IJCSN - International Journal of Computer Science and Network, Volume 6, Issue 5, October 2017
ISSN (Online) : 2277-5420
www.IJCSN.org
Impact Factor: 1.5

A Service Level Agreement Effective Optimal Cloud Based Road Traffic Management System

[1] Md. Rafeeq; [2] Dr C.Sunil Kumar; [3] Dr. N. Subhash Chandra; [4] Dr Arunavaranasi

[1] CMR Technical Campus, Kandlakoya, JNT University,
Hyderabad, India
[2] Professor in CSE,SNIST, JNTU Hyderabad,
Hyderabad, India,
.
[3] Professor in CSE, CVRCE, JNTU Hyderabad,
Hyderabad, India,

[4] Professor & Head in CSE,SNIST, JNTU Hyderabad,
Hyderabad, India,

Abstract—With the fabulous growth of population and the increasing road traffic, the demand for optimized traffic data collection and management framework is also demanding. The collection of traffic data using various sensors and other capture devices are been addressed in multiple researches deploying the mechanism using geodetically static sensor agents. However to avoid the congestion, the parallel research works has proposed frameworks based on cloud based data centers. Thought, those approaches does not propose any technique to decrease the cost and improve the service level agreements to match with the existing industry and research demands. Thus, this paper proposes a cloud based automatized framework for VM (VM) migration to increase the Service Level Agreement (SLA). Without negotiating the cost for storage and energy. The major attainment of this work is to minimize the SLA violation compared to existing VM migration methods for load balancing (LB). The extensive practical demonstrations of virtualization and migration advantages are also carried out in this work. With the extensive experimental setup the work provides the comparative analysis of simulations for popular existing methods and the proposed framework.

Keywords — *optimal migration, SLA improvement, VM, image formats, cost, performance, evaluation matrix*

1. Introduction

LB(LB) Techniques on cloud computing (CC) is the generic framework based process where the generated workloads are distributed over multiple data center resources. The LB techniques bring the advantage of lower response time [1]. However the cost of duplication of resources is also to be taken care as an extra cost. The cloud data center based LB is distinguished from the domain name service based LB.

The domain name service LBs deploys the hardware and software components to balance load for the hardware resources, whereas the cloud based LB techniques deploys the software algorithms or protocols to distribute the load over multiple data center nodes. Also it is to be understood that, the cloud based LB techniques allows the customers to use the global or geodetically distributed services based on geodetically distributed servers **[2, 3]**. Several parallel researches are been carried out to demonstrate the benefits of LB on cloud based data centers as handling the high unexpected traffic generally referred to cyber spikes. Making the application scalable based on demand without degrading the performance, increases the reliability at the cost of VM migration.

However the recent researches constraint to achieve the optimal SLA violation during VM Migration. Thus this work demonstrates A SLA Effective Optimal VM Migration Technique for LB on cloud data centers using proposed three phase optimal VM migration technique.

2. Virtualization Advantages for Cloud Data Centers

This work also highlights the benefits of VM migrations and also evaluates the parameters influencing the performance and productivity [3].

2.1 Open Access Control

Table I: Parameters For Open Access Control **[3]**

Parameter Type	Parameter Name	Access Permissions	
		Traditional	VM Migration
Processing	CPU Type	Not Allowed	Allowed
	Allocation	Allowed	Allowed
	Priority	Allowed	Allowed
Memory	Size	Allowed	Allowed
	Buffer	Not Allowed	Allowed
Storage	Access IDE Bus	Not Allowed, Physical	Allowed, Logical
	Capture Mode	Not Allowed	Allowed
	Library Group	Allowed, Physical	Allowed, Logical
Network	IP Address	Allowed	Allowed
	MAC Address	Not Allowed	Allowed
	Internal Network	Partially Allowed	Allowed

Copyright (c) 2017 International Journal of Computer Science and Network. All Rights Reserved.

