# 7 A Telecom Analytics Framework for Dynamic Quality of Service Management*

**Álvaro Guadamillas[1], Miguel Ángel López[2], Nikolaos Maravitsas[3], Alberto Mozo[1], Fabio Pulvirenti[4]**

[1]Escuela Técnica Superior de Ingeniería de Sistemas Informáticos, Universidad Politécnica de Madrid, [2]SATEC, Madrid, Spain, [3]ADAPTIT SA, Athens, Greece, [4]Dipartimento di Automatica e Informatica Politecnico di Torino

{a.guadamillas@ict-ontic.eu}, {miguelangel.lopez@satec.es}, {nmar@adaptit.gr}, {a.mozo@upm.es},{fabio.pulvirenti@polito.it}

**Abstract:** Since the beginning of Internet, Internet Service Providers (ISP) have seen the need of giving to users' traffic different treatments defined by agree- ments between ISP and customers. This procedure, known as Quality of Service Management, has not much changed in the last years (DiffServ and Deep Packet Inspection have been the most chosen mechanisms). However, the incremental growth of Internet users and services jointly with the application of recent Machine Learning techniques, open up the possibility of going one step forward in the smart management of network traffic. In this paper, we first make a survey of current tools and techniques for QoS Management. Then we introduce clustering and classifying Machine Learning techniques for traffic characterization and the concept of Quality of Experience. Finally, with all these components, we present a brand new framework

(*)This research was supported in part by project ONTIC-619633 funded by the European Commission under the Seventh Framework Programme.

that will manage in a smart way Quality of Service in a telecom Big Data based scenario, both for mobile and fixed communications.

**Keywords:** Machine Learning, Big Data, Traffic Characterization, Quality of Service, Quality of Experience

## 7.1 Introduction

Accurate identification and categorization of network traffic, according to application type, is a major task for large scale network monitoring and management aspects, with the most crucial of them being QoS evaluation, capacity planning and intrusion detection. An enormous amount, in the order of Terabytes, of data may be transferred through the core network of an ISP on a daily basis, and an exponential growth in traffic is expected, as more than 50 billion devices will be connected to the Internet in the forthcoming future. This prediction imposes a tough challenge for network data capture and analysis.

Current approaches that try to tackle the problem lack scalability and accuracy and are mostly based in empirical formulas. So this leave an open opportunity to develop an accurate and massively scalable platform for both online and offline characterization of network traffic pattern evolution is missing that should be a key element in facing the challenges. Top applications potentially benefitting from such a platform include proactive congestion control mechanisms and intrusion detection systems

To this end, the ONTIC project proposes to design, develop and evaluate:

1. A novel architecture of massively-scalable online techniques able to, at one hand, characterize network traffic data streams, identifying traffic pattern evolutions, and on the other hand proactively detect anomalies in real time at very high network speeds, i.e., hundreds of thousands of packets per second.

2. An innovative set of massively-scalable offline data mining techniques to characterize network traffic, exploiting big-data analytic approaches and cloud-based distributed computation paradigms on extremely large network traffic datasets.

The current project will attempt to integrate the above online and offline techniques into an autonomous unsupervised network traffic classification platform. Classification of traffic is essential for a proper QoS management. In addition to this classification, it is important to exactly find out which features define each class so that it is possible to know how best assign network resources and achieve a good QoS management. If more the classification of traffic does not remain static, the network will better know how to treat the traffic in different situations and both the user experience and the network management will improve. This is the target of the proposed framework in this paper, which will use QoS principles to improve the traditional QoS management. Our proposed framework solution will combine traditional techniques and has to keep in mind that (1) traffic pattern will keep in constant evolution and (2) the amount of traffic will be raised in time. Latest studies on traffic characterization are focusing on ML algorithms to face traffic pattern evolution [1], and Quality of Service Management needs to move to these ML algorithms to evolve its traditional out of date techniques. In addition, traffic needs to be handled as Big Data, and the new architecture shown in this paper must meet scalability and parallelization requisites, as we are going to see in the proposed framework.

## 7.2 Related Work

Traffic characterization in QoS/QoE environment is increasingly gaining importance, as noticeable from the massive number of works on these topics. Traffic characterization is already the core of many fundamental network operation and maintenance activities, such as the enforcement of Quality of Service guarantees or Traffic Engineering, and it has the potential to solve difficult network management problems and security issues, such as intrusion detection and prevention. In addition, the explosion of the IP-based applications and services, and, on the other hand, the multitude of access networks from which to choose, made the experience quality perceived by the users a primary subject of interest for the ISPs. Facing this kind of quality competition, the concept of Quality of Experience (QoE) emerged, combining user perception, experience, and expectations, with non-technical and technical parameters such as application and network level quality of service (QoS).

