown in Figure 4 is a web service that enables customers to launch and manage Linux/UNIX/Windows server instances in the data centers of Amazon.

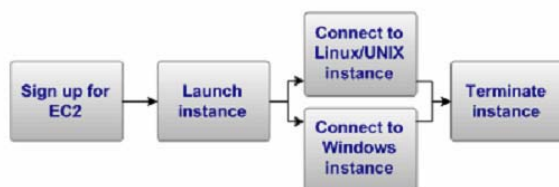


Figure 4: Amazon Elastic Compute Cloud (Amazon EC2).

Amazon EC2 provides the following major features:

- a) Resource delivered as AMI (Amazon Machine Image).
- b) Compute instance.
- c) Explicit access control

5.2) Amazon Simple Storage Service (S3):

Amazon S3 is storage for the Internet. It is designed to make web-scale computing easier for developers. Amazon S3 provides a simple web services interface that can be used to store and retrieve any amount of data, at any time, from anywhere on the web. It gives any developer access to the same highly scalable, reliable, secure, fast, inexpensive infrastructure that Amazon uses to run its own global network of web sites. The service aims to maximize benefits of scale and to pass those benefits on to developers. The features of Amazon S3 are listed as follows.

- Write, read, and delete objects containing from 1 byte to 5 gigabytes of data each. The number of objects you can store is unlimited.
- Each object is stored in a bucket and retrieved via a unique, developer-assigned key.
- Objects stored in a Region never leave the Region unless you transfer them out. For example, objects stored in the EU (Ireland) Region never leave the EU.
- Authentication mechanisms are provided to ensure that data is kept secure from unauthorized access. Objects can be made private or public, and rights can be granted to specific users.
- Uses standards-based REST and SOAP interfaces designed to work with any Internet-development toolkit.
- Built to be flexible so that protocol or functional layers can easily be added. The default download protocol is HTTP. A BitTorrent protocol interface is provided to lower costs for high-scale distribution.

5.3) Amazon Simple Queue Service (SQS):

Amazon SQS is intended to give you a high-level overview of this web service. A queue is a temporary repository for messages that are awaiting processing. Amazon SQS is a distributed queue system that enables web service applications to quickly and reliably queue messages that one component in the application generates to be consumed by another component. Using Amazon SQS, you can decouple the components of an application so they run independently, with Amazon SQS easing message management between components. Any component of a distributed application can store any type of data in a fail-safe queue. Any other component can then later receive the data programmatically using the SQS API.

SQS ensures delivery of each message at least once, and supports multiple readers and writers interacting with the same queue. A single queue can be used simultaneously by many distributed application components, with no need for those components to coordinate with each other to share the queue.

Amazon SQS provides the following major features:

- **Redundant infrastructure:** Guarantees delivery of your messages at least once, highly concurrent access to messages, and high availability for sending and retrieving messages
- **Multiple writers and readers:** Multiple parts of your system can send or receive messages at the same time SQS locks the message during processing, keeping other parts of your system from processing the message simultaneously.
- **Configurable settings per queue:** All of your queues don't have to be exactly alike. For example; one queue can be optimized for messages that require a longer processing time than others.
- **Variable message size:** Your messages can be up to 8 KB in size for even larger messages; you can store the contents of the message using the Amazon Simple Storage Service (Amazon S3) or Amazon SimpleDB and use Amazon SQS to hold a pointer to the Amazon S3 or Amazon SDB object. Alternately, you can split the larger message into smaller ones.
For more information about the services, go to the Amazon S3 detail page and the Amazon SimpleDB detail page.
- **Unlimited queues and messages:** You can have as many queues and messages in the Amazon SQS system as you want.

5.4) Amazon SimpleDB (SDB):

Amazon SimpleDB is a highly available, scalable, and flexible non-relational data store that offloads the work of database administration. Developers simply store and query data items via web services requests, and Amazon SimpleDB does the rest.