The VMs come with a reduced abstraction in the system level and allows the provider, customer and researchers to access more properties of the system. The access to computing environment data, system level codes, hardware utilization statistics, traces of the active application, failing and down timing component configurations and the guest operating system configuration parameters and the ability to control them independently helps to understand the performance perimeters (Table 1).

The access to computing environment data, system level codes, hardware utilization statistics, traces of the active application, failing and down timing component configurations and the guest operating system configuration parameters and the ability to control them independently helps to understand the performance perimeters (Table 1).

2.2 Optimum Hardware Control

VMs come with a flexibility to change or alter the operating system and hardware components seamlessly. After the initial cost for setting up a virtual environment, the users are free to modify the computing system including the operating system, libraries, tools and other supporting patches without investing the full time needed for computing system change or upgrade (Table 2).

Table 2: Reduced Hardware Upgrade Constraints [4]

Parameter Type	Parameter Name	Accessibility	
		Traditional	Virtual Machine Migration
Operating System	Version	Available	Available
	Interoperability	No Continuous Availability	Available
	Patch	Available	Available
Development Environment	Patch	Available	Available
	Device Driver	No Continuous Availability	Available
	Version Control	Available	Available
Configuration	Configuration Delay	Very High	Low

2.3 Optimal Replication Control

The replication of the VMs using the snapshot feature allows the users to take timely and on demand backups of the VM images. Thus the backups help to quickly reproduce the same computing environment without investing the complete setup time (Table 3).

Table 3: Reduced Replication Duration [4]

Parameter Type	Replication Time	
	Traditional	VM Migration
Windows Server	50 to 90 Mins	Just in Time
MAC Servers	40 to 60 Mins	Just in Time
Linux Servers	30 to 40 Mins	Just in Time

2.4 Service Provider Support for VM Migration

Table 4: Service Provider Support For Migration [5]

Server Type	Amazon Cloud	Microsoft Azure Cloud	Google App Engine Cloud	IBM Bluemix Cloud	Private Hosted Cloud
Windows Server	YES	YES	YES	YES	NO
MAC Servers	YES	YES	YES	YES	NO
Linux Servers	YES	YES	YES	YES	NO

The VMs are hosted by all service providers with similar configurations but with added advantages. Hence adapting to VM computing is the best choice to avoid the lack of support and facility availability (Table 4).

2.5. Optimum Manageability

Application on VMs hosted on cloud is always liable for automatic and regular updates from the service provider without any extra cost. However in the other side, hosting the traditional system demands the cost and time implications for updates

2.6 Optimum Cost Control

Due to the tremendous competition in the cloud service provider space, the drop of price for each virtualization component used in the VM configuration is dropping with an increasing speed. Hence rather than up-gradation cost for traditional systems, the cloud based VMs are very much cost effective (Table 5).

Table 5: Cost Reduction for VM Migration [5]

Server Type	Amazon Cloud	Microsoft Azure Cloud	Google App Engine Cloud	IBM Bluemix Cloud
2013	$0.64	$0.70	$0.63	$0.61
2014	$0.48	$0.45	$0.49	$0.47
2015	$0.35	$0.39	$0.31	$0.30
2016	$0.28	$0.26	$0.29	$0.26

Copyright (c) 2017 International Journal of Computer Science and Network. All Rights Reserved.

Cost Compatibility is projected in this work (Figure. 1)

