International Journal of Enhanced Research in Science, Technology & Engineering
ISSN: 2319-7463, Vol. 5 Issue 5, May-2016

# Challenges of Heterogeneity in IP Networks

Tarek Heggi[1], Maryam Hazman[2], Fathy Amer[3]

[1,2] Central Lab. for Agricultural Expert Systems, Agricultural Research Center, Egypt
[3] Faculty of Computers and Information, Cairo University

**ABSTRACT**

**Providing services with different characteristic through IP network requires essential requirements to be considered: QoS, QoE, Traffic Identifications, and Measurements. QoS and QoE are essential requirements for the service provisioning whereas Traffic Identifications and Measurement are used to apply and ensure the quality of the delivered services. On the other hand, user requirements can be categorized into functional and non-functional requirements. The users and applications requirements can be achieved using either deterministic or stochastic approaches. The dynamicity in the levels of service agreement between the user and the service provider makes it difficult to guarantee services level either if there service is requested from either one service provide or through service broker that get service from multiple providers. Traffic Identification has a continuous enhancement and several open issues due to the fact that the several parameters that should be taken into consideration when developing algorithms for traffic identification such as: computational time, computational cost, accuracy, ground truth, and scalability. Measurements include multiple levels to be monitored and measured: network levels and user level. This paper shows the impact of heterogeneity in network performance and focus on the necessity to a framework for addressing the heterogeneity in services and users requirements to provide a dynamic QoS in IP network.**

**Index Terms**: **IP networks, Performance management, Quality of Service, Traffic Identification, Simulation**

## I. INTRODUCTION

This paper explores the impact of specific parameters on the network and application performance, the findings of the results approach the need for dynamicity for overcoming the heterogeneity. During the process of provisioning quality of service, we should pass through two main procedures, to identify the traffic in which a classification mechanism is applied then applying the Quality of Service Policy that is meet the requirements specification for each class of traffic. The Quality of Service (QoS) is defined in [1] as "set of qualities related to the collective behavior of one or more objects". QoS consists of a multi valued concepts that have specifications such as: reliability, performance, robustness, and cost [2]. QoS properties have two classes: functional and non-functional [3]. There many methods for deploying QoS such as Over-provisioning, DiffServ, and Multi Protocol Label Switching. Over-provisioning is one of the solutions that provide enough resources to handle QoS issues [1][4]. DiffServ QoS model works at network layer [5], whereas IntServ model works at MAC layer [6] and IEEE.11e in wireless network [7].

Multi Protocol Label Switching (MPLS) [8] was proposed to improve packet forwarding over the network. Other approaches are created to facilitated the management and improve QoS in IP home network such as the Autonomic Computing paradigm Approach [9] suggests self-properties for autonomous systems which consists of self-configuration, Self-healing, Self-optimization, and Self-protection [10] to reduce network management complexity. IP home networks): Class-based QoS and Session-based QoS [1]. In applications that are implemented using Service Oriented Architectures as e-Business, it is essential to have QoS selection service [11]. Getting Web services with highly reliable and available are increasing which leads to the need of diverse range of QoS support [12]. Accompanied with implementing QoS, there is a necessity to evaluate performance and reliability, stochastic methods as single queuing, or Petri nets are used [13]. Models that are used for assessing performance can be categorized into two classes: analytical and simulation [13]. QoS has heterogeneous constraints with different metric units; these units should be converted to unified measurement units for the purpose of monitoring and matchmaking [14].

Traffic Identification is a main component used for putting traffic into classes or clusters to be able to apply QoS, therefore, the urgent need for the IP traffic classification is question less because it plays an important role in identifying applications for applying both QoS, and recognizing attacks to prevent intrusion for the network resources. Port-based technique depends on identifying the applications into UDP and TCP IP connection oriented or connectionless flows using ports that are registered in IANA. There are limitations in this approach because of emerging of new applications with undefined or dynamic port numbers and other application that use well known ports, these limitations have been indicated by many studies and emphasize on the inefficient of the Port-based classification [15-22]. Coralreef is an example for Port-based classification technique; it is a package of device drivers, libraries, classes, and applications [23]. The CoralReef software suite is a comprehensive collection of tools developed by CAIDA to collect, store, and analyze traffic data. This technique is not suitable for the time being because most of applications don't follow the registered udp/tcp ports in IANA and many researches indicated that this techniques doesn't achieve accuracy more that 70% [19], [20], [21], [24], [25]. Payload-based technique on the inspection of the packet content to identify the application, therefore, the classifier of this technique should be able to recognize on the pattern of the application [26]. The payload technique isn't an accepted solution in many cases due to the following reasons: it is computationally expensive, it can't deal with encrypted content, it requires the packet contents, and it violates the privacy legalization [27]. It is not suitable for real time cases [28].Behavioral-based technique relied on the captured social interaction [26].BLINC [29] is an example for this category of classifiers. BLINC has a host behavior patterns that is compared to the captured profile of the host at three levels: social, functional and finally the applications. BLINC has 28 different parameters to tune [30].The key idea in this technique is that it doesn't rely on either the port numbers or the payload. This technique has some limitations such as: the inability to be used in real time environment due its classification speed, it need relevant amount of traffic flow to be able to correctly identify the application pattern [31].

