Power and Time Efficient Structural Cloud for Hierarchical Peer to Peer Networking

R.A.Karthika
Computer Science & Engineering Department
Noorul Islam University, Tamilnadu
India-629 180

A.Shajin Nargunam
Computer Science & Engineering Department
Noorul Islam University, Tamilnadu
India-629 180

ABSTRACT
Cloud computing is a promising profitable infrastructure framework that assures to eradicate the requirement for sustaining expensive computing facilities by companies and institutes alike. Information systems supported on the cloud computing and peer-to-peer (P2P) model are now receiving popularity. In the cloud computing form, a cloud of servers sustain few clients with different types of service like Web pages and databases. On the other hand, each computer is peer and there is no centralized system is maintained in the P2P model. It is getting further important to confer how to decrease the computational time and power consumption, multi task handling in distributed Hierarchical P2P (HP2P) network for cloud computing to achieve maximum potentiality. To minimize the time and power consumption in cloud computing environment, the proposed work considers a Web and FTP application on cloud based P2P distributed network. To start with, evaluated the process of identifying level of each server peer consuming power and task execution time to perform web requests from client peers. The document which has to be shared are clustered and shared efficiently among the different systems in the cloud computing environment. The hierarchical peer distribution process is initiated to evolve power and time efficient algorithm for client peers to select server peer in a collection of server peers with satisfied constraint on deadline task response time. The experimental evaluation of the performance of Cloud Computing Services (CCS) including Amazon EC2, which is presently the main commercial cloud are evaluated in terms of F-measure, entropy, number of documents attached to distributed peers, participated peer node ID.

General Terms
Cloud structure, Hierarchical cloud, Peer to Peer cloud.

Keywords
Cloud computing, P2P network, distributed hierarchical model, similarity clustering, cloud computing services.

1. INTRODUCTION
Scientific computing entails an increasing amount of resources to transport results for continually increasing problem sizes in a sensible time frame. Cloud computing presented an option in which resources are no longer hosted by the researchers' computational services, but are chartered from huge data centers only when desired. Even with the subsistence of numerous cloud computing offerings such as Amazon and Go Grid, the prospective of clouds for scientific computing remains chiefly unexplored.

Clouds can be a substitute to supercomputers and dedicated clusters, greatly more consistent policy than grids, and a much more scalable platform than the major of product clusters. The cloud computing services are of three types, IaaS (Infrastructure as a Service), that is, delivery of computer infrastructure as a service, PaaS (Platform as a Service), that is, APIs for creating applications on an abstract platform, and SaaS (Software as a Service), that is, support for running software services from the remote place. Many clouds have already been expanded, but not all offers virtualization, or even computing services. The scientific community has not yet adopted PaaS or SaaS solutions, chiefly to discard legacy applications and for the lack of required scientific computing services, correspondingly. The cloud computing environment is shown in Figure 1.

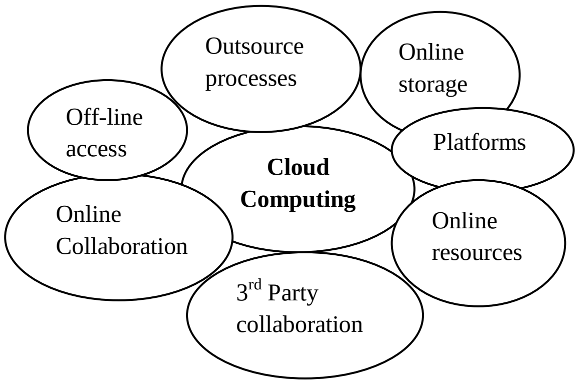

Figure 1: Cloud Computing Services

Through cloud computing environment, it is easy to access the computers from anywhere in the world. This is a way to boost capability or add capabilities on the fly without spending in new infrastructure framework, exercising new workforces, or licensing new software. Cloud computing includes any type of service within a short valid time over the Internet and enlarges IT's existing capacity. A cloud computing environment comprises of online storage processes which are used to accumulate the online satisfied items lively. The organizer chooses links for data sharing and collaboration. Or, the organizer may previously be in a discussion and decides to add data conferencing. Users admit cloud computing using networked devices such as desktop computers, laptops and smart phones. Some of these devices - cloud clients - rely on cloud computing for all or a mainstream of their applications so as to be basically ineffective without it. Examples are thin clients and the browser-based Chrome Book. Many cloud applications do not involve definite software on the client and as a substitute use a web browser to cooperate with the cloud application. With Ajax and HTML5 these Web user interfaces can attain a parallel or even better look and feel as resident applications. Some cloud applications, though, sustain exact client software devoted to these applications (e.g., virtual desktop clients).