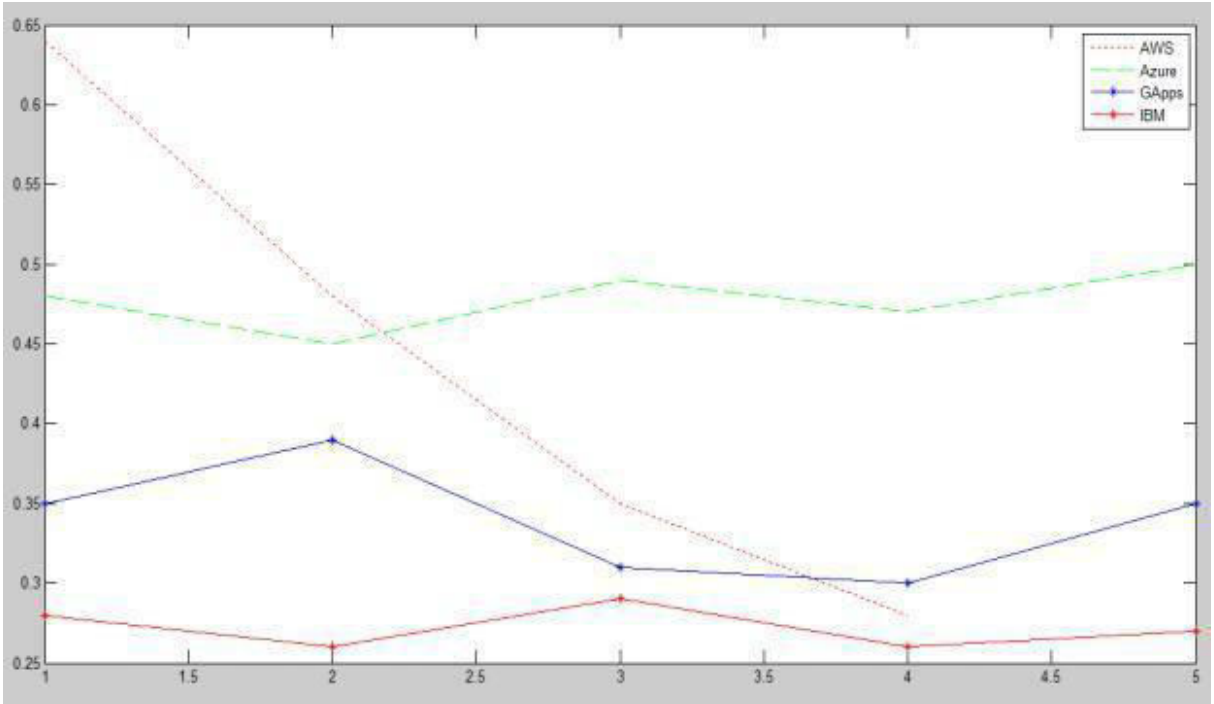

Figure. 1. Cost for VM migration/hosting [5]

Henceforth it is been demonstrated that the VM migration and hosting are been advocated by all major service providers.

3. Proposed Optimal Migration Framework

This work deploys a cost evaluation function to determine the most suitable VM to be migrated considering the least SLA violation. The framework for optimal migration is presented here (Figure. 2).

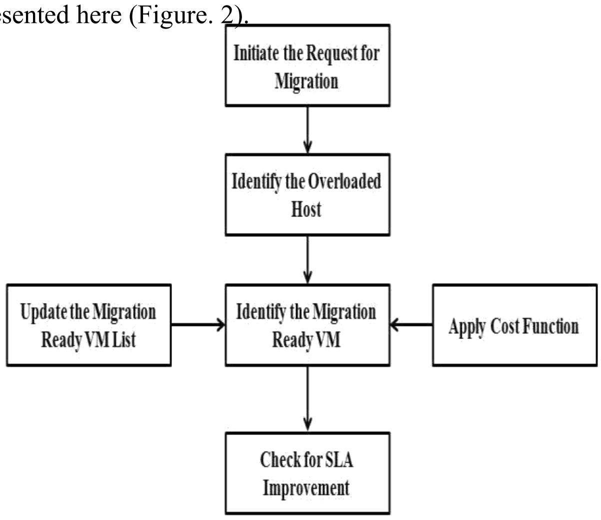

Figure. 2. Optimal framework VM migration **[6]**

The proposed framework is classified into three major

algorithm components as VM identification, VM migration and Cost Function. Algorithms for all three phases are been discussed here:

3.1 VM Identification

The first phase of the algorithm analyses the highest loaded node and migrates the VM to the available less loaded node. After identifying the source and destination, the algorithm identifies the VM to be migrated **[6]**. The outcome of this algorithm is to obtain optimal load balanced condition for the data center after VM migration. The detail of the algorithm is explained here:

Step-1.1. Calculate the load on each node in the data center

Copyright (c) 2017 International Journal of Computer Science and Network. All Rights Reserved.