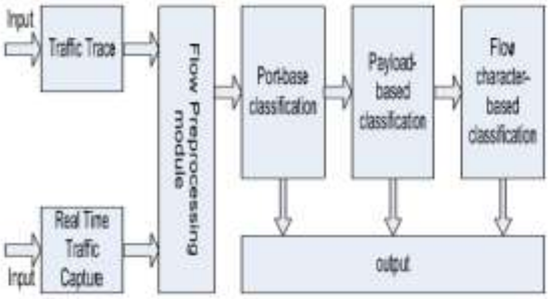


**Figure 1. Multi-level Classifier Components**

Due the limitations in port-based and payload-based, new multilevel techniques have been developed to overcome these limitations as [32] as indicate in Fig.1. The input to the classifiers may be online or offline. Machine Learning Classification techniques can train data and use data mining for several types of tasks. Weka application is an example that consists of a collection of algorithms that are used for executing classification problem. Most Weka Algorithms treats with the application classification as a statistical problem which depend on statistical feature such as: number of packets, packet size, inter arrival time The key advantage of this algorithm is that there is no need for packet payload [33]. QoS and Traffic Identification has multiple approached to be applied. To address the heterogeneity in types of both QoS, and Traffic Identification, we can use methods as Ontology, Semantic, and taxonomies.

Defining logical correlation using Semantic associations can be used in the recent heterogeneous telecom networks [34]. [1] The ITU-T Rec. X.641 ontology allows us to describe the QoS requirements of multimedia services. The author in [35] introduces a semantic QoS ontology model which avail a standard generic ontology for arbitrary QoS features**.** Ontology is a defined as a specification of a conceptualization of a knowledge domain, qualities of services in our case [13][34]. Ontology in QoS is used to provide methods to advertise offerings, to rate the services, to provide a shared and common description for the QoS which reduces the semantic heterogeneity between component descriptions, and provides reasoning mechanisms on [13][36][2]. OWL [37] is the Web Ontology Language which is based upon the Resource Description Framework (RDF), and RDF is built upon XML. OWL-S [38] is OWL ontology for describing web services.[36] QoSOnt was developed by a process of examining existing QoS specification languages [39][40]. QoSOnt represents many of the commonalities discovered between the QoS specification languages [36]. The ontologies fall into three layers as shown in Figure 2[36].

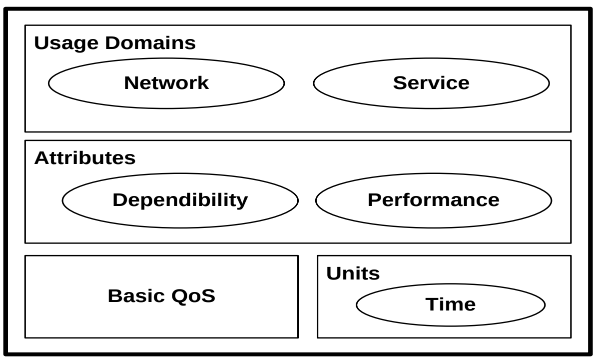


**Figure 2. Layers of Ontology**

The base QoS layer contains generic concepts relevant to QoS [36]. It also allow to define the network resource nonfunctional features with network resource ontology and integrating it into the resource profiles enables quality-aware network resource selection and composition and networking interoperability [41]. A semantic based approach for QoS driven service discovery is proposed to assist clients to select the best services [14] which are selected according to the functional requirements and dynamic Qualities of Services (QoS) requirements. There are roles in Web Service model that need to be defined using QoS ontology [42].

SLA contract includes QoS requirements and penalties between customer and service providers [43]. It includes offer, counter offer, negotiation, provisioning, and continual monitoring services [44][45]. Other authors, Buyya and colleagues, indicated aspects of Cloud as dynamic scalability, and service level agreement [43]. To maximize the resource usage, some algorithms as [46] describe a QoS brokering service while trying to meet the user-defined QoS requirements. Using of Ontologies also address dynamic service selection [47].

According to the definition [48], “A Cloud is a type of parallel and distributed system consisting of a collection of inter-connected and virtualized computers that are dynamically provisioned and presented as one or more unified computing resource(s) based on service-level agreements established through negotiation between the service providers and consumers”. Major motivations to adopt Cloud services include reasonable price as they are offered in economy of scale, and transferring responsibility of maintenance, backups, and license management to Cloud service providers. This paper is aiming at finding out the requirements for addressing the heterogeneity phenomena by designing a network setup that host three applications and study the impact of changing the application characteristics and the metrics of the QoS and envision the needs for getting a dynamic and adaptable network environment.

