Migration Management in Cloud Computing

K C Gouda[#1], Dines Dwivedi[*2], Anurag Patro [*3], Nagaraj Bhat[#4]

[#]*CSIR Centre for Mathematical Modeling and Computer Simulation (C-MMACS)*
Wind Tunnel Road, Bangalore-37, India
[*]*Department of Computer Science and Engineering, National Institute of Technology*
Rourkela, Odisha, India, 769008

Abstract **Cloud computing has been rapidly and radically changing the dynamics of IT consumption. The core cloud computing proposition is being able to buy compute technology as a service in an on-demand, elastic, and pay-as-you-go model. Public cloud offerings to the IT industry are very prominent. But the organizations, which have already invested money in their own infrastructure, are interested in setting up the private clouds within their organizational boundaries to avail the benefits of cloud computing technologies. In the present study a clear concept of the migration mechanism is presented which can be easily implemented in the cloud computing environment for the efficient, robust and cost effective management.**

***Keywords*— Cloud computing, Migration, cost effective, Cloud Management**

I. INTRODUCTION

Cloud computing is just a metaphor for internet. It is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, storage, servers applications, and services) which can be rapidly provisioned and released with minimal management effort or service provider interaction. It is basically virtualizing computing hardware and software by providing services so that the user is at a greater liberty at the choice of computing and storage resources.

In May 1997, NetCentric made an attempt to trademark "cloud computing" but abandoned it in April 1999 (Patent sl.no. 75291765). One of the foremost milestones in cloud computing evolution pathway was the arrival of Salesforce.com in 1999, which lead the way of delivering enterprise applications via a simple website. This services firm paved the way for specialist as well as mainstream software firms to provide applications over the internet.

In April 2001, the New York Times published an article by John Markoff regarding Dave Winer's negative reaction to Microsoft's then new '.net' services platform known as Hailstorm which used the phrase "'cloud' of computers".

In August 2006, Eric Schmidt of Google described their approach to SaaS of cloud computing at a search engine conference. This was the first famous public usage of the term, where not only "cloud" but also "cloud computing" was used to denote to SaaS and since it was in the perspective of Google, the term added PaaS/IaaS connotations related with the Google way of management of data centers and infrastructure.

The next big development was of Amazon Web Services in 2002, which provided a cloud-based services suite including storage, computation and even human intelligence through the 'Amazon Mechanical Turk'. Again in 2006, Amazon launched its Elastic Compute cloud (EC2) as a commercial web service that allowed small companies and individuals to rent computers to run their own computer applications. Amazon EC2/S3 was the first extensively accessible cloud computing infrastructure service (IaaS) which provided its SaaS online video platform to UK's TV stations and newspapers.

Another big turnaround came in 2009, as Web 2.0 hit its heights, and Google and others started to offer browser-based enterprise applications, though various services such as Google Apps. Very important contribution to cloud computing has been the advent of 'killer apps' from leading technology goliaths such as Microsoft and Google. When these companies started to deliver services in a way that was dependable and consumer friendly, the tech industry as a whole had a greater general acceptance of many online services.

II. COMPONENTS OF CLOUD

The main components of cloud are the distributed servers i.e. data centers, the clients and the databases.

A. Clients

Clients can be defined as the devices that the end users use to interact with the cloud when they require the cloud services. They can be personal computers, laptops, smart phones, tablets etc. They are divided into two types:-

1)Thin Clients: These are the computers that do not contain internal hard drives and just display the information received from the server directly. It has more security as the computation is done in the server and not in the client machine.

2)Thick Clients: These are just normal computation devices that connect to the cloud using web browsers without engaging a hard drive.

B. Data Center

It is an assembly of servers where the client subscribed application is stored. It can be placed anywhere in the world and can be accessed using the internet. A better approach is to

use virtual servers through a single physical server. Softwares can be used that allow multiple instances of virtual servers to run whenever the physical server is accessed.

C. Databases

These are the components in cloud computing where the data is stored and accessed accordingly.