To improve the cloud computing process, the proposed work presented an evaluation of the process of cloud computing Cloud computing is a promising profitable infrastructure framework that assures to eradicate the requirement for sustaining expensive computing facilities by companies and institutes alike. Nevertheless, the present profitable clouds constructed to sustain web and small database workloads, which are extremely diverse from distinctive systematic computing workloads. Furthermore, the exploit of virtualization and resource time-sharing might initiate considerable performance consequences for the challenging systematic computing workloads. In 2010, Alexandru Iosup [1] examined the presentation of cloud computing services for systematic computing workloads.

Cloud computing has appeared as a novel tool that presents huge amount of calculation and data storage capability to its users with a guarantee of enlarged scalability, high accessibility, condensed organization and preservation costs. To deal with some concerns like, transparency of obtaining and discharging effective computing resources, other virtualization and network connection overheads Alexandru Iosup have considered and employed C-Meter [2] which is a manageable, extensible, and user-friendly structure for producing and presenting test workloads to workout clouds. So, it is of huge significance to review the performance of figuring out clouds in terms of diverse metrics. As the employment of cloud computing environment increases, it turns out to be decisive to appreciate the presentation of these environments. In 2010, Alexandru Iosup [3] examined the reliability of cloud computing services, but it does not achieve examination of other construction cloud services.

As a type of promising business computational form, in 2010, Shufen Zhang [4] analyzed that Cloud Computing dispense computation job on the resource team which comprises of enormous computers. Consequently, the application systems can increase the estimation power, the storage space and software examination consistent with its demand. As cloud computing develops into other widespread, the energy utilization of the network and computing resources that support the cloud will rise. Jayant Baliga et al. [5] presented an examination of energy utilization in cloud computing. The study judges both public and private clouds [6], and contains energy utilization in controlling and broadcast with data processing and data storage in a secure way [7]. But the approach defined fails to process the other cloud computing services. To manage the services in the cloud computing environment, it is necessary to process it with a particular interval of time by adapting the approach like Distributed and Scalable Time Slot Allocation technique [8]. The research has taken over the background and principle of cloud computing [9], the quality, approach and reality. But the research does not analyze the hidden troubles. Through the exploit of virtualization and source time-sharing, clouds provide with a distinct place of substantial resources a huge user stand with diverse needs. Thus, clouds have the possibility to present services based on time and power requirements of P2P networks.

2. LITERATURE REVIEW

their holders the remuneration of a wealth of scale and, at the similar time, develop into an option for scientists to collect grids, and similar construction environments [10]. A time and cost scheduling algorithm [11] is defined with Instance-Intensive Cost-Constrained Workflows on Cloud Computing Platform by conducting a simulation [12].

The crucial ideology of cloud computing is to construct computing which can be dispensed in a huge number of dispersed computers, relatively than restricted computer or remoter server. As present cloud computing services are inadequate for scientific computing at huge, they might still be a fine resolution for the scientists who require resources instantaneously and provisionally. In this work, we made an additional analysis of cloud computing services offered by clouds based on power and time requirements.

3. CLOUD COMPUTING SERVICES OF HP2P NETWORKS

The proposed cloud computing services for power and time requirements are efficiently designed for evaluating the performance of service processes of P2P network by sharing the documents based on clustering process and conceptual similarity technique used for searching the document based on a keyword. The proposed system also analyses the process of identifying the requirements of network data transfer rate. The proposed system carries four types of operations such as peer node document distribution, conceptual search similarity, similarity clustering and moving cluster object to other systems. The architecture of CCS for power and time requirements of Hybrid P2P network is shown in Figure 2.

The first phase describes the process of distributing the document to be shared to the peer nodes available in the cloud computing environment.

The second phase describes the process of conceptual similarity based on feature words and conceptual words. The feature words are the words which are closely related to a particular concept. For instance, take a word “football” the featured words are goal, referee, foul, and penalty. The conceptual words are words which describes the information related to that given data.