This paper is organized as follows. Section 1: introduce a background about the different components of to provide QoS, Section 2: the related works, Section 3: configuration scenarios, Section 4: discussion, section 5: conclusion and future work.

## II. RELATED WORK

The network configuration management in [49] is considered a complex process due to the heterogeneity in the command line interface that is developed by each manufacture. The author in [49] discusses the heterogeneity in the configuration of network management, it develop a data model using restricted form of ontology to accommodate network configuration in a vendor-independent fashion. This approach is used for alleviating the interoperability issues in heterogeneous network environment by easing out discrepancies between the command line syntax and structure of various devices [49].

## III. PROBLEM STATEMENT

Heterogeneity in computer networks leads to difficulties in providing QoS to service requesters. We will focus in three kinds of heterogeneity: network technologies, network applications, and users’ services requirements. A heterogeneity phenomenon is against scalability, flexibility, and unity. To address heterogeneity, standardization among the different parties is required to provide interoperability. Achieving adaptation to dynamic changes in the three different parameters is our concern to address the heterogeneity problem. From our reading, heterogeneity appeared in eighteen’s. The significant of addressing the heterogeneity issue has two perspectives, the user which represented in getting the best performance for the required service and the service broker will assist the user to get the correct decision in the selecting the appropriate service provider. The second perspective is the service provider which represented in using a reusable/shared Ontology will assist in achieving the required service level of quality for the customers.

## IV. PROPOSED NETWORK DESIGN

In this setup, we get the number of the staff that should use the Egyptian Universities Network (EUN) in 2014 [59][60] Egypt includes 24 public universities and according to Central Agency for Public Mobilization and Statistics (CAPMAS), the number of graduates from public universities in 2012/2013 is 325,358[60][61] the staff members in these universities is 83,375[62]. Egyptian Universities Network (EUN) was founded in 1987 and it serves staff members in all public universities [63]. EUN provides academic community with Internet using 1Gbps internet bandwidth [63].

### A. Network Design

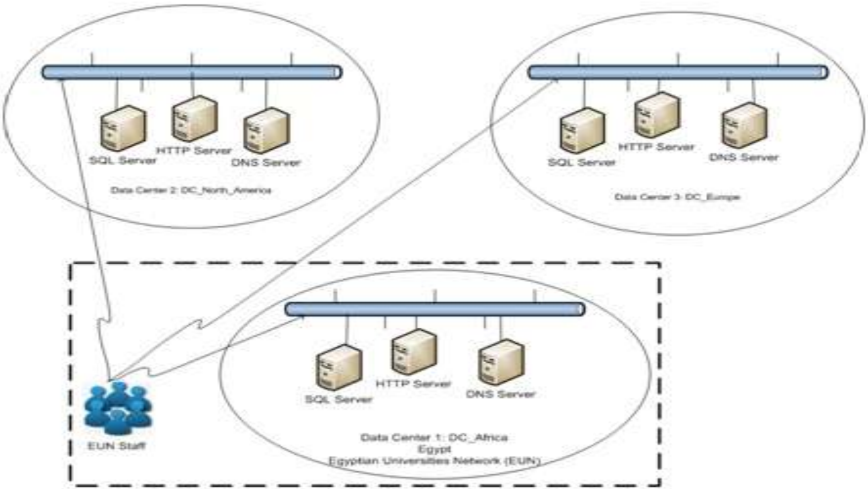


**Figure 3. Network Design Layout**

In our configuration scenarios, we will assume that we have a data center in EUN that host applications that will be used for staff members and the students in the graduation year to implement their projects and research papers that needs virtual machines with specific: hardware, platform, software applications, and Internet bandwidth. Assuming that the number of users from students and staff members that need virtual machines to develop and test their activities is 81,746, this number represents 20 percent of the total number of staff members and students. We have assumed that there three applications that will be used by the university staff which are HTTP, SQL, and DNS, the request size of the requests for each applications are based on the type of each application[65][66][67]. Our proposed network setup is represented in Figure 3.

### B. Used Methods

Cloud Analyst [64][61] is a simulator that will be used to illustrate the impact of heterogeneity in cloud environment. CloudAnalyst provides information about metrics such as: response time of requests, and processing time of requests Realizing the impact of variation in applications and the number of simultaneous users is difficult to be done in real tested, the existing of tools such as CloudAnalyst facilitate this process [64].

## V. EXPERIMENT

In configuration scenarios, the used configuration parameters are classified into two categories. Category 1 is the global constants which represent the parameters that will not be changed in all experiments. Category 2 is the constants parameters that will be changed for each experiment and it will not be change during the trails in each case.