III. LAYERS IN CLOUD COMPUTING

A. Cloud Application Layer

It is the most visible layer to the end-users of the cloud. Usually, the users access the services provided by this layer through web-browsers, and are sometimes needed to pay to use them. This layer is also referred to as Software as a Service (SaaS). Examples are Google Docs-Spreadsheet and Google Docs-Presentation etc.

B. Cloud Software Environment Layer

The second layer is the cloud software environment/ platform layer. The users of this layer are generally cloud application developers, implementing their applications for and deploying them on the cloud. The service provided in this layer is Platform as a Service (PaaS). E.g., Google's App Engine, which provides a python runtime environment and APIs for applications to interact with Google's cloud runtime environment.

C. Cloud Infrastructure Layer

The cloud software infrastructure layer makes available fundamental resources to the other higher layers. Cloud services provided in this layer can be categorized into: data storage, computational resources and communications.

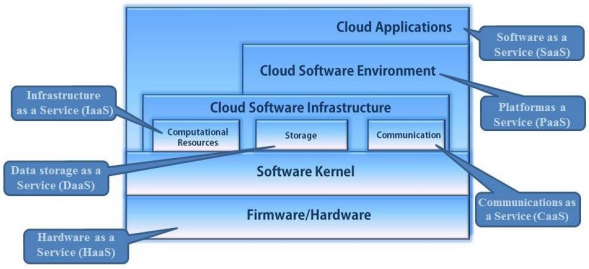

Fig.1 Schematic of cloud computing Environment

D. Software Kernel

This cloud layer consists of the basic software management for the various physical servers that make up the cloud. Software kernels at this level can be implemented as an hypervisor, OS kernel, and virtual machine monitor and/or clustering middleware.

E. Firmware and Hardware

The bottom most layer of the cloud structure is the actual physical hardware and switches that form the base of the cloud. Users of this layer of the cloud are generally big enterprises with huge IT requirements in need of Hardware as a service (HaaS). Figure 1 represents the schematic of the cloud computing environment with the associated components.

IV. MIGRATION IN CLOUD COMPUTING

The process of transitioning all/part of a company's data/applications/services from on-site premises to the cloud or moving them from one cloud environment to another, where the information can be accessed over the Internet is called cloud migration. The process of transition to a different cloud service provider is known as cloud service migration. Successful migration to a service provider's environment requires the use of various middlewares, such as a cloud integration tool, to link any gaps between the vendor's and the customer's technologies.

Cloud Migration is broadly classified into two categories: Big Bang Migration and Trickle Migration

A. *Big bang migration*

Big bang migrations involve completing the entire migration at a time in a very small, defined processing window. This involves system downtime while the data is extracted from the source system, processed, and transferred to the destination, followed by the transfer of processing over to the new environment. Although it completes the migration in the shortest-possible time, it carries several risks. Very Few organizations can work with a core system being unavailable for long periods, so there is extreme pressure on the migration.

B. Trickle Migration

In Trickle migrations an incremental approach is taken to migrate data to cloud. Rather than aiming to complete the whole event in a stipulated time window, trickle migration involves running the old and new systems in parallel and migrating the data in phases. This method innately provides the zero downtime that critical applications requiring full time operation need. Trickle migration can be implemented with other real-time processes to move data, and these processes can also be used to maintain the data by passing future changes to the target system. Adopting the trickle approach does add some complexity to the design, because it must be possible to track which data has been migrated. If this is part of a system's migration, it may also mean that source and destination systems are operating in parallel, with end-users having to switch between both, depending on where the required information is currently located. On the other hand, the old system(s) can continue to be operational until the entire migration is completed, before users are transferred to the new system. In such a case, any changes to data in the source system(s) must trigger remigration of the appropriate records so the target is updated correctly.

Following are some of the motivations for the company's decision of migration to the cloud:

A. Standardization

Standardization means simplifying the system by dealing with less number of configurations, easily facilitated automation and much simpler support. Along with it, the cloud environments being very flexible allows easy provision in various ways. Also it is very user friendly.