IJCSN - International Journal of Computer Science and Network, Volume 6, Issue 5, October 2017
ISSN (Online) : 2277-5420
www.IJCSN.org
Impact Factor: 1.5

$$Phy_{CPUCapacity} = \sum_{i=1}^{n} VM(i)_{CPUCapacity} \quad (1)$$

$$Phy_{MemoryCapacity} = \sum_{i=1}^{n} VM(i)_{MemoryCapacity} \quad (2)$$

$$Phy_{IOCapacity} = \sum_{i=1}^{n} VM(i)_{IOCapacity} \quad (3)$$

$$Phy_{NetworkCapacity} = \sum_{i=1}^{n} VM(i)_{NetworkCapacity} \quad (4)$$

$$\Pi = (Phy_{CPUCapacity} + Phy_{MemoryCapacity} + Phy_{IOCapacity} + Phy_{NetworkCapacity}) \quad (5)$$

Step-1.2. In the second step, the algorithm identifies the highest and lowest loaded node in the data center

$$MAX = \begin{cases} If\ \Pi_i > \Pi, \ then\ \Pi_{MAX} = \Pi_i \\ Else\ \ \Pi_j > \Pi_i, \ then\ \Pi_{MAX} = \Pi_j \end{cases} \quad (6)$$

$$MIN = \begin{cases} If\ \Pi_i < \Pi_j, \ then\ \Pi_{MIN} = \Pi_i \\ Else\ \ \Pi_j < \Pi_i, \ then\ \Pi_{MIN} = \Pi_j \end{cases} \quad (7)$$

Step-1.3. Once the source and destination is identified as MAX and MIN respectively, the identification of VM to be migrated is carried out. During the identification, the optimal load balanced condition is identified [5]-[7].

$$VM(i) = VM(i)_{CPUCapacity} \quad VM(i)_{MemoryCapacity} \quad VM(i)_{IOCapacity} \quad VM(i)_{NetworkCapacity} \quad (8)$$

$$MAX - VM(i) = \Delta_{Source} \quad (9)$$

$$MIN \quad VM(i) \quad Destination \quad (10)$$

Step-1.4. After the calculation of the new load, the source and destination nodes must obtain the optimal load condition, where the loads are nearly equally balanced [6].

$$\begin{cases} If\ \Delta_{Source} \approx \Delta_{Destination}, \ Then\ Migrate\ VM(i) \\ Else\ i = \in (n) \end{cases} \quad (11)$$

where n is total number of VMs in Source node.

3.2 *VM Allocation*

During the second phase of the algorithm, this work analyses the time required for VM allocation for the selected virtual machine with other parameters like Energy consumption, Number of host shutdowns, Execution time - VM selection time, Execution time - host selection time and Execution time - VM reallocation time. These parameters will help in generating the cost function

Step-2.1. Calculate the Energy consumption at the source before migration:

$$E_{Source} = \sum_{i=1}^{t} (E_{CPU} + E_{NETWORK} + E_{IO} + E_{MEMORY}) \quad (12)$$

Step-2.2. Calculate the Energy consumption at the destination after migration:

$$E_{Destination} = \sum_{i=1}^{t} (E_{CPU} + E_{NETWORK} + E_{IO} + E_{MEMORY}) \quad (13)$$

Step-2.3. Calculate the difference in Energy consumption during migration:

$$E_{Diff} = \left| E_{Source} - E_{Destination} \right| \quad (14)$$

Copyright (c) 2017 International Journal of Computer Science and Network. All Rights Reserved.

$$\begin{pmatrix} Host_{Down} & VM_{SelectionTime} \\ Host_{SelectionTime} & VM_{Re\,allocationTime} \end{pmatrix} \quad (15)$$

Henceforth the comparative analysis is been demonstrated in the results and discussion section.