The third phase describes the process of clustering based on similarity search. The peer nodes in the cloud computing environment displays a set of nodes that has a set of similar clustered documents displayed with their corresponding node ID.

The fourth phase describes the process of perceptual changes on the content of the document. This is possible by tracking the peer node object. The clustered document can also been moved easily from one peer node to other peer node available in the cloud computing environment.

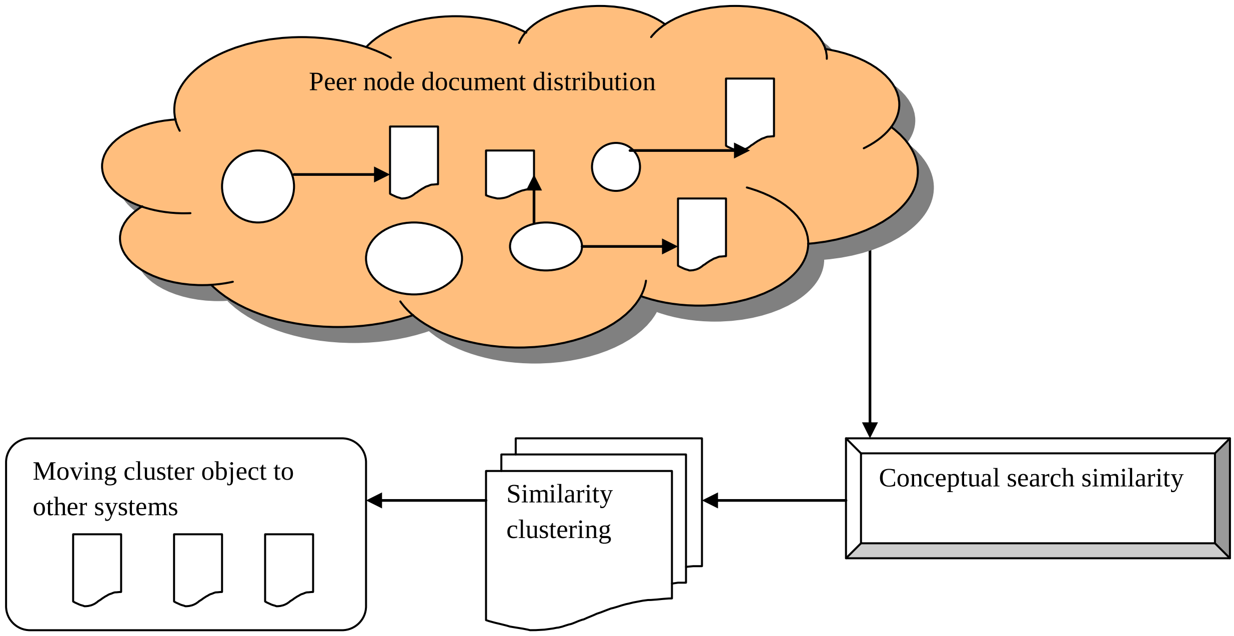

Figure 2: Architecture diagram of CCS for HP2P networks

3.1 Conceptual Search Similarity

The proposed CCS first chooses the document which is to be distributed with other peer nodes. The document is sent to the other peer nodes based on its requests. Then the conceptual similarity paradigm is done based on the similarity retrieval. The first two operations (Figure 3) are described here using procedures.

Input: Set of peer nodes N_i where i = {1,2,…,n}, set of documents doc

// Document distribution

Step 1: For Each node N

Step 2: Maintain document (docs.)

Step 3: End For

Step 4: If node N_2 wants to send its doc

Step 5: N_2 select the destination peer node $N_{1`}$ (to which node N_2 wants to send a document)

Step 6: N_2 select the document doc_i from the list of documents to be sent e.g., crick3.txt

Step 7: Store the selected document doc_i (crick3.txt) at the node-specified destination (N_1)