B. Self Service

With self-service, the user has the control and more cost and usage choices, along with increased visibility.

C. Virtualization

Virtualization ensures flexibility, increasing the utilization

thus being energy efficient. Infrastructure Abstraction and Soft Configuration are characteristics of virtualization.

D. Automation

Automation mean low human involvement and swift deployment. It also ensures repeatable configuration thus improving compliance.

E. Cost Savings

Using cloud, capital expenditure can be decreased by not having to buy and maintain costly hardware. A cloud service provider can deploy the data to their high performance systems, with no need to maintain and upgrade expensive softwares and systems, instead the employees can be used to do some productive work for the organization.

F. Better Collaboration

Good collaboration is the new business success *mantra* and migration to the cloud makes it much easier to achieve. A more mobile workforce can be achieved who using their own devices can be more productive.

G. Improved Network Performance

If organisations are using remote data-centres of their cloud service providers to work on their data, the workload on their on-premise networks can be greatly reduced, thus improving performance of functions using the on-premise internal network.

H. Improved Integration and Compatibility

The upcoming big data needs of organisations needs them to be capable of accessing and analysing data stored across various platforms. Consequently, this requires close integration and compatibility between them. With cloud migration the focus shifts to extraction of data rather than operating systems and thus abolishing the problems of compatibility and integration.

I. Reliability and Security

While many argue that data security is the main obstacle with cloud computing, there are arguments that prove otherwise. Reliable cloud service providers can take off the security concerns of the company data and applications to much extent. With data redundancy the chance of data loss or decay is minimized due to accidents or any other reasons.

IV. CLOUD MIGRATION PATTERNS

A. Re-Host (P1)

An application (component) is re-hosted as-is on cloud platform(s). Re-Host on cloud environments make use of elastic resources, multiple cloud deployment for failover and scalability. It improves backup and failover, coarse-grained scalability at application level and simplifies coarse grained re-deployment. Some risks associated with this is integration which can cause complexity as existing architecture contains portability, deployment time and cost and scalability.

B. Cloudification (P2)

An application (component) is hosted on-premise as-is but use public cloud services for extending capabilities instead of on-premise components. With this pattern there is improved time to market but integration may still produce complexity.

C. Relocation and Optimization (P3)

A component re-hosted (or relocated) on a cloud platform is optimized but without evolution of the application architecture. Some benefits are component re-hosting in cloud and optimized performance. The problem with this pattern is that type of application requests change over time. Cloud provider does not provide the necessary services to wrap up the optimizations around the applications without re-architecting.

D. Multi-Cloud Relocation (P4)

A component re-hosted(or relocated) on a cloud platform is enhanced by using the environmental services of the other cloud platforms. Its benefits are components re-hosting in multiple cloud platforms, improve availability and vendor lock-in. The risks associated with this are that the Cloud provider does not provide the necessary services to run in multiple cloud platforms without re-architecting or rewriting the code.

E. Multi Cloud Refactor (P5)

An on premise application is re-architected for deployment on cloud platform to provide better Quality of Service. It gives optimal scalability and performance, wider range of multi cloud deployment options and it has got agility to respond to business and IT changes in future. As the on premise application is modernized in isolation it does not ensure consistent architecture.

F. Hybrid Refactor (P6) with on Premise Adaptation (P7)

A re-architected application is deployed partially on cloud environment and partially to its on premise platform. This way the sensitive data remains isolated and in house. But integration is dependent on a particular cloud platform and may cause vendor lock-in.

G. Hybrid Refactor with Cloud Adaptation (P8)

An interface is implemented to provide loose coupled access to components re-hosted on cloud platform. It ensure loose coupling which is more effective and its platform independent in operation. But there may be lack of suitable API on legacy application. Existing application may not have suitable API and may not comply with message based interaction in cloud.

- Hybrid refactor with hybrid adaptations(P9)

H. Multi Cloud Rebinding (P10)

A re-architected application is deployed partially on multiple cloud environments and enables the application to continue to function using secondary deployment when the primary platform fails. Although there will be downtimes as the faulty services become okay again, traffic can be delivered, returning system performance to maximum levels.