3.3 Cost Analysis of Migration

Step-2.4. Calculate the Number of host shutdowns, Execution time - VM selection time, Execution time - host selection time and Execution time - VM reallocation time during migration:

The optimality of the algorithm focuses on the SLA. During the final phase of the algorithm, the migrations is been validated with the help of the cost function to measure the optimality of the cost. The final cost function is described here[8]:

$$Cost(VM) = E_{Diff} + \begin{pmatrix} Host_{Down} & VM_{SelectionTime} \\ Host_{SelectionTime} & VM_{Re\,allocationTime} \end{pmatrix} + SLA_{Violation} \quad (16)$$

4. Performance Evaluation Matrix

A novel matrix to evaluate the performance of the proposed migration algorithm is been coined in this work. The parameters names, details of the parameter with the optimality expectation are been proposed here (Table 6):

Table 6: Performance Evaluation Matrix and Parameters **[6]**

Parameter	Details	Optimality Expectation
Number of hosts	Number of Host Machines during the simulation or testing	Same throughout all simulations
Number of VMs	Number of VMs during the simulation or testing	Same throughout all simulations
Total simulation time	Duration of the Simulation	Same throughout all simulations
Energy consumption	The amount of Energy difference during migration	Expected to be Minimum
Number of VM migrations	Total number of VM migrations	Expected to be Mean of all the techniques
SLA performance degradation	SLA performance degradation due to migration	Expected to be Mean of all the techniques
SLA time	SLA time per active host	Expected to be Mean of all the techniques
SLA violation	Overall SLA violation	Expected to be Minimum
Average SLA violation	Average SLA violation	Expected to be Mean of all the techniques
Host shutdowns	Number of host shutdowns	Expected to be Maximum
Host shutdown – Mean	Mean time before a host shutdown	Expected to be Mean of all the techniques
Host shutdown – Standard Deviation	Standard Deviation time before a host shutdown	Expected to be Minimum
VM migration time - Mean	Mean time before a VM migration	Expected to be Minimum
VM migration time – Standard Deviation	Standard deviation time before a VM migration	Expected to be Minimum
VM selection mean	Execution time for VM selection in mean	Expected to be Minimum
VM selection time - Standard Deviation	Execution time - VM selection standard deviation	Expected to be Minimum
Host selection time - mean	Execution time for host selection in mean	Expected to be Minimum
Host selection time - Standard Deviation	Execution time for host selection in standard deviation	Expected to be Minimum
VM reallocation time - Mean	Execution time for VM reallocation in mean	Expected to be Minimum
VM reallocation time - Standard Deviation	Execution time for VM reallocation in standard deviation	Expected to be Minimum
Total execution time – Mean	Total Execution time for VM reallocation in mean	Expected to be Minimum
Total execution time - Standard Deviation	Total Execution time for VM reallocation in standard deviation	Expected to be Minimum

5. Results

This work has performed extensive testing to demonstrate the improvement over the existing migration techniques [6]-[9]. The various considered migration techniques are listed with the used acronyms here (Table 7):

Copyright (c) 2017 International Journal of Computer Science and Network. All Rights Reserved.