Unbound by the strict requirements of a relational database, Amazon SimpleDB is optimized to provide high availability, ease of scalability, and flexibility with little or no administrative burden. Behind the scenes, Amazon SimpleDB creates and manages multiple geographically distributed replicas of your data automatically to enable high availability and data durability. The service responds to changes in traffic by charging you only for the compute and storage resources actually consumed in serving your requests. You can change your data model on the fly, and data is automatically indexed for you. With Amazon SimpleDB, you can focus on application development without worrying about infrastructure provisioning, high availability, software maintenance, schema and index management, or performance tuning.

5.5) Amazon Flexible Payment Service:

Amazon Flexible Payments Service is the first payments service designed from the ground up for developers. It is built on top of Amazon's reliable and scalable payments infrastructure and provides developers with a convenient way to charge Amazon's tens of

millions of customers (with their permission, of course!). Amazon customers can pay using the same login credentials, shipping address and payment information they already have on file with Amazon.

With Amazon FPS, developers can accept payments on their website for selling goods or services, raise donations, execute recurring payments, and send payments.

Amazon FPS service is extensive so that its functionality is divided into five Quick Start implementations below.

- a) Amazon FPS Basic Quick Start
- b) Amazon FPS Marketplace Quick Start
- c) Amazon FPS Advanced Quick Start
- d) Amazon FPS Aggregated Payments Quick Start
- e) Amazon FPS Account Management Quick Start

5.6) Amazon DevPay

You may get paid for products that you build on either Amazon EC2 or the Amazon S3 through Amazon DevPay which is a bill and account management service. Amazon DevPay designs the order pipeline, billing and account system for you.

6) Google Cloud:

6.1) Google Engine

Other than supporting the Python standard library, Google App Engine also supports Application Programming Interfaces (APIs) for the datastore, Google Accounts, URL fetch, image manipulation, and email services [13].

6.2) Google Apps:

Google Apps is one of the most sophisticated and comprehensive collaborative products available. The program includes applications for email, calendars, instant messaging, room reservations, document storage and editing and video sharing.

6.3) Google File system (GFS):

GFS provides a reliable distributed storage system that can grow to petabyte scale by keeping data in 64 megabyte "chunks" stored on disks spread across thousands of machines. Each chunk is replicated, usually 3 times, on different machines so GFS can recover seamlessly from disk or machine failure. Figure 5 shows its Architecture.

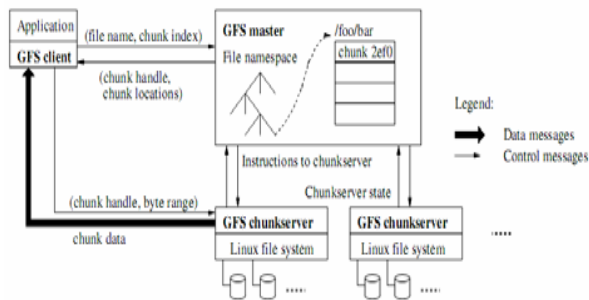


Figure 5: Google File System Architecture.

6.4) Work queue and MapReduce:

Work queue provides a handling mechanism for scheduling a job to run on a cluster of machines. It schedules jobs, allocates resources, reports status, and collects the results. Its execution flow is demonstrated in Figure 6.

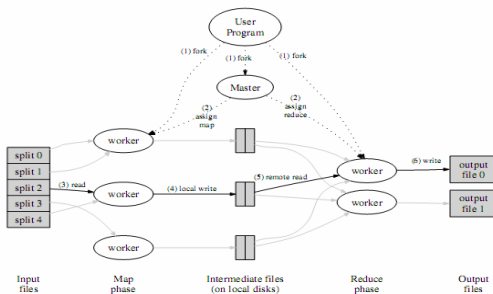


Figure 6: MapReduce Execution overview.

6.5) Eucalyptus:

Eucalyptus is a software platform for the implementation of private cloud computing on computer clusters. There are enterprise edition and open-source edition. Currently, it exports a user-facing interface that is compatible with the Amazon EC2 and S3 services but the platform is modularized so that it can support a set of different interfaces simultaneously. The development of Eucalyptus software is sponsored by Eucalyptus Systems, a venture backed start-up. Eucalyptus was designed modularly and its components were a set of web services interoperating using standard communication protocols. A description of the components within the Eucalyptus is briefly as follows.