Furthermore, the boom of mobile devices, content, server virtualization, and advent of cloud services drove the networking industry to re-examine traditional network architectures and made Big Data an emerging hot topic. Existing network architectures were not designed to meet current requirements, hence, Software Defined Networking (SDN) is spreading, promising to be dynamic, manageable, cost-effective, adaptable, seeking to be suitable for today's and future applications. One of the main reasons, and, at the same time, effects of SDN diffusion is Big Data explosion which made conventional analytics, such as traditional Machine Learning and Data Mining techniques, unsuitable.

### 7.2.1 QoS architecture on Internet and UMTS

Quality of Service has traditionally focused its efforts in giving a differentiated treatment to the traffic that is traversing the network. In order to determine which kind of traffic deserves better or worst treatment, the QoS management strategy has evolved according to time and needs of both users and Internet Service Providers (ISPs) [2].

Since the beginning of Internet, Quality of Service management has been based on static classification of the traffic with no assumption that traffic may change. One of the first most popular architectures defining a QoS field was the IP architecture with the inclusion of the field Type of Service, originally defined in RFC 791 [3]. This RFC uses 3 bits (defining ”precedence type”) to divide traffic in 8 predefined classes with different discard priority each in case of congestion.

However, the 8 possible IP precedence types were aimed to ARPAnet needs, which originated the protocol under the Department of Defense, so the use if this field was not intended to be commercial. The Integrated Services architecture (IntServ), defined in RFC 1633 [4], allows the flows of an end-to-end service making a reservation of resources in each router of the path. This reservation (made through RSVP protocol) guarantees the QoS needed for the service, but every device along the path needs to maintain state information for each reservation, what makes scalability with hundreds of thousands of flows become an issue.

Due to the disuse of the Type of Service field, in RFC 2474 [5] the definition of this field was entirely changed in order to define the new Differentiated Services architecture (Diff-

Serv). In DiffServ, the differentiation of the traffic became customer-oriented. Although this architecture keeps backward compatibility with IP type of service, DiffServ uses the IP Type of Service header (6 bits) as Differentiated Service (DS) where the upper 6 bits contain a value called the Differentiated Service Code Point (DSCP).This DSCP is used to differentiate traffic in 4 classes with 3 different drop priorities each. The classes are used in the core of the network to divide the available link bandwidth so, for instance, a class with assigned 40% of available bandwidth of the link can be defined by an ISP for customers that pay more for their services. DiffServ also defines the possibility of pre-processing the packets at the edge of the network by controlling customers' traffic through a Service Level Agreement (SLA) document. This SLA will contain user contracted service rates and will let ISP, for example, drop packets in case that traffic form a specific user is exceeding the contracted wideband. Lastly, DiffServ also defines routing policies for each packet in the different nodes (routers) that implement DiffServ architecture. This is known as Per Hop Behavior (PHB) and the RFC distinguishes 4 PHBs: the Default PHB specifies that a packet marked with a DSCP value (recommended) of '000000' gets the traditional best effort service; Class-Selector PHB use DSCP values of the form 'xxx000' (with x as 0 or 1) to preserve backward compatibility with the IP-precedence scheme mentioned above; Expedited Forwarding (EF) PHB is the key ingredient in DiffServ for providing a low-loss, low-latency, low-jitter, and assured bandwidth service, giving to the packets a guaranteed bandwidth service (the recommended DSCP value for EF is '101110'); Assured Forwarding (AF) PHB uses the defined DiffServ classes to give different forwarding assurances as explained before. DiffServ has been widely accepted by ISPs and latest researches of this architecture are focusing in the implementation of DiffServ into virtualized network architectures [6].

In mobile communications, QoS management plays also an important role. UMTS is the standardization of the access to the network for mobile devices produced by the 3rd Generation Partnership Project (3GPP). As its study scope goes from the mobile device until the edge of the core, the QoS differs from the mentioned so far: it cares more about the quality of the radio communication. Nevertheless, the same principles of DiffServ can be found in UMTS Management of QoS: architecture capable of providing different levels of QoS (by using UMTS specific control mechanisms); mechanisms to allow efficient use of radio capacity (in DiffServ case, bandwidth capacity); and UMTS also meets the QoS requirements contracted by the user. UMTS ETSI TS 127 107 [7] defines 4 classes with different QoS requirements, where the main distinguishing factor is how delay sensitive the traffic is:

- Conversational class, meant for traffic which is very delay sensitive and is intended to carry real-time traffic flows, like video telephony.
- Streaming class, a bit less delay sensitive than conversational, but it is used too for real-time traffic flows.
- Interactive class is intended to a user (either human or machine) who is on line requesting data from remote equipment (such as web browsing, data base retrieval or server access).
- Background class is aimed to traffic where users and receivers exchange traffic in the background.