Step 8: End If

Step 9: End

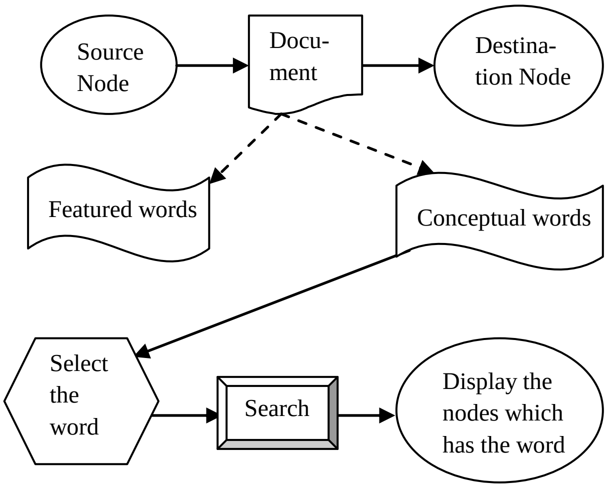

Figure 3: Process of conceptual search similarity

// Conceptual Similarity Search

Step 10: For each doc_i

Step 11: Extract features from document doc_i (crick3.txt)

Step 11.1: select considerable sentences

Step 11.2: partition considerable sentences into feature vectors

Step 12: End For

Step 13: Cluster the features into concepts

Step 14: Identify the primal conceptual words C_i

Step 15: Node selects the conceptual word C_1 to be searched in a document list

Step 16: Identify the documents which have a conceptual word C_1

Step 16: End

The conceptual search similarity is done based on the document which is to be shared among the peer nodes present in the cloud computing environment.

3.2 Similarity Clustering and Structural Object Movement

Clustering in the proposed CCS (Conceptual Similarity Search) is based on similarity search word obtained through CSS briefly described in section 3.1. The clustering process and object movement is described below using pseudo code.

// Similarity Clustering

Input: Let $f_1, ..., f_n$ be feature vectors,

G the number of generated clusters

u and v are controlled parameters.

Step 1: For each Document doc_i

Step 2: Identify the feature vectors fi

Step 3: End For

Step 4: Select the feature to be searched in a doc_i

Step 5: Start search in each doc_i

Step 6: If found,

Step 7: Retrieve the document

Step 8: Form a cluster C_a

Step 9: Store the document in C_a

Step 10: Else

Step 11: Ignore the document

Step 12: End If

Step 13: Stop until all doc_i processed

Step 14: End

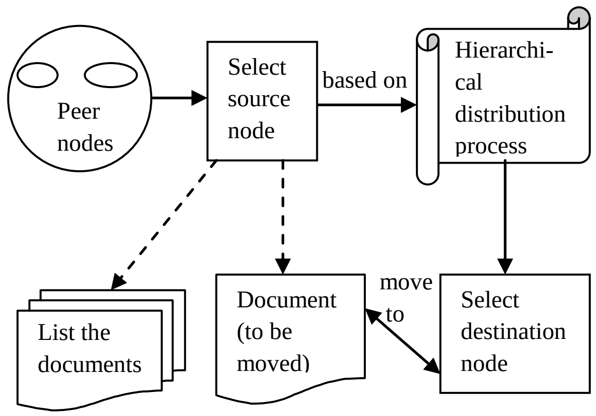

Figure 4: Process of movement of document object from P2P

In our similarity clustering method, if terms in a feature vector f_i are controlled in the cluster C_a, put f_i into C_a and recompute C_a such that $C_a = (C_a + f_i) / 2$. If terms in fi are contained in C_a, then ignore f_i. If terms in f_i are not present, put f_i into other clusters. If f_i is not assigned to any clusters, create a new cluster that contains f_i.

// Object Structural movement

Step1: For a peer,

Step 2: Find all peers which have documents contains similar conceptual words.

Step 3: End For

Step 4: Store all the peers in a list.

Step 5: Select the source peer node from a list

Step 6: List the documents of source peer

Step 7: Select a document d_a from document list to be sent

Step 8: If the destination peer node list is not empty,

Step 8.1: select a peer in the list to which document is to be sent.

Step 8.2: Move the document d_a to destination peer node

Step 8.3: End if

Step 9: End

The hierarchical distribution process is used for the movement of document object from source to the destination. The processing documents are distributed among the nodes in the network from a source peer node to destination peer node. The merits of distributed architectures expand from enhanced compensability, permitting nodes to be assembled by putting together diverse subsystems. The peer node document object tracking presented a strong prospective to sustain immediate behaviors more efficiently.