- 1) Cloud Controller (CLC)
- 2) Cluster Controller (CC)
- 3) Node Controller (NC)
- 4) Storage Controller (SC)

CONCLUSION:

In this paper, we have proposed Cloud computing paradigm from a variety of aspects, such as definitions, features, and technologies. Moreover, we have illustrated several representative platforms for the state-of-the-art Cloud computing. Cloud computing needs to be extended to support negotiation of QoS based on Service Level Agreement (SLAs). The corresponding algorithms should be designed for allocation of VM resources to meet

SLAs between service providers and end-users. The risks of the violation of SLAs must be managed effectively. Furthermore, we must extend some protocols to support interaction between different Cloud service providers.

REFERENCES:

- [1] Windows Azure Platform, v1.3—Chappell. <http://www.microsoft.com/windowsazure/>.
- [2] Amazon Elastic Compute Cloud [URL].<http://aws.amazon.com/ec2/>
- [3] IBM Blue Cloud project. <http://www-03.ibm.com/press/us/en/pressrelease/22613.wss/>.
- [4] Global Cloud computing test bed. http://www.hp.com/hpinfo/newsroom/press/2008/080729x_a.html/.
- [5]http://en.wikipedia.org/wiki/Cloud_computing#cite_note-0
- [6] Lamia Youseff, Maria Butrico, Dilma Da Silva. Toward a Unified Ontology of Cloud Computing. GCE'08, 2008.
- [7]Yike Guo. Introduction to Cloud Computing. TR. 11, 2009.
- [8] Armbrust, M., Fox, A., Griffith, R., Joseph, A., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., and Stoica, I. (2009). Above the clouds: A Berkeley view of cloud computing. *EECS Department, University of California, Berkeley, Tech. Rep. UCB/EECS-2009-28*.
- [9] Reynolds, E. and Bess, C. (2009). Clearing Up the Cloud: Adoption Strategies for Cloud Computing. *Cutter IT Journal*, 22(6/7):14–20.
- [10] Rimal, B., Choi, E., and Lumb, I. (2009). A Taxonomy and Survey of Cloud Computing Systems. In *Fifth International Joint Conference on INC, IMS and IDC*, pages 44–51. IEEE.
- [11] Amazon (2010). Amazon Virtual Private Cloud (Amazon VPC). Available from <http://aws.amazon.com/vpc/>. Accessed in April 20th 2010.
- [12] Armbrust, M., Fox, A., Griffith, R., Joseph, A., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., and Stoica, I. (2009). Above the clouds: A Berkeley view of cloud computing. *EECS Department, University of California, Berkeley, Tech. Rep. UCB/EECS-2009-28*.
- [13] Google Apps. <http://www.google.com/apps/business/index>.

S.Nagaprasad Received the Bachelor of Computer Applications (B.C.A.) From Osmania University,

Hyderabad, in 2001, M.Sc (I.T.) From Sikkim Manipal University, Manipal in 2003 and M.Tech (C.S.) From Allahabad Agricultural University, Allahabad in 2005. Presently Pursing Ph.D in Computer Science & Engineering from Acharya Nagarjuna University, Guntur.



Prof.A.Vinay Babu obtained his Bachelors degree in Electronics & Communication Engineering from Osmania

University. Has dual Masters Degree, one in Computer Science & Engg and the other one is in ECE from JNTU. He obtained his PhD from JNTU, Hyderabad. His research area is Data Mining and Image processing.



Prof. K.Madhukar received his Masters in Engineering Physics and Instrumentation from Osmania University in

1980 and PhD from Osmania University in 1989. Presently he is Professor and Head, Dept of Physics and Electronics, Nizam College, Osmania University, Hyderabad. He worked as ICT4D programme specialist in United Nations Development Programme (UNDP) for more than four years. His research area is Data mining, Knowledge Discovery and Networks. Presently eight students are working for their doctoral degree under his supervision.