## 7.2.2 Traffic characterization

With growth of Internet users, Internet Service Providers have seen increased their need of a deeper understanding of the composition of traffic [2]. By having a wide knowledge of the traffic, it is possible for ISPs to achieve important tasks as re-allocation of network resources, improve capacity planning and provisioning or improve fault diagnosis. These network activities will obtain best results if traffic is properly associated to source applications or services, and a complete understanding of these sources has been achieved. This is the reason why traffic characterization is important for a good management of Internet and, particularly for us, a best optimization of Quality of Service Management.

Traditionally traffic characterization techniques have been based on the assumption of a specific traffic pattern. First uses of TCP/IP Internet (after DARPA's initial experiments) were intended to communicate companies and universities with simple services as mailing, remote access or document sharing. Later on, with the expansion of Internet for domestic use and web 2.0, HTTP traffic became popular [8]. The availability of broadband user connections spread the use of P2P, multimedia traffic and cloud services [9] [10]. Lately, the popularization of wireless access technologies is increasing the use of services such as Voice over IP or Skype. In other words, the continuous evolution of Internet has led to a complex existence of traffic patterns in the networks that still needs to be completely understood [2].

The constant change in traffic has resulted in different traffic characterization techniques. IANA registered ports has for long been used as traffic classification technique for application recognition. However, many applications use unpredictable or unregistered port numbers and the inevitability of IPv4 address exhaustion has motivated port address translation used by Network Address Translators (NATs) [2][11].

When port detection for traffic classification became inaccurate, Deep Packet Inspection (DPI) techniques were popularized. This technique uses application payload to associate flows with application, what means that payload is visible (there is no encryption of the packet) and the application protocol can be interpreted. It is easy to detect the problems of DPI: customers may use encryption or tunnelling for their communications, and application protocol interpretation has to be updated with every new version of each application protocol. In addition, some countries may impose privacy regulations to ISPs, and the costs of analysing every packet in a high-speed link impose a heavy operation load [1] [2] [11].

### 7.2.3 Machine Learning algorithms

QoS and, more generally, Network traffic classification issues have been object of attention of many researchers in the last years. A lot of them, for many reasons, developed and adapted machine learning and data mining techniques to this environment.
Machine learning algorithms aims at extracting previously unknown interesting information, such as dependencies (frequent itemset mining and association rules), groups of data objects (clustering algorithms) and categories of new observations (classification and regression).

Association rule mining is a data analysis method able to discover interesting and hidden correlations among data. It consists of two steps: (i) Frequent Itemset Extraction, in which all the frequent patterns are mined from the input dataset and (ii) Association Rules generation.

An itemset is frequent if the number of transactions in which it appears is over a minimum support threshold. Association rules are extracted from frequent itemsets and highlight correlations among items.
Unsupervised learning, instead, aims at discovering hidden structures in unlabeled data. The most popular algorithms are the Clustering ones: their target is to group sets of objects in such a way that objects in the same groups (clusters) are more similar to each other than to those in other groups (clusters). Finally, Supervised Learning is the learning task of extracting a function (or a model) that best approximates the distribution of the input dataset. This kind of algorithms usually work on a labeled sample of the input data (training set) and produce an inferred function which can be used for mapping (label) new examples.

These techniques have been strongly involved and adopted in QoS analysis and Network traffic classification. The two topics are correlated because all QoS schemes have some degree of IP traffic classification implicit in their design. In fact, both Diffserv and Intserv expect that routers are able to recognise and differentiate fine-grained classes of traffic. Moreover, real-time traffic classification is the core-component of automated QoS architectures [12].

Hence, traffic classification importance has increased in the last years and, because of the key role of information mining in this scenario, many researchers have adopted machine learning techniques. Also, both the growing number of services of their classes and the size of the network has pushed towards the spread of data mining approaches.

In [13], classification algorithms as Nearest Neighbours have been used to build a signature-based framework for mapping QoS Classes, while [14] and [15] have exploited Genetic algorithms and Naive Bayesian classifier for the same issue. Some Call Admission Control (CAC) schemes, mechanisms providing QoS by limiting the entry of traffic at the edges of the network, have been implemented by means of Neural and Bayesian Networks [16]. Finally, [17] have exploited Support Vector Machines and Decision Trees to evaluate and build Quality of Experience prediction models of Multimedia Streams. In a less specific field of research, like general traffic classification, we have witnessed a huge number of works using the most various Machine Learning techniques. Clustering algorithms have been exploited in [18],[19],[20] and [21], in order to group unlabeled data in traffic classes, often connected to applications. Classification algorithms, instead, have been adopted in [22],[23] and [24]. Precisely [25] focuses on real-time classification. Finally, also Association Rules and Frequent Itemset Mining have been employed as an effective tool to highlight and summarize the most important features of large datasets ([26], [27]).