Table 7: Techniques Used for Performance Compression **[9]**

Used Name in this Work	Selection Policy	Allocation Policy
IQR MC	Maximum Correlation	Inter Quartile Range
IQR MMT	Minimum Migration Time	Inter Quartile Range
LR MC	Random Selection	Local Regression
LR MMT	Minimum Migration Time	Local Regression
LR MU	Minimum Utilization	Local Regression
LR RS	Rom Selection	Local Regression
LRR MC	Maximum Correlation	Robust Local Regression
LRR MMT	Minimum Migration Time	Robust Local Regression
LRR MU	Minimum Utilization	Robust Local Regression
LRR RS	Rom Selection	Robust Local Regression
MAD MC	Maximum Correlation	Median Absolute Deviation
MAD MMT	Minimum Migration Time	Median Absolute Deviation
MAD MU	Minimum Utilization	Median Absolute Deviation
MAD RS	Rom Selection	Median Absolute Deviation
THR MC	Maximum Correlation	Static Threshold
THR MMT	Minimum Migration Time	Static Threshold
THR MU	Minimum Utilization	Static Threshold
THR RS	Rom Selection	Static Threshold
OPT ALGO	Proposed Algorithm Part – 1	Proposed Algorithm Part – 2

The simulation of the algorithm is based on cloud sim, which is a framework for modeling and simulation of cloud computing infrastructures and services. The experimental setup used for this work is been explained here (Table 8):

Table 8: Experimental Setup **[6]-[9]**

Setup Parameters	Number of Physical Hosts	Number of VMs	Total Simulation Time (In Sec)
VALUES	800	1052	86400.00

Copyright (c) 2017 International Journal of Computer Science and Network. All Rights Reserved.

Third, this work analyses the percentage of SLA violation during the proposed method and compare with the existing policies (Table 9):

Table 10. VM Selection Time **[4]**

Policies	VM Selection Time (In Sec)	Change((In Sec)	(Increased / Decreased)
IQR MC	0.00134	0.00089	DECREASED
IQR MMT	0.00022	0.00023	INCREASED
LR MC	0.0013	0.00085	DECREASED
LR MMT	0.00044	-	-
LR MU	0.00017	0.00028	INCREASED
LR RS	0.00104	0.00059	DECREASED
LRR MC	0.00022	0.00023	INCREASED
LRR MMT	0.00054	0.00009	DECREASED
LRR MU	0.00011	0.00034	INCREASED
LRR RS	0.00016	0.00029	INCREASED
MAD MC	0.0022	0.00175	DECREASED
MAD MMT	0.00022	0.00023	INCREASED
MAD MU	0.00027	0.00018	INCREASED
MAD RS	0.00071	0.00026	DECREASED
THR MC	0.00223	0.00178	DECREASED
THR MMT	0.00017	0.00028	INCREASED
THR MU	0.00005	0.0004	INCREASED
THR RS	0.00011	0.00034	INCREASED
OPT ALGO	0.00045	-	-

The proposed framework, demonstrates reduction of VM selection time compared to the 50% of the existing policies. Finally, the proposed technique is been tested for the LB with the below furnished simulation setup (Table 9):

6. Conclusions

LB can be achieved through VM migration. However the existing migration techniques constraints to improve the SLA and often compromise to a higher scale on the other performance evaluation factors. This work, demonstrates the optimal three phase VM migration technique with up to 70% improvement to retain SLA compared to the other VM migration technique. The work also elaborates on the VM image operability most suitable for migration and determines the best format. However the proposed technique is independent of the VM image format and demonstrates the same improvement.

The comparative analysis is been done with the proposed technique with the existing techniques like IQR MC, IQR MMT, LR MC, LR MMT, LR MU, LRR MC, LRR MMT, LRR MU, LRR RS, LR RS, MAD MC, MAD MMT, MAD MU, MAD RS, THR MC, THR MMT, THR MU and THR RS. The work also furnishes the practical evaluation results from the simulation to retain the improvement of the other parameters at least to the mean of other techniques during SLA improvement. Also this proposed technique for VM migration demonstrates no loss in existing CPU utilization during LB [9].

Acknowledgements

The authors thank to the principal and the management for constant support and encouragement.