### A. Global Constant Parameters

Table 1 indicates the constants parameters that will be used through all experiments:

**Table 1. Global Constant Parameters**

| | |
|---|---|
| Simulation Time | 24 hours |
| Number of Users on average | 80,000 |
| Maximum Number of Users on average | 20,000 |
| Physical Servers | 10 |
| User grouping factor | 80,000 user |
| Request Grouping factor: | 60 concurrent request |
| Executable instruction length | 250 byte |
| Type of load balancing | Round robin |
| Service Broker Bandwidth | 100,000 Mbps |
| Physical Machine Bandwidth | 1000 Mbps |

International Journal of Enhanced Research in Science, Technology & Engineering
ISSN: 2319-7463, Vol. 5 Issue 5, May-2016

### B. Case 1: Changing in Number of Data Centers

Table 2 indicates the constants parameters that will be used in Case 1:

**Table 2. Case 1, constant parameters**

| | |
|---|---|
| Applications | MS SQL |
| Data Size Per Request | 1048576 byte |
| Virtual Machines | 10 |

The Variables used in Case 1 are indicated as follows:

- # of data centers: 1, 2, and 3.
- Broker Policy: Time Optimization or the closest or dynamic reconfiguration.

Figure 4 indicates the results of Case 1

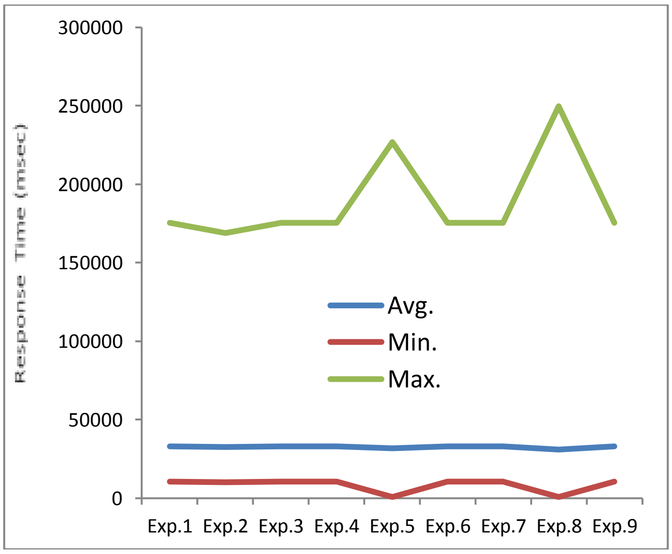


**Figure 4. Case 1 Changing in Number of Data Centers**

### C. Case 2: Changing in Number of Virtual Machines and Applications

Table 4 indicates the constants parameters that will be used in Case 2 (a), Case 2 (b), and Case 2 (c):

**Table 4. Case 2: constant parameters**

| | |
|---|---|
| # of Data Centers | 3 |
| DNS request Size | 520 byte |
| HTTP request size | 1600 Kbyte |
| SQL request size | 1048576 byte |

The Variables used in Case 2 are indicated as follows:

- # of Virtual Machines: 10,15,20,25,and 30
- Applications: DNS, MS SQL, and HTTP

Figure 5 indicate the results of Case 2 (a, b, c), in this case only DNS traffic is running

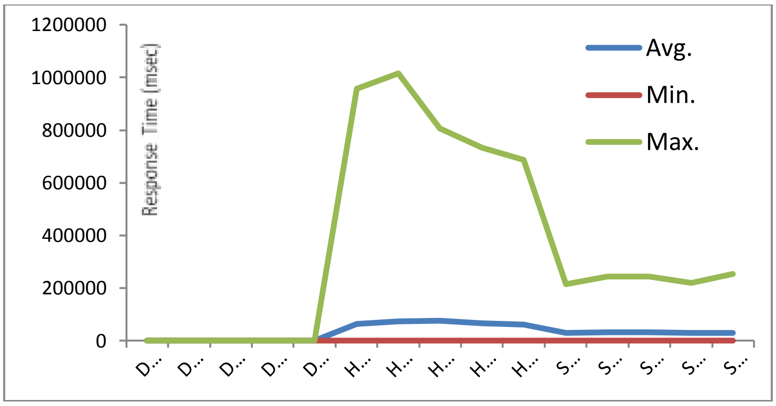


**Figure 5. Case 2, Changing in Number of Virtual Machines and Applications**

### D. Case 3: Changing in Each Data Center Bandwidth

**Table 8. Case 3: constant parameters**

| | |
|---|---|
| Applications | HTTP |
| # of Data Centers | 3 |
| HTTP request size | 1600 Kbyte |
| Virtual Machines | 10 |

The Variables used in Case 3 (a) are indicated as follows: Internal Bandwidth: 3000, 3500, 4000, 4500, and 5000 Mbps. Figures 6 and 7 indicate the results of Case 3 (a, b), in this case of changing internal and external bandwidth of the data centers.