- Multi cloud rebinding with cloud brokerage(P11)

I. Replacement (P12)

Individual capabilities of re-architected solution are re-provisioned rather than being re-engineered. This results in improving the solution through best in-class cloud services while saving re-engineering costs. But then the cloud services

use specific protocol which may pose as a problem during replacement. The different replacements are Replacement with on premise adaptation (P13), Replacement with cloud adaptation (P14) and Multi-application modernization (P15).

On premise applications are re-architected as a portfolio and deployed on the cloud environment. It ensures consistent information in shared components as well as the rules which in turn reduces the maintenance cost of the shared components. The cloud migration pattern analysis are summarized in table 1 below.

Table 1: Cloud Migration Patterns

	Rehost	Cloudification	Relocation		Refactor					Rebinding		Replacement			Modernization
Aim	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15
Marketting time	✓	-	x	x	x	-	-	-	-	x	x	✓	✓	✓	✓
New Capabilities	x	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	-	-	-	-
Reduce Operational costs	✓	✓	-	-	x	-	-	-	-	x	x	✓	✓	✓	✓
Leverage Investments	✓	✓	-	-	-	✓	✓	✓	✓	✓	✓	x	x	x	✓
Free up on premise resources	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Scalability	x	-	-	-	✓	✓	✓	✓	✓	✓	✓	-	-	-	-
Operational efficiency	✓	-	-	-	✓	-	-	-	-	✓	✓	✓	✓	✓	-

V. MIGRATION TO CLOUD- STEP WISE APPROACH

A. Process Review

The Cloud migration process is mainly based on the guided identification and analysis of relevant factors that might influence the cloud selection and migration task. Those factors are usually related to the organization, the target legacy application, and the cloud provider. Therefore, identifying those characteristics is the initial approach towards determining whether and if yes then how it would be possible to migrate components to the cloud. To help developers in characterizing their organization, legacy application and candidate cloud providers, the process relies on the creation of entity profiles based on a set of provided profile templates. As the number of defined profiles increases, which are stored and made easily accessible using automated tools, it becomes easier for the developer to find profile templates whose characteristics resemble those of the entities involved in the migration. Once the required entity profiles have been created, the next step is to cross-analyze them in order to identify potential migration risks and constraints, as well as to find ways to avoid or tackle them. In this, analysis is done iteratively, starting from a more organizational context to a more technical one. A detailed explanation of this process follows.

B. Process Workflow

The process can be divided into nine sub-processes, starting from the characterization of the organization profile to final step of migration of the target component to the selected cloud service provider. These nine sub processes should be carried out according to the flowchart shown in Figure 2.

The granular details of these processes are explained further.

1) Produce Organization Profile: The goal of this sub process is to create a profile for the organization. The organization profile should contain information regarding legal or administrative characteristics that can be considered relevant to the migration task like policies, guidelines, laws that the organization must abide to. The plan is to detect potential organizational constraints that might affect the cloud migration decision, before carrying out any further analysis.

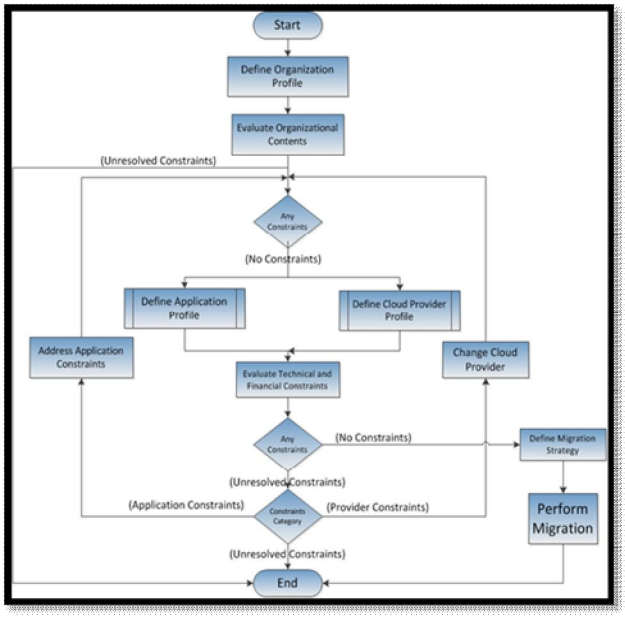

Fig. 2. Flow chart of Migration in Cloud Environemnet

The following points can be used to define the organization profile:

- The motivation that led the organization to take up cloud migration
- Benefits of the organization from the migration.
- The organization's procedure of acquiring and allocating its computing resources?
- The organization place to develop, test and deploy its software products and services?
- Any law or legal restrictions on the physical location of its components?

2) Assess Organizational Constraints: In this sub-process, the developer should conduct a primary evaluation of plausible organizational constraints regarding the deployment to a particular cloud model, based on information in the organization profile. The aim is to detect critical factors within the organization that may pose obstacle for migrating to the cloud. The following instances can be used for identifying potential organizational constraints:

- Resistance by employees of the organization to changes affecting them resulting from the migration.
- Inconsistency or disagreement of policies followed by organization and cloud provider;
- Losing control over existing IT resources;
- Dependence on legacy applications and/or data that cannot be accessed from outside the organization;
- Risk of an unauthorized third party accessing critical business data that is kept in the cloud;

• Legal restrictions on the physical location of critical IT resources

If any critical constraint is found at the organizational level that hampers cloud migration, the process is terminated prematurely. This prevents the developer from performing any subsequent analysis of technical and/or financial factors, which are generally more complex in nature. Otherwise, the developer proceeds to the next steps of the process.

3) Prepare Application Profile: In this sub-process, a profile is made for the application to be migrated to the cloud. It should search for any characteristic of the application that may be a factor regarding its migration to the cloud. This sub-process is again divided into two more sub-processes, related to the relevant application information regarding its usage and technical details, respectively.

a) Usage characteristics: Characteristics of the application related to its use and operation are to be identified here. Main aim is to enlist key functional and non-functional characteristics of the application that may pose as an obstacle to its migration to the cloud. The following points can be used to define the application profile better:

• The main characteristics of the application.

• Number of users accessing the application and their locations.

• Modes of usage based on demand and period of use.

• The cost needed to maintain and operate the application.

b) Technical characteristics: Characteristics related to the technology usage by the application along with any other relevant technical information involved are evaluated. The following points can help formation of the technical profile of the application.

• Application's architecture.

• Technologies used to implement the application.

• Technologies required to run the application.

• Data handling technology used by the application.

• Data traffic caused by the application.

• Any specific quality-of-service required by the application.

• Minimum hardware requirement to run the application properly.

• Any other applications, the target application is dependent on and their above mentioned details.

4) Produce Cloud Service Provider Profile: In this sub-process, the profile for one or more cloud providers must be produced for comparison of their competence regarding the various constraints discussed above. Below are some points that can be taken into consideration while creating the cloud service provider profile:

• The service models made available by the provider (e.g., Saas, Iaas, etc...).

• The resources the cloud service provider offers as part of each of its service models.

• The cost models (e.g., per hour on demand, per hour reserved, market bidding) for each type of resource.

• The provider's offer of any form of service level agreement (SLA) guarantees.

• The number and location of data centers of the cloud service providers.

• Any other useful additional service offered by the cloud service provider.

• The security mechanisms put in place by the cloud service provider.

• The implementation of technologies and resources offered by the provider support?

• Is the access to its internal operational logs allowed by the Cloud service provider or not.

• The kind of support services offered by the cloud service provider.