In the last years, applying machine learning techniques in the network context has often entailed working on huge amount of data like network traffic datasets (e.g. Tstat [28], NetFlow [29], etc.). These types of databases are often so large that they are a typical example of Big Data. In these cases, traditional approaches are starting to show their limitations. Furthermore, the shift towards horizontal scaling in hardware has highlighted the need of parallelization in Machine Learning techniques. Hence, recently, an increasing number of researchers have adopted the MapReduce programming paradigm. Designed to simplify the processing of large databases, MapReduce main idea is to share the processing of the data in independent parallel tasks; Hadoop [30] is one of the most widely diffused MapReduce framework. We can find an increasing number of MapReduce implementations of the most important Machine Learning algorithms.

The Frequent Itemset Mining problem has been addressed by [31] and [32]: these works present distributed implementation of the most popular Frequent Itemset Mining algorithms. Even Clustering problems have been addressed by many researchers in the last years. [33] and [34], have focused on K-means-based [35] distributed implementations while [36] takes particularly into account data exchanged between nodes.

The most important milestone among the distributed classification algorithms, instead, is certainly Planet [37]: they developed a MapReduce implementation of a Regression Tree in the context of predictions over ads.

### 7.2.4 Quality of Experience

In the recent past, big networks have been tested in an objective and tangible way by taking into account a number of variables that determine the network. As we've seen in the previous sections, such measures compose the Quality of Service (QoS) of the network. The term QoS refers to the ability of the network to achieve a more deterministic and is crucial for the minimum-required quality delivery in many ways, including data transported with a minimum packet loss, delay, maximum bandwidth, minimum and stable jitter and so on. The aggressive adoption of QoS-ensuring formulas in network monitoring and management, occurred simply because it was cost-effective and was enough for the Internet services of that time.

With the explosion of information and real time Internet application, QoS management is not enough to ensure end-user satisfaction. One could argue that QoS does not consider the user's perception. This important gap, is filled with techniques that take into account the user's opinion and compose Quality of Experience (QoE).

The QoE is an end-to-end subjective metric that involves human influence factors. It combines and bonds together user perception, user expectations, and QoS measurements that result in the experience of the end application and network performance. It essentially describes the satisfaction of the user in respect with the service . To that extent ISPs and service providers need to know whether their network is delivering the right QoE to the end users. They need to integrate QoE paradigms into their traffic management system. This is why the empowerment of more thorough understanding of quality as perceived by end-users is receiving ever increasing attention from network operators and large organizations.

Developing and implementing a QoE-aware system requires passing over several challenges, as QoE relies on both complex non-deterministic human metrics and technical factors such as:

- Context, which can include access type, movement (mobile/stationary), location, environment, social and cultural background, purpose of usage

- User factors such as expectations, requirements, perception, psychological factors etc.

- Application factors such as Application type, for example Web access, VoIP, streaming stored video and media, live video streaming

- Application QoS measures such as initial delays, video (stalls), video buffering strategies, transport characteristics such as UDP, HTTP/TCP

- Network-level QoS such as delay, and jitter, bitrate, packet loss, etc.
- Content, such as video codec, format, resolution, duration, content and type of video.
- End-device characteristics such as , device performance, user interface
- Service characteristics such as reliability, availability, etc.

Interestingly, QoE can be approximated under both subjective and objective approaches. Heavy user involvement is substantial for subjective measurements as required by the ITU-T standard Mean Opinion Score (MOS), where the users are asked to quantify the quality using a 5-point scale scoring system in the form of 5-Excellent, 4-Good, 3-Fair, 2-Poor, 1-Bad. Realistically subjective tests have to go further than that. To increase accuracy server other techniques are uses such as crowdsourcing. Crowdsourcing requires the installation of special software on end-devices that measure several aspects of user behaviour during the usage of the service. In the case of video streaming it is interesting to investigate how many times and at what point the user has stopped the video due to bad quality, has restarted the video due to jitter etc. After that studies can be conducted on laboratory environment where the validation of the crowdsourcing test results and the filtering of unreliable user ratings takes place.
Objective QoE measurements estimates the QoE with parametric models, with little to no user involvement. In that case the parametric model is usually a function of the network-level QoS . It is obvious that the complexity of this task rises as a QoE-aware system has to define (a)which factors influence the QoE and (b)accurate mapping between QoE measurements and QoS metrics. The functions that describe the mapping of Network-Level QoS to QoE can behave differently for different types of applications. Such functions can be [38]: linear, exponential, logarithmic, power.