4. PERFORMANCE EVALUATION

The proposed cloud computing services for HP2P networks based on its power and time requirements are implemented in Java. The experiments run on an Intel P-IV machine with 2 GB memory and 3 GHz dual processor CPU. The performance evaluation tests aimed at comparing the direct invocation for cloud computing services using hierarchical distribution process with challenging interactions through traditional scientific computing. The cloud computing services architecture for HP2P network is shown in Figure 2. At first set up, it identifies the source and destination node from cloud computing environment for moving the document object from source peer node to destination peer node. The source node will effectively choose the destination node based on hierarchical distribution process. So, the object movement is successfully done at cloud computing environment. The proposed evaluation for cloud computing services carries three types of operations generic (Identifying destination node, conceptual similarity search, hierarchical distribution process…). Operations can be assigned to different services and components in the infrastructure.

The performance of the proposed cloud computing services based on its time and power requirement is evaluated by the following metrics:

i) F-measure

ii) Entropy

iii) Number of documents attached to distributed peer nodes

F-Measure is used to test the effectiveness of the retrieval of information from the nodes by considering both precision (p) and recall (r).

p = number of retrieved docs correctly / number of retrieved docs

r = number of correct docs / number of docs that should have been returned

Based on Keith van Rijsbergen's, effectiveness measures:

$$E = 1 - \left(\frac{\alpha}{P} + \frac{(1-\alpha)}{R} \right)^{-1}$$ ------- (equation 1)

Their relationship is,

$$F_\beta = 1 - E \text{ where } \alpha = \frac{1}{1+\beta^2}$$

------- (equation 2)

β = number of peer nodes in cloud computing environment

Entropy is defined as a measure of the loss of information in a transmitted message from one peer node to other peer node.

Number of documents attached with the destination peer node should be with minimal loss of information.

5. RESULTS AND DISCUSSION

In this work, we have seen how a HP2P networks can be designed for cloud computing service composition to capture the power and time requirements for service performance to other systems written in mainstream languages such as Java. We run independent tests with growing number of applications, and constant number of service requests sent by each user. The performance graph and table describes the evaluation of the performance of cloud computing service using hierarchical distribution model.

Table 1. Number of clustered document vs. F-measure

Number of clustered document	**F-measure**	
	Proposed CCS using hierarchical distribution process	**Existing CCS using scientific computing**
10	1.15	1.08
20	2.54	1.19
30	3.24	2.31
40	4.65	3.45
50	5.9	4.21

The above Table 1 described the performance of the proposed CCS using Hierarchical Distribution Model (HDM) based on F-measure.

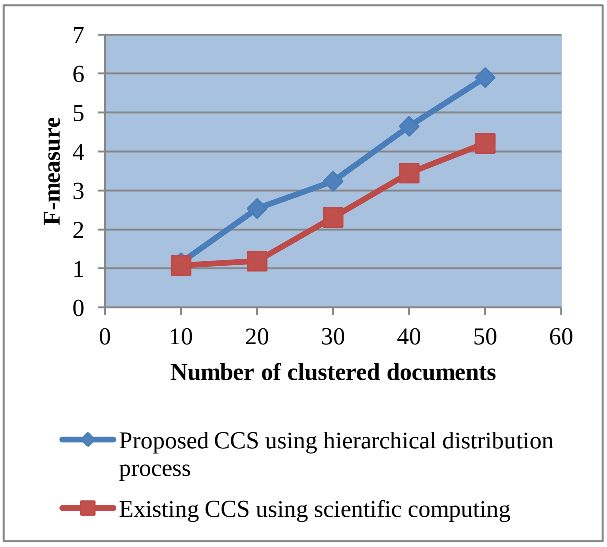

Figure 5: Number of clustered document vs. F-measure

Figure 5 describes the F-measure for performing the document object sharing in the cloud computing environment. Various numbers of clustered documents are used in the experimentation to validate the proposed CCS using HDM. Comparison result of the proposed CCS using HDM with an existing CCS based on scientific computing is measured in terms of F-measure. When a number of clustered documents in the cloud computing environment increases, the effectiveness of the process of document retrieval is high in the proposed CCS using HDM contrast when compared to an existing CCS based on scientific computing. The performance graph of the proposed CCS using HDM is shown in the Figure 5. The variance in the F-measure for retrieving clustered document would be 10-15% high in the proposed CCS using HDM.

Table 2. Number of clustered document vs. Entropy

Number of clustered document	**Entropy**	
	Proposed CCS using hierarchical distribution process	**Existing CCS using scientific computing**
25	15	20
50	30	40
75	43	59
100	56	75
125	62	86

The above Table 2 described the performance of the proposed CCS using Hierarchical Distribution Model (HDM) based on Entropy.