5) Evaluation of Technical and Financial Constraints: The goal of this sub-process is to the combined evaluation of the organization profile, the application profile and the profile of the cloud service provider. Till now in this workflow we have considered those constraints directly or indirectly affecting the cloud migration of the component (application or data). These constraints were chosen based on early reviews of the cloud migration.

a) Evaluation of financial constraints: It should be done whenever possible. Financial constraints should be evaluated before any other technical constraints. It is because cost restrictions generally are critical to many small to Medium-size organizations. A profound example of financial constraint is the cost to operate the application after its migration to the cloud. Another category of financial constraint is the cost of migration itself, which generally involves the transfer of large volumes of data from the organization to the cloud provider abiding by the various above mentioned constraints.

b) Evaluation of organizational constraints: Any organizational constraint regarding the application or the candidate cloud provider should be evaluated. Examples are physical location of data constraints and language constraints

c) Evaluation of security constraints: These constraints main aim is to match the cloud service provider's security with the organization's requirement like encryption level implemented by the provider for communication within and outside the cloud infrastructure or protection level imposed on the provider's virtualized resources.

d) Evaluation of communication constraints: This is related to the application's communication requirements (bandwidth, latency, data transfer rate, etc.). It is mostly dependent on the quality of the network services at the organization, and the physical distance to the cloud service provider's data center e.g. an application requires low latency for data transmission between its client and server components, which does not require migration.

e) Evaluation of performance constraints: Performance constraints refer to the capacity of the application in servicing its users in an acceptable amount of time. It is directly related to the capacity of the cloud resources offered by the provider as well as to the quality of the communication facilities with the provider's data center. An example is when the cloud resources offered by a provider are insufficient to fulfill the application's memory or processing requirements..

f) Evaluation of availability constraints: These constraints relate to the availability of the application once it has been

migrated to the cloud. Availability constraints are linked with the SLA guarantees offered by the cloud provider. They can also be affected by communication constraints of the application users. An example is when application users are located in a region with unreliable Internet connections, as this may affect their capacity to reliably communicate with application components deployed on remote cloud servers.

g) Evaluation of suitability constraints: These constraints deal with changing the application suitable making it proper for cloud migration like when the application depends on a software license that is not available on the cloud provider's agreement and also when the application uses a data format not compatible with the data format supported by the provider.

After all types of constraints have been evaluated, there are multiple paths to follow. If no constraints are violated, the migration is recommended and the process moves on to the next stage, which involves the definition of an adequate migration strategy for the legacy application. Otherwise, there are three courses of action to choose from:

(a) Addressing the violating constraints in the application.

(b) Selecting another cloud service provider;

Abort the process in case of chronic constraints which are unsolvable.

6) Addressing Application Constraints: In this path, there should be made an attempt to resolve any application constraints identified in the above mentioned processes. This can be done either by changing the application or by reducing or increasing the migration. In either case, the application profile should be accordingly updated, so that the constraint evaluation path flow can be restarted. This path flow goes on until no more constraints are found, or the decision to abort the migration due to chronic constraints is reached.

7) Change Cloud Provider: The main aim of this path is to select alternative cloud providers, in an attempt to redress violating constraints identified in the previous evaluation step. Depending on the nature of those constraints, different selection criteria can be used like constraints related to the physical location of the application data can be solved by selecting a provider whose data center is located within the required boundaries. Similarly, constraints related to operational cost can be resolved by finding a provider with better resource prices or a cheaper pricing model. After a new candidate cloud provider is selected, a new provider profile should be prepared for it, so that the constraint evaluation path flow can be restarted. As with the previous path, the evaluation cycle goes on until no more constraints are found or it is decided to abort the migration due to chronic constraints.

8) Define Migration Strategy: This path should be executed once there are no critical constraints precluding application migration to the cloud. In this case, a migration strategy should be defined that takes into account all the issues raised during the process execution. Such strategy should minimize the initial migration cost and also the cost of managing and operating the application after it has been migrated to the cloud. The following points can be used to plan the migration task efficiently:

• The type of activities that must be performed to successfully accomplish the migration.

• The strategies those are most appropriate for the migration effort.

Given the emerging nature and relative immaturity of current cloud technologies, it is always recommended that the organization starts with a pilot migration project, in order to investigate if the behavior of the application in the cloud will be as expected. In this basis, it is highly advisable to consider multiple migration scenarios, possibly involving different migration strategies and costs. This approach will build confidence that the organization will find a cloud deployment scenario that effectively fits its needs and constraints.