There are several QoE/QoS correlation models for Internet Services in recent studies. One of the most used ones is the IQX hypothesis In [39], the authors assume a linear dependence on the QoE level following the differential equation:

$$\frac{\partial QoE}{\partial QoS} \sim -(QoE - \gamma)$$

The above equation is solved by a generic exponential function called IQX hypothesis, which suggest that QoE and QoS are related in this manner:

$$QoE = a*(exp(-b*QoS)+c)$$

where a,b,c are positive parameters and have to do with several application settings. In this study, the QoE has been considered as the MOS, while the QoS has been represented from variables like packet loss, jitter, response and download times. In [40] there is a significant example that focuses on YouTube QoE and utilizes the IQX hypothesis with results shown in 7.1.

$$f(x) = \alpha e^{-\beta x} + \gamma \text{ or } QoE = \alpha e^{-\beta QoS} + \gamma$$

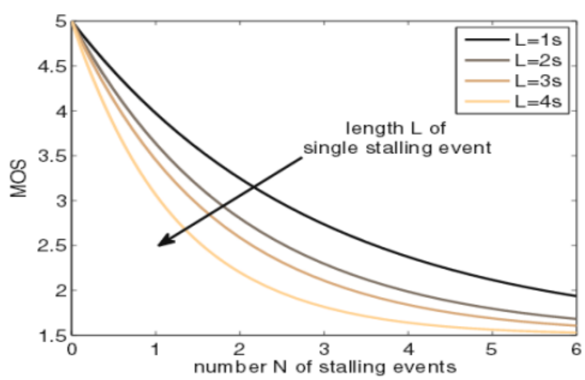


Figure 7.1: YouTube MOS to QoS mapping

In [41] a more holistic objective approach is proposed that considers a multidimensional QoS/QoE mapping for video applications, where the QoE is measured by the Video Quality Metric (VQM) [42]. Thus, the VQM is of n-dimensional a function of several QoS parameters.

$$VQM = f(x_1, x_2, x_3, x_4, \ldots, x_n), x_i \text{ is a QoS parameter}$$

In [43] a method that only relies on limited user involvement is considered. Viewers were presented with the a video sample where the testers alter the video quality. The users where asked to point the moment where the change of quality became noticeable by using the method of limits [44]. After that, several statistical analysis methods where used such as Discriminate Analysis, to produce a prediction model that maps QoS changes to QoE changes. The aforementioned methods were evolved in more recent studies, such as in [45] in which QoE models were rebuilt to more complete prediction systems, utilizing Machine Learning methods, such as of Decision Trees [46] and Support Vector Machines (SVM) [47]. A very appealing work is done in [48] where SVMs and DTs are used to create a QoE model and are compared against other Machine Learning methods like Naive Bayes, kNearest Neighbours, Random Forest and Neural Networks.

## 7.3 A Telecom Analytics Framework

We propose a framework that will make use of ML clustering and classifying algorithms in order to characterize traffic in real time. The characterization of traffic will evolve in time according to traffic patterns and, with the information acquired per class of traffic, the treatment in every packet entering in real time to the core of the network will fit the needs of both users (e.g. give preference to the traffic most important at that moment for a customer) and network (e.g. if congestion is detected, incoming traffic will have different dropping precedence in order to less contribute to a worse congestion). ONTIC framework will be focused on the core of the ISP network, so we will consider that traffic arriving is originated both from Mobile and Fixed networks. 7.2 shows a high level description of the main modules that will contain ML algorithms for QoS/QoE dynamic management and traffic characterization mechanisms.

According to this figure, the first module receiving traffic is called Enforcement Point (EP), a node that will do real-time processing of incoming traffic. As we may assume, this module will be in charge of classifying the traffic for QoS/QoE management and make a first treatment of the traffic depending on (1) the characteristic of the traffic, (2) the state of the network (i.e., if it is congested at that moment or not many network resources are available) and (3) the profile of the users that are using the network.