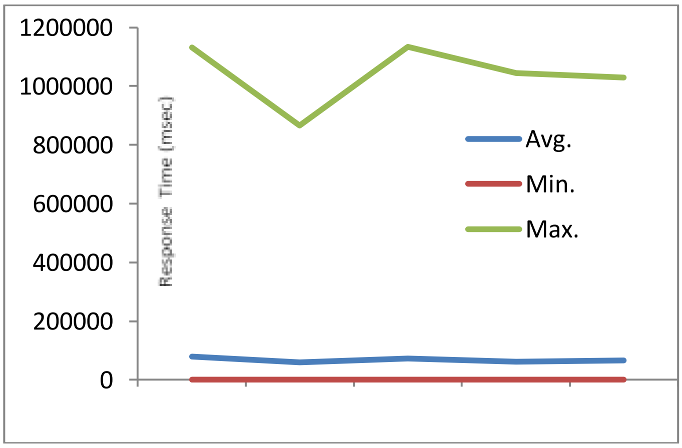


**Figure 6. Case 3(a), Changing in Each Data Center Bandwidth**

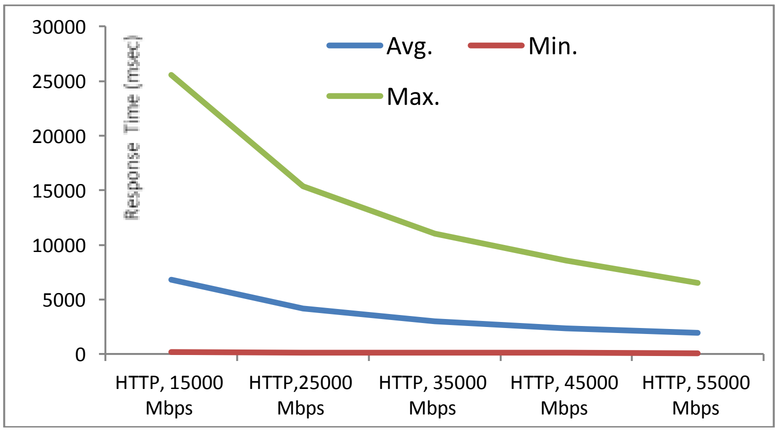


**Figure 7. Case 3(b), Changing in Bandwidth among**

**Data Centers**

### E. Case 4: Changing in Latency Time

Table 11 indicates the constants parameters that will be used in Case 4

**Table 11: Case 4 Constant Parameters**

| | |
|---|---|
| # of Data Centers | 3 |
| Applications | HTTP |
| Data Size Per Request | 1638400 byte |
| Virtual Machines | 10 |

The Variables used in Case 4 are Delay: 30, 35, 40, 45, and 50 ms. Figure 8 indicates the results of Case 4, in this case of changing latency.

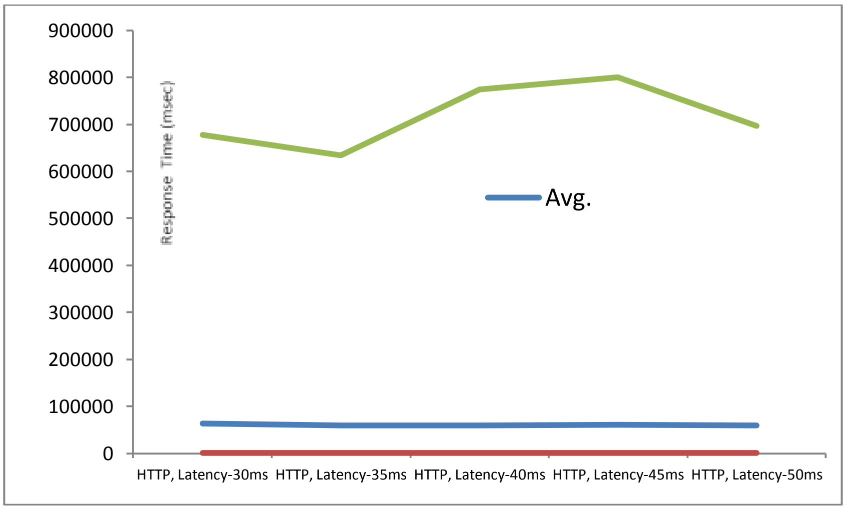


**Figure 8. Case 4, Changing in Latency Time**

## V. DISCUSSION

**In case 1:** Figure 4 indicates that increasing the number of data centers and selecting the proper service broker policy improve the average time response from 32764.80 msec to 30830.64 msec using three data centers to provide the hosting services for students in Africa region, it is also noted in case 1 that the Service Time is almost the same because there no changes in the number of virtual machine and their specifications are the same.

**In case 2:** Figure 5 indicates that for DNS, HTTP, and SQL traffic, there are very few changes with the changing in the number of virtual machines from 10 to 30 VM. The minimum average response time was 109 msec in DNS, and increased to 62588 msec in HTTP which shows the effect of application type in the increasing of response time.

**In case 3,** Figures 6,7 indicate that when changing the bandwidth concerned with each data center, from 3000 Mbps to 5000 Mbps the average response time changed from 67 sec to 62 sec which is not suitable for HTTP traffic. We have assumed that the accepted response time value should be less 2 sec. Therefore, we have run other experiments starting from 15 Gbps to 55 Gbps till we get a response time less than 2 sec.