9) Perform Migration: At this point the organization should perform the actual migration of the application to the cloud, according to the migration strategy. This activity encompasses any concrete action required to perform the migration, from moving application components and data to the selected cloud provider, to carrying out all changes and tests necessary to ensure that the application will run in the cloud according to needs and expectations of the organization.

VI. RELATED WORK

A number of works have been carried out in the concept of cloud computing including the Architectural structure and the fundamental needs [1-6]. The need and motivation and categories of data migration is well explained in the work by [7]. The work explains the efficient procedural work flow of cloud migration [8]. In another work [9] the authors explains evaluation of the various constraints for the real time migration in the cloud environment, and the steps to overcome them [10].

VII. CONCLUSION

Using Cloud Computing the complexity and cost of owning and operating computers and networks can be significantly reduced. Customization of Cloud services can be done and it is also flexible to use, advanced services can be offered by the providers so that an individual company might not have to spend money or expertise to develop rather choose to migrate its systems to the cloud. There are various categories of migration for the user to choose from for the best of his/her interests. By following detailed workflow in this paper of migration procedure, the user can ensure successful and fruitful migration of its IT resources to the cloud by choosing the best cloud service provider. If migration to cloud is done in all proper way, it can be bear wonders for the institution but if it is not then damage to secured data and applications and other inconveniences are inevitable.

ACKNOWLEDGEMENT

The two authors Dines and Anurag acknowledge to SPARK program of CSIR C-MMACS for providing the internship, HOD, Dept. of Computer Science and Engineering

and Director, National Institute of Technology, Rourkela for providing the necessary infrastructures to carry out the work.

REFERENCES

[1] Gouda K C, Radhika T V, Akshatha M, Priority based resource allocation model for cloud computing, International Journal of Science, Engineering and Technology Research (IJSETR), ISSN: 2278 – 7798, (2013), pp 215-219.

[2] Gouda, K. C., and Dwaipayan Acharjee. "An Outlook of cloud computing." International Journal of Science, Engineering and Technology Research 2.6 (2013): pp-1403-1406.

[3] Jamunarani M R., Gouda, K. C., & Nirmala, H. Load Balancing Technique for Climate Data Analysis in Cloud Computing Environment, International Journal of Computer Science, Engineering and Technology, 3(5) (2013): pp-183-185.

[4] K C Gouda, Sambeet Kar, Rohan Patra. "Cloud Computing: What, How, Why", International Journal of Engineering Trends and Technology (IJETT). V4(6):2453-2457 Jun 2013.ISSN:2231-5381. www.ijettjournal.org.

[5] http://en.wikipedia.org/wiki/Cloud_computing

[6] Kris Jamsa, Cloud Computing,.Jones & Bartlett Learning, 2013.

[7] Oracle 9i Database migration Release 2(9.2) October 2002 part no. A96530-02

[8] Quang Hieu Vu ; Asal, R. Legacy Application Migration to the Cloud: Practicability and Methodology Services (SERVICES), 2012 IEEE Eighth World Congress on Digital Object Identifier: 10.1109/SERVICES.2012.47 : (2012) , Page(s): 270 - 277

[8] An Oracle White Paper October (2011) Successful data migration

[9] An Extensible Architecture for Detecting Violations of a Cloud Environment's Constraints during Legacy Software System Migration Frey, S. ; Hasselbring, Wilhelm Software Maintenance and Reengineering (CSMR), 2011 15th European Conference on Digital Object Identifier: 10.1109/CSMR.2011.33: (2011) , Page(s): 269 - 278

[10] Reliability-Based Design Optimization for Cloud Migration Qiu, W. ; Zheng, Z. ; Wang, X. ; Yang, X. ; Lyu, M.R. Services Computing, IEEE Transactions on Volume: 7 , Issue: 2 Digital Object Identifier: 10.1109/TSC.2013.38, (2014) , Page(s): 223 – 236.