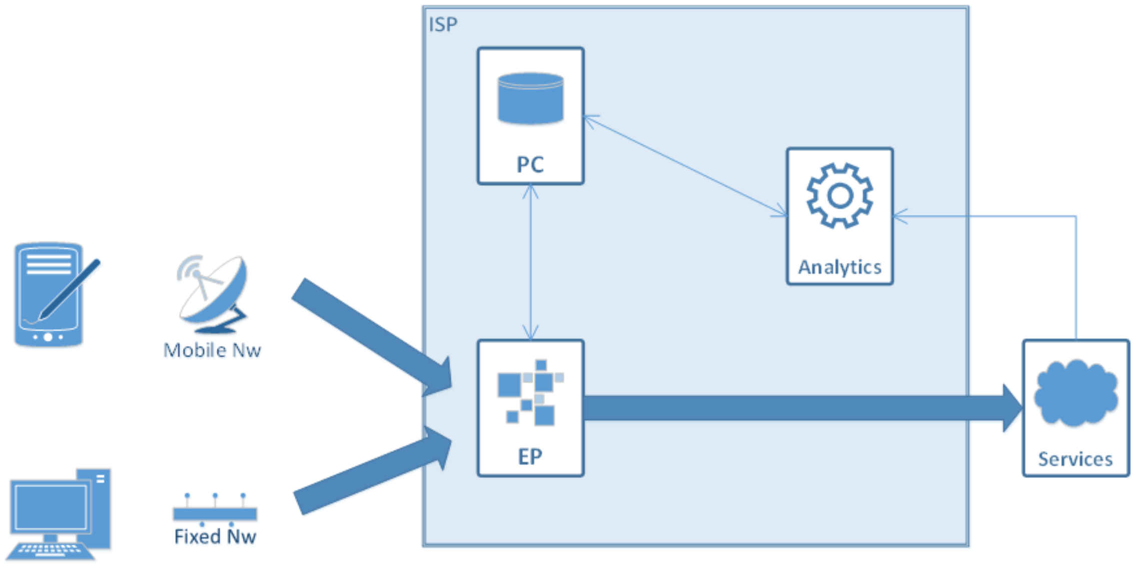


Figure 7.2: ONTIC analytics framework

For EP to gather enough information to make decisions on the traffic, it will make use of Policy Controller. This PC node will contain the results of the Machine Learning algorithms contained in the third node, the Analytics node. PC will hold (1) the information that classifies the packets in flow aggregates with similar characteristics and (2) the policies to apply in each flow aggregate in the different situations of the network that EP module can find.

Finally, Analytics module is the node in charge of performing ML algorithms on the traffic. This will be the novelty in the framework: Analytics module will generate knowledge by studying (1) the traffic that has traversed the network (both stored in a Database or in real time) and (2) studying how users of the network use their traffic. The conclusions of these studies will be delivered to PC module so that actions performed by EP are continuously up to date.

In the next sub sections we are going to see in deeper detail each one of the modules.

### 7.3.1 Analytics

As we have already seen, this module will use ML to generate for the framework updated knowledge about the traffic and the users of the network. 7.3 shows the main modules to achieve this knowledge.

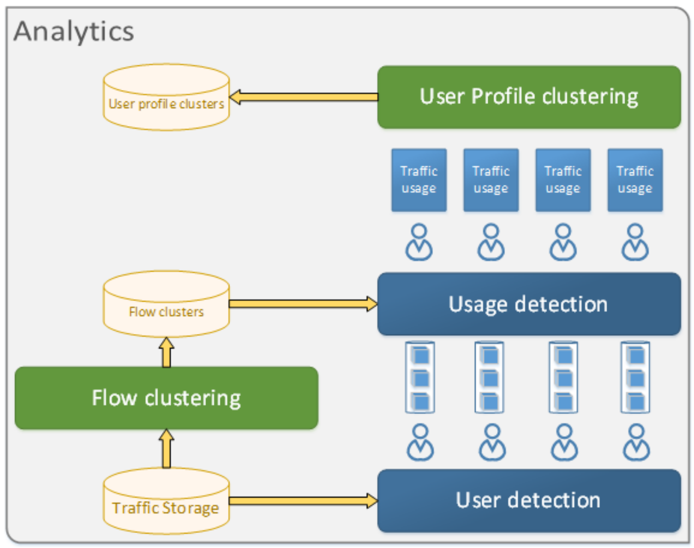


Figure 7.3: Analytics module architecture

The source point of the module is the traffic that is going through the network if the algorithms are doing online analysis or a database with the records of the packets if it is offline analysis. In both cases, they are represented by Traffic Storage. With this network traffic, the node follows two different paths: one for flow clustering and another for user profile clustering.

The Flow Clustering aims to achieve a clusterization of traffic so that flows remain grouped by flow parameters with direct implications on QoS such as minimum, mean, maximum or standard deviation of packets length, inter-arrival times or duration of the flows. Tstat tool [28] can extract a big amount of flow features that could be included for flow clustering and the survey made in 2008 [1] makes a selection of the most useful features used for this task.