**In case 4,** Figure 8 indicates that changing latency time which represents the time of network equipment inside each data center from 50 msec to 30 msec doesn't have a remarkable impact because the value of this latency time is small.

## CONCLUSION AND FUTURE WORK

The comparative analysis for the four cases indicates that for our assumed scenario of staff of Egyptian University Network, if there are connectivity with two other data centers and applications DNS, HTTP, and SQL. 55 Gbps is appropriate value for HTTP traffic. Case 1 indicates that the changing of the service broker policy have a great impact in traffic distribution among the different data center and improve the performance by average response time by 6 percent approx. Case 2 shows that the increasing in number if virtual machines doesn't have an impact in improving the response time. Also, case 2 indicate the application type determines the request size and this value increase the average response time value from 109.49 in DNS to 62588.38 in HTTP. Case 3 indicates that the bandwidth between the users and data center have a great impact than latency time due to the fact that the increasing in number of users increase the amount of the required bandwidth among the data centers and the users and the experiments indicate that 55 Gbps bandwidth is sufficient for an accepted HTTP response time. Case 4 indicates that latency time plays less important role in the assumed case due to the low value which is near to the reality. The overall cases indicate the necessity to have a model the heterogeneity in: the characteristic of the existing networks, the needs of applications and the user requirements. The new model will include common components and specific components. Common components represent user requirements and network applications and it consists of the essential components that are required for the provisioning of services. Specific components that belong to each provider. The main feature for this framework is to address the heterogeneity phenomena with the IP networking.

## REFERENCES

[1]. International Telecommunication Union. Series x: data networks, open system communications and security. Open distributed processing. Information technology – Open Distributed Processing – Reference model: Foundations. Recommendation ITU-T X.902, October 2009. (Cited on page 18.).

[2]. Lamia Yessad and Zizette Boufaida, "A QoS ontology-based component selection", International Journal on Soft Computing ( IJSC ), Vol.2, No.3, August 2011.

[3]. José L. Poza, Juan L. Posadas, José E. Simó, "Adding an ontology to a standardized QoS-based MAS middleware", 2009.

[4]. M. Menth, R. Martin, and J. Charzinski. Capacity overprovisioning for networks with resilience requirements. SIGCOMM Comput. Commun. Rev., 36:87–98, August 2006. (Cited on page 12.)

[5]. S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss. An Architecture for Differentiated Service. RFC 2475 (Informational), December 1998. Updated by RFC 3260. (Cited on pages 2, 6, 12, 21, 34, 130 and 134.).

[6]. R. Braden, D. Clark, and S. Shenker. Integrated Services in the Internet Architecture: an Overview. RFC 1633 (Informational), June 1994. (Cited on pages 2, 6, 12, 21, 130 and 134.).

[7]. IEEE. Standard for Information technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Specific requirements. Part 11Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications. Amendment 8 Medium Access Control (MAC) Quality of Service Enhancements. IEEE Standard 802.11e-2005, September 2005. (Cited on page 12.).

[8]. E. Rosen, A. Viswanathan, and R. Callon. Multiprotocol Label Switching Architecture. RFC 3031 (Proposed Standard), January 2001. Updated by RFC 6178. (Cited on pages 2, 6, 22, 130 and 134.).

[9]. P. Horn. Autonomic Computing: IBM's Perspective on the State of Information Technology, October 2001. (Cited on pages 6, 80 and 134.).

[10]. IBM. An architectural blueprint for autonomic computing. White paper, April 2003. (Cited on pages xi, 6, 80, 81, 82 and 134.).

[11]. Vu, Le-Hung, Manfred Hauswirth, and Karl Aberer. "QoS-based service selection and ranking with trust and reputation management." On the Move to Meaningful Internet Systems 2005: CoopIS, DOA, and ODBASE. Springer Berlin Heidelberg, 2005. 466-483.

[12]. Tian, Min, and Heinz Schweppe. "Qos integration in web services." (2003).

[13]. Marzolla, Moreno, and Raffaela Mirandola. "Qos analysis for web service applications: a survey of performance-oriented approaches from an architectural viewpoint." Department of Computer Science University of Bologna, Bologna, Italy (2010).

[14]. Li, Wen-jun, et al. "Ontology-based qos driven gis grid service discovery." Semantics, Knowledge and Grid, 2006. SKG'06. Second International Conference on. IEEE, 2006.

[15]. Gabriel Gómez Sena, Pablo Belzarena, Early traffic classification using Support Vector Machines. Facultad de Ingeniería, Universidad de la República, Montevideo, Uruguay,2009.

[16]. H.-C. Kim, K. Clay, M. Fomenkov, D. Barman, M. Faloutsos, and K. Lee. Internet traffic classification demystified: Myths, caveats, and the best practices. In ACM CoNEXT 2008.