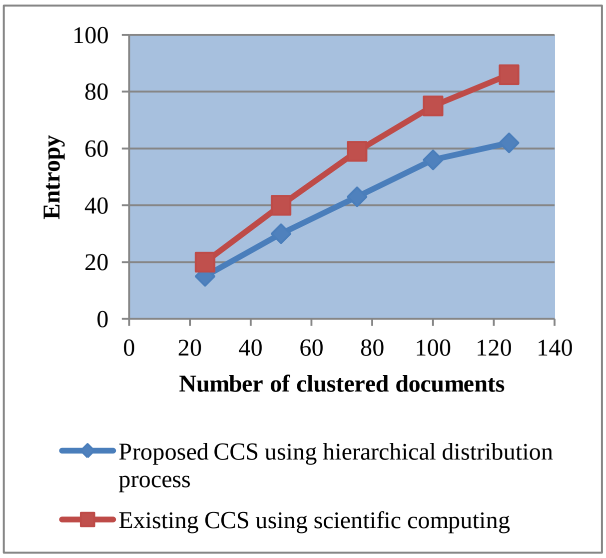

Figure 6: Number of clustered documents vs. Entropy

Figure 6 describes the entropy for identifying the loss of information on clustered document in the cloud computing environment. Various numbers of clustered documents are used in the experimentation to validate the proposed CCS using HDM. Comparison result of the proposed CCS using HDM with an existing CCS based on scientific computing is measured in terms of entropy. When number of clustered documents in the cloud computing environment increases, the loss of information while document retrieval is low in the proposed CCS using HDM contrast to an existing CCS based on scientific computing. The performance graph of the proposed CCS using HDM is shown in the Figure 6. The variance in the entropy for retrieving clustered document would be 20-25% low in the proposed CCS using HDM.

Table 3. Number of peer nodes vs. number of documents attached

Number of peer nodes	Number of documents attached	
	Proposed CCS using hierarchical distribution process	**Existing CCS using scientific computing**
10	5	2
20	14	10
30	24	19
40	35	24
50	42	30

The above Table 3 described the performance of the proposed CCS using Hierarchical Distribution Model (HDM) based on number of documents attached with the destination peer node.

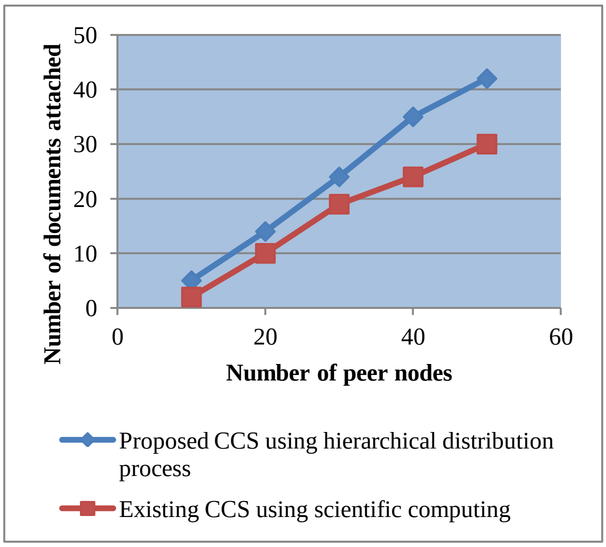

Figure 7: Number of peer nodes vs. number of documents attached

Figure 7 describes the number of documents attached to the distributed peer nodes in the cloud computing environment. Comparison result of the proposed CCS using HDM with an existing CCS based on scientific computing is measured in terms of number of documents attached. When number of peer nodes in the cloud computing environment increases, the document retrieval and attachment is high in the proposed CCS using HDM contrast to an existing CCS based on scientific computing. The performance graph of the proposed CCS using HDM is shown in the Figure 7. The variance in the entropy for retrieving clustered document would be 30% high in the proposed CCS using HDM.

Finally, it is observed that the proposed Hierarchical Distribution Model used for cloud computing service is based on the requirements of power and time based peer clouds in the cloud computing environment. The process of choosing the destination peer node is done effectively by the HDM and the conceptual similarity search is done efficiently and also the document retrieval process is done with minimal loss of information.