However, we will need to study which features best group the flows for QoS management purpose.

The other use of traffic will serve to detect user profiles. A first step performed by User Detection module will determine which packets belong to which user of the net; once the different users are grouped with their traffic, Usage Detection module will create relations between flow clusters and user flows. The result of these relationships will be a vector per user containing weights of usage of traffic clusters called ”Traffic usage” in the figure. In this way:

$$\text{User } j \text{ traffic usage} = [w_{c1} w_{c2} w_{c3} \dots w_{cn}]$$

Where $w_{ci}$ is the weight of the usage of the cluster $i$ and $\sum_{i=1}^{n} w_{ci} = 1$.

Finally, once it is known how users make use of the clusters, we can use ML algorithms in the User Profile clustering to create user profiles.

### 7.3.2 Policy Controller (PC)

Policy Controller will create the instructions for EP to best treat traffic in every situation. It will use both flow and user profile clusters from Analytics module and generate classification and shaping rules for EP node.

When both flow and user profile clusters have been obtained, it is important to create associations between both sets of clusters. Initially, two sets of rules have been defined: (1) classification rules, the will help EP to group flows with similar characteristics; and (2) shaping rules, that will help EP to decide how to best let traffic enter in case of different situations in the network (e.g. congestion detection or major popularity of certain users in the network). PC Rules Generator module will be in charge of this task, and it is intended to be a manual process at the beginning of the development and turn automatic when enough experience is acquired.

The most important aspect to determine in this module is how often the rules should be updated. To decide this, a study on how often patterns change both in traffic and in user profiles can help to obtain conclusions.

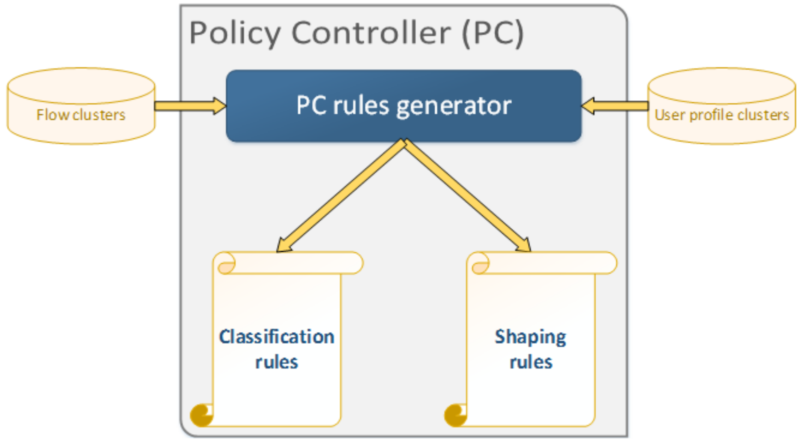


Figure 7.4: Policy Controller architecture

### 7.3.3 Enforcement Point (EP)

Finally, in this architecture Enforcement Point will be the element that will take real-time decisions on the traffic, with help of all the information gathered in both PC and Analytics modules. 7.5 shows the main modules that are part of the EP architecture.

As we can see in the architecture, the packets will first arrive to flow detection module. Here, there will be an association between the packets and the current flows they are part of. From this point, the system will treat each packet as a part of a certain flow.

Once the packets are related to flows, the classifier and marker module will make use of classification rules from PC to determine to which cluster that packet will belong. We have seen that each cluster will be characterized by features directly related to QoS, so this classification is important to properly treat the packet in the next step. With the classification, the DSCP of the packet will be changed to indicate which class the packet is part of. This way, in the core of the network the packet will be able to receive appropriated treatment too.

Finally, shaper module will make use of meter and shaping rules to decide which treatment the packet must receive (e.g. if the packet must enter the network, must be buffered or must be dropped). As the decisions must be taken per flow, the meter will take measures of every