[17]. A. Moore and K. Papagiannaki. Toward the Accurate Identification of Network Applications. In Proceedings of the Passive y Active Measurement Workshop (PAM2005), March/ Apri 2005.

[18]. S. Valenti, D. Rossi, M. Meo, M. Mellia, and P. Bermolen. A behavioral classification framework for p2p-tv applications. Technical Report WP3.1, TELECOM ParisTech (France), Politecnico di Torino (Italy), January 2009.

[19]. Thuy Nguyen and Grenville Armitage., A Survey of Techniques for Internet Traffic Classification using Machine Learning IEEE Communications Surveys & Tutorials, vol. 10 no. 4 pp. 56-76, 2008.

[20]. Ali Asghar Yarifard and Mohammad Hossein Yaghmaee ,The Monitoring System Based on Traffic Classification, World Applied Sciences Journal 5 (2): 150-160, 2008.

[21]. Karagiannis, T., K. Papagiannaki and M. Faloutsos, 2005. BLINC: Multilevel Traffic Classification in the Dark. In SIGCOMM'05, Philadelphia, USA.

[22]. Sen, S., O. Spatscheck and D. Wang, 2004. Accurate, Scalable In-Network Identification of P2P Traffic Using Application Signatures. In WWW2005, New York, USA.

[23]. The CoralReef software suite as a tool for system and network administrators (2001) by David Moore , Ken Keys , Ryan Koga , Edouard Lagache , k claffy

[24]. Chengjie Gu, Shunyi Zhuang, Yanfei Sun, Junrong Yan, Multi-levels Traffic Classification Technique, 2010

[25]. Nigel, W., Z. Sebastain,et al. (2006). "A preliminary performance comparison of five machine learning algorithms for practical IP traffic flow classification." SIGCOMM Comput.Commun. Rev. 36(5): 51-16.