6. CONCLUSION

In this paper, Cloud Computing Service has efficiently implemented using Hierarchical Distribution Model for constructing cloud computing to fulfill the interoperability and reliability of Information Technology. The distributed peer nodes are accessed to perform the message transformation from one peer node to other peer node with minimal loss. The CCS based on power and time requirements using Hierarchical Distribution Model infrastructure framework is measured with metrics such as F-measure to test the effectiveness of information retrieval, entropy to test the loss of information and number of documents attached to the distributed peer in the cloud computing environment. Standard users' tasks are taken to conduct the performance evaluation of the proposed power and time based requirements for cloud computing services. The results showed that the proposed CCS using Hierarchical Distribution Model is 70% better in performing the users' tasks based on the information retrieval process amongst the peer nodes available in the cloud computing environment.

7. REFERENCES

[1] Alexandru Iosup et al., 2010 "Performance Analysis of Cloud Computing Services for Many-Tasks Scientific Computing", IEEE Transactions on Parallel and Distributed systems.

[2] Nezih Yigitbasi, Alexandru Iosup, and Dick Epema, "C-Meter: A Framework For Performance Analysis Of Computing Clouds," in *Proceedings of CCGRID'09*, pp. 472–477.

[3] Alexandru Iosup, Nezih Yigitbasi, and Dick Epema, Jan 2010 "On The Performance Variability Of Production Cloud Services," TU Delft, Tech. Rep. PDS.

[4] Shufen Zhang et al., 2010 "Analysis and Research of Cloud Computing System Instance", Second International Conference on Future Networks, ICFN '10.

[5] Jayant Baliga et al., 2011 "Green Cloud Computing: Balancing Energy in Processing, Storage, and Transport", Proceedings of the IEEE.

[6] Luqun Li, 2009 "An Optimistic Differentiated Service Job Scheduling System for Cloud Computing Service Users and Providers", Third International Conference on Multimedia and Ubiquitous Engineering, MUE '09.

[7] Zhidong Shen et al., 2010 "The security of cloud computing system enabled by trusted computing technology", 2nd International Conference on Signal Processing Systems (ICSPS).

[8] Chih-Kuang Lin et al., 2011 "A Distributed and Scalable Time Slot Allocation Protocol for Wireless Sensor Networks", IEEE Transactions On Mobile Computing, VOL. 10, NO. 5.

[9] Shuai Zhang et al., 2010 "Cloud Computing Research and Development Trend", Second International Conference on Future Networks ICFN '10.

[10] Xiao Liu, Dong Yuan et al., 2010 "SwinDeW-C: A Peer-to-Peer Based Cloud Workflow System", Handbook of cloud computing / Borko Furht and Armando Escalante (eds.), part 2, chapter 13, pp. 309-332.

[11] Ke Liu, Hai Jin et al., 2010 "A Compromised-Time-Cost Scheduling Algorithm in SwinDeW-C for Instance-Intensive Cost-Constrained Workflows on Cloud Computing Platform," International Journal of High Performance Computing Applications.

[12] R. N. Calheiros, R. Ranjan, C. A. F. De Rose, and R. Buyya, 2009 "CloudSim: A Novel Framework for Modeling and Simulation of Cloud Computing Infrastructures and Services," Technical Report, Grid Computing and Distributed Systems (GRIDS) Laboratory, Department of Computer Science and Software Engineering, The University of Melbourne.

8. AUTHOR'S PROFILE

R.A.Karthika received her Master's degree in Computer Science and Engineering in 2011 from Noorul Islam University, Tamilnadu, India. She is currently a PhD student in the Faculty of Computer Science and Engineering at Noorul Islam University (NIU), Tamilnadu, India. Her research interests include Cloud Computing, Peer-To-Peer Computing, and Computer Architecture.

A.Shajin Nargunam is a Professor of Computer Science and Engineering at Noorul Islam University (NIU), Tamilnadu, India. His added designation is "Academic Director" of Noorul Islam University (NIU) at Noorul Islam Centre for Higher Education (NICHE). He received his PhD in Computer Science and Engineering from National Institute of Technology (NIT), Calicut in 2009. In 2003, he won the appreciation award for designing a low cost super computer at Science and Technology exhibition. Shajin had been working at NIU for the past 15 years, since 1997. He has got so many International and National Journal Publications into his credit. His research interests include Mobile Computing, Cloud Computing, Virtualization Technology, Cluster Computing, Peer-To-Peer Computing, Network Storage and Network Security.