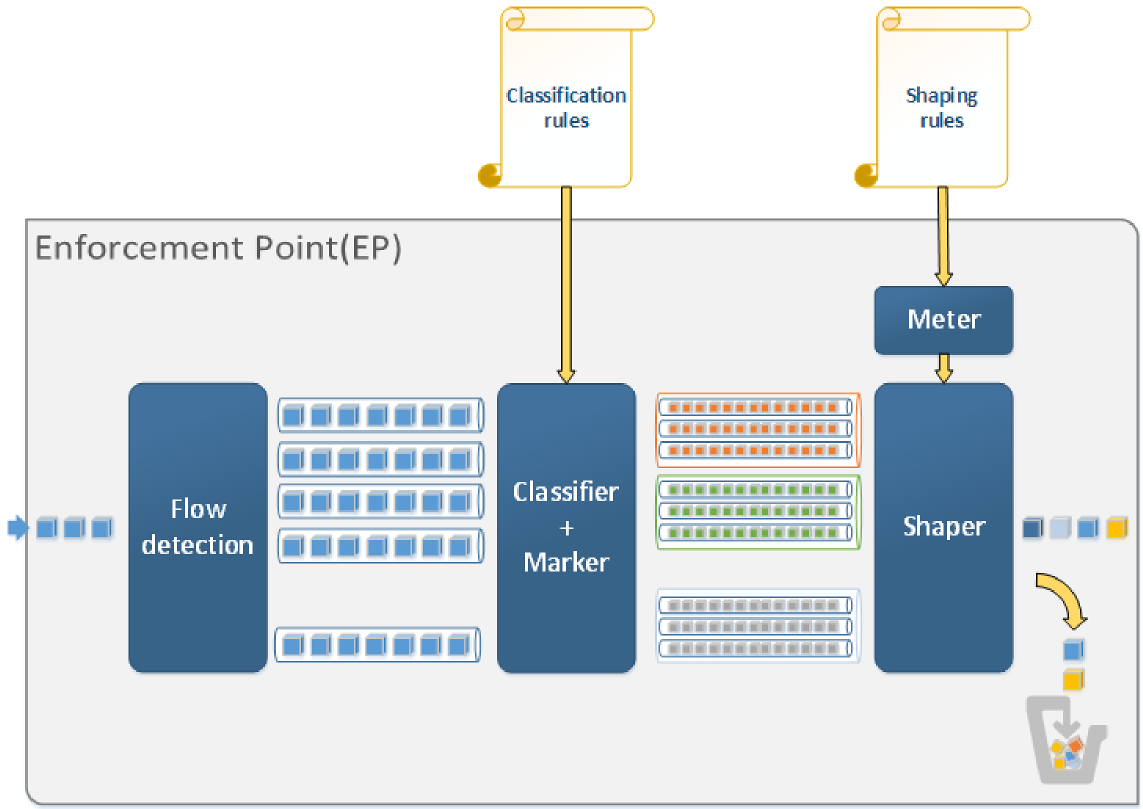


Figure 7.5: Enforcement Point architecture

packet and associate these measures to the flow, so that it can have a control, for instance, of how many packets of one flow have passed through, what max, min or mean inter arrival time currently exists in that flow or the wideband associated.

By combining both meter measures and shaping rules, the packet enters to the network (or not) by accomplishing a balance between user satisfaction and network capacities in every moment.

## 7.4 The Process Architecture for Online and Offline Scalability

The proposed implementation model is based on three main criteria: the type of processing to be performed, the nature of the input data and the need for system scalability.
Taking into account the description of the scenario, detailed in Section 3, the above criteria are particularized as follows:

- The "Enforcement Point" process is executed for each packet coming from the network, should be run online and has to be scalable in terms of maximum network packet rate (this rate is expected to grow in the future). Therefore this process requires a real-time implementation with an execution cycle equal or less than the minimum time between consecutive packet arrival $\left(T_{exec} \leq \frac{1}{v_t}\right)$.
- Although the "Analytics" module also processes packets arriving from the network, this may need to consult old packages, so it will take an offline model in which scalability should be achieved in terms of volume of input data (data which will be stored in a repository).
- In both cases the processing of the packets has a sequential structure or pipeline of tasks.

### 7.4.1 Development and Implementation Model

In order to carry out the implementation of the scenario, according to their characteristics (sequential input data and pipelines of tasks) and needs (scalability and real-time execution), the use of a distributed real-time computing environment is necessary. This environment will have the following features:

- Capacity to continuous execution of sequential input data (input data streaming).
- Capacity of parallel execution of both processes (for example independent JVMs) and tasks (Threads).

- Must ensure the processing of every input data.
- It will provide the opportunity to design each program as a set of tasks connected with each other through input and output data connectors. This topology of tasks, at the same time, will be deployed automatically as a hierarchical structure formed by: processes that can be duplicated and can be executed in parallel on different machines of a cluster of servers, threads that allow for parallel sequences which share the execution space and finally tasks the same type that runs within each thread. The number of each of these three components (processes, threads and tasks) may be configured to increase the parallelism of the system designed according to their needs in terms of both number of processes and threads and the distribution of data between the parallelized tasks.

### 7.4.2 Implementation of the Scenario

The design of the process architecture for the "Dynamic QoS" scenario, as shown in 7.6, presents an structure of two main processing modules (Enforcement Point (EP) and Analytics) and an additional module called "Policy Controller" considered as a repository of the classification rules and which must be synchronized with the other two modules because the classification rules are produced in the