References

[1] T. N. Y. Times. (2014). The cloud factories: Power, pollution and the internet. [Online]. Available: http://www.nytimes.com/2012/09/23/technology/data-centers-waste-vast-amounts-of-energy-belying-industry-image.html

[2] Beloglazov and R. Buyya, "Managing overloaded hosts for dynamic consolidation of VMs in cloud data centers under quality of service constraints," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 7, pp. 1366-1379, 2013

[3] H. Xu and B. Li, "Anchor: A versatile and efficient framework for resource management in the cloud," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 6, pp. 1066-1076, 2013

[4] S. Di and C. L. Wang, "Dynamic optimization of multi-attribute resource allocation in self-organizing clouds," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 3, pp. 464-478, 2013.

[5] J. Zhan, L. Wang, X. Li, W. Shi, C. Weng, W. Zhang, and X. Zang, "Cost-aware cooperative resource provisioning for heterogeneous workloads in data centers," *IEEE Trans. Comput.*, vol. 62, no. 11, pp. 2155-2168, 2013

[6] X. Liu, C. Wang, B. Zhou, J. Chen, T. Yang, and A. Zomaya, "Priority-based consolidation of parallel workloads in the cloud," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 9, pp. 1874-1883, 2013.

[7] D. Carrera, M. Steinder, I. Whalley, J. Torres, and E. Ayguad, "Autonomic placement of mixed batch and transactional workloads," *IEEE Trans. Parallel Distrib. Syst.*, vol. 23, no. 2, pp. 219-231, 2012.

[8] T. Ferreto, M. Netto, R. Calheiros, and C. De Rose, "Server consolidation with migration control for virtualized data centers," *Future Generation Comput. Syst.*, vol. 27, no. 8, pp. 1027-1034, 2011.

[9] K. Mills, J. Filliben, and C. Dabrowski, "Comparing vm-placement algorithms for on-demand clouds," in *Proc. IEEE 3rd Int. Conf. Cloud Comput. Tech. Sci.*, 2011, pp. 91-98.

Authors –

Md. Rafeeq is working as an Associate Professor in Department of Computer Science and Engineering at CMR Technical Campus, Hyderabad. He completed B. Tech, M. Tech in Computer science and engineering and Pursuing PhD (CSE) in Cloud Computing at JNTHU Hyd. He is having 12 years of teaching experience. He has 15 research publications at International/National Journals and Conferences

Copyright (c) 2017 International Journal of Computer Science and Network. All Rights Reserved.

Dr. C. Sunil Kumar did his B.E in Computer Science and Engineering from University of Madras, Vellore, India, in 1998, M. Tech in Computer Science and Engineering from SRM university, Chennai, India, in 2005 and PhD (CSE) from JNTUH in 2012. Currently, he is Professor in IT, SNIST, Hyderabad, India. He has 58 research publications at International / National Journals and Conferences. Research interests are Distributed Databases, Data warehousing and Data Mining.

Dr. N. Subhash Chandra, Professor in CSE, CVRCE. He is actively engaged in research work, six scholars are pursuing their doctoral work under his supervision. He has got more than 35-research publications to his credit including in prestigious international/ national journals. He has been on the panel of Referees for many prestigious international & national conferences. He was conferred with the prestigious Ideal Teacher and Best Teacher awards. He has been honored with Best paper award in the Map world forum, 2007.

Dr. Aruna Varanasi did her M. Tech in Computer Science and Engineering from Andhra University, Visakhapatnam in the year 2003, PhD in C.S.E, JNTU Hyderabad, awarded in the year 2014. Currently, she is working as Professor and head in the Department of Computer Science and Engineering (CSE), Sreenidhi Institute of Science and Technology (SNIST), Hyderabad, India. She was awarded "Suman Sharma" by Institute of Engineers (India), Calcutta for securing highest marks among women in India in AMIE course. She award as best faculty from Accenture and SNIST. She has 35 research publications at international/ National journals and conferences.

.

Copyright (c) 2017 International Journal of Computer Science and Network. All Rights Reserved.