[26]. S. M. Parvat, , Prof. Dr. S. D. Lokhande, Traffic Classification Technique in Computer Networks, Department of E & TC, Sinhgad college of Engineering, Pune, India
[27]. T. Choi, C.Kim, S.Yoon, J.Park, B.Lee, H.Kim, and H.Chung. Content-aware internet application traffic measurement and analysis. In IEEE/IFIPNOMS, April2004
[28]. The CoralReef software suite as a tool for system and network administrators (2001) by David Moore , Ken Keys , Ryan Koga , Edouard Lagache , k claffy.
[29]. Thomas Karagiannis, Konstantina Papagiannaki, and Michalis Faloutsos, "Blinc: multilevel traffic classification in the dark," SIGCOMM Comput. Commun. Rev., vol. 35, no. 4, pp. 229–240, 2005.
[30]. Hyunchul Kim, Marina Fomenkov, kc claffy, Nevil Brownlee, Dhiman Barman, Michalis Faloutsos, "Comparison of Internet Traffic Classification Tools",2006.
[31]. Fivos Constantinou, Panayiotis Mavrommatis, "Identifying Known and Unknown Peer-to-Peer Traffic ", Fifth IEEE International Symposium on Network Computing and Applications (NCA'06) 0-7695-2640-3/06 2006 IEEE.
[32]. Chengjie Gu, Shunyi Zhuang, Yanfei Sun, Junrong Yan, Multi-levels Traffic Classification Technique, 2010.
[33]. Nigel, W., Z. Sebastain,et al. (2006). "A preliminary performance comparison of five machine learning algorithms for practical IP traffic flow classification." SIGCOMM Comput.Commun. Rev. 36(5): 51-16.
[34]. BĂLAN, T., F. Sandu, and V. Cazacu. "ONTOLOGY BASED RESOURCE MANAGEMENT OF REAL AND EMULATED TELECOM SYSTEMS." Bulletin of the Transilvania University of Brasov, Series I: Engineering Sciences 4.1 (2011).
[35]. Afandi, Raja, Jianqing Zhang, and Carl A. Gunter. "Ampol-q: Adaptive middleware policy to support qos." Service-Oriented Computing–ICSOC 2006. Springer Berlin Heidelberg, 2006. 165-178.
[36]. Dobson, Glen, Russell Lock, and Ian Sommerville. "QoSOnt: a QoS ontology for service-centric systems." Software Engineering and Advanced Applications, 2005. 31st EUROMICRO Conference on. IEEE, 2005.
[37]. W3C, "Web Ontology Language (OWL)", http://www.w3.org/2004/OWL
[38]. DAML, "DAML Services",http://www.daml.org/services/owl-s/
[39]. Glen Dobson, "Quality/ of Service in Service-Oriented Architectures", http://digs.sourceforge.net/papers/qos.pdf
[40]. Glen Dobson, Russell Lock,http://wiki.nesc.ac.uk/read/pa9?ParametersOfQoS
[41]. Serrano, M., Foghlú, M., Donnelly, W.: Enabling Information Interoperability in the Future Internet: beyond of a SOA Design Requirement. In: FIA Session Linked Data in the Future Internet, FIA Assembly, Ghent, 15-17 December 2010, online proceedings. Available at: http://linkeddata.futureinternet.eu.
[42]. Lei Li and Ian Horrocks, A Software Framework For Matchmaking Based On Semantic Web Technology, WWW2003, May 20-24, 2003, Budapest, Hungary.
[43]. Dastjerdi, Amir Vahid, Sayed Gholam Hassan Tabatabaei, and Rajkumar Buyya. "A dependency-aware ontology based approach for deploying service level agreement monitoring services in Cloud." Software: Practice and Experience 42.4 (2012): 501-518.
[44]. Kotsokalis C, Yahyapour R, Gonzalez MAR. SAMI: The SLA Management Instance. In 2010 Fifth International Conference on Internet and Web Applications and Services. IEEE, 2010; 303–308.
[45]. Dastjerdi, Amir Vahid, et al. "CloudPick: a framework for QoS-aware and ontology-based service deployment across clouds." Software: Practice and Experience 45.2 (2015): 197-231.
[46]. Saaty TL. Fundamentals of Decision Making and Priority Theory with the Analytic Hierarchy Process, Vol. 6. RWS Publications: Pittsburgh, PA, 1994.
[47]. Dastjerdi AV, Tabatabaei SGH, Buyya R. An effective architecture for automated appliance management system applying ontology-based cloud discovery. In 2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing (CCGrid). IEEE, 2010; 104–112.
[48]. Narasimhan B, Nichols R. State of cloud applications and platforms: The cloud adopters' view. Computer 2011; 44(3):24–28.
[49]. Ngoupé, Éric Lunaud, et al. "A data model for management of network device configuration heterogeneity." Integrated Network Management (IM), 2015 IFIP/IEEE International Symposium on. IEEE, 2015.
[50]. Howarth, Michael P., et al. "Provisioning for interdomain quality of service: the MESCAL approach." Communications Magazine, IEEE 43.6 (2005): 129-137.
[51]. Google. Google app engine. http://code.google.com/appengine/.
[52]. Varia J. Best practices in architecting cloud applications in the aws cloud. Cloud Computing: Principles and Paradigms, Wiley Press, New Jersey, USA 2011; :459–490.
[53]. Zhou, Chen, Liang-Tien Chia, and Bu-Sung Lee. "QoS measurement issues with DAML-QoS ontology." E-Business Engineering, 2005. ICEBE 2005. IEEE International Conference on. IEEE, 2005.
[54]. Ardagna, D., Pernici, B.: Adaptive Service Composition in Flexible Processes. IEEE T. Software Eng. 33, 369–384 (2007)
[55]. Roy Sterritt, Manish Parashar, Huaglory Tianfield, and Rainer Unland. A concise introduction to autonomic computing. Adv. Eng. Inform., 19:181–187, July 2005. (Cited on page 82.)
[56]. Roy Sterritt. Autonomic computing. Innovations in Systems and Software Engineering, 1:79–88, 2005. (Cited on page 82.)
[57]. H. Tianfield. Multi-agent autonomic architecture and its application in e-medicine. In Intelligent Agent Technology, 2003. IAT 2003. IEEE/WIC International Conference on, pages 601 – 604, oct. 2003. (Cited on page 82.)

[58]. Kotsokalis C, Yahyapour R, Gonzalez MAR. SAMI: The SLA Management Instance. In 2010 Fifth International Conference on Internet and Web Applications and Services. IEEE, 2010; 303–308.

[59]. ElSayad, H. (2014). Reform of Higher Education Institutes in Egypt, Comparative & International Higher Education (6), 12 – 15.

[60]. http://dar.aucegypt.edu/handle/10526/4393, 2015

[61]. CAPMAS(2015).Egypt in Figures. Retrieved from:http://www.capmas.gov.eg/pdf/Egyptin Figures2015/EgyptinFigures/ pages/english%20Link.htm.

[62]. http://www.capmas.gov.eg/Pages/Publications.aspx?page_id=5104

[63]. http://wcm.portal.eun.eg:10040/wps/portal

[64]. Wickremasinghe, Bhathiya, Rodrigo N. Calheiros, and Rajkumar Buyya. "Cloudanalyst: A cloudsim-based visual 203nalyzi for 203nalyzing cloud computing environments and applications." Advanced Information Networking and Applications (AINA), 2010 24th IEEE International Conference on. IEEE, 2010.

[65]. DNS, http://serverfault.com/questions/79424/estimating-dns-bandwidth-average-size-of-a-dns-request

[66]. HTTP, http://www.websiteoptimization.com/speed/tweak/average-web-page/

[67]. SQL,http://www.info.teradata.com/HTMLPubs/DB_TTU_14_00/index.html#page/Database_Management/B035_1094_111A/appb.059.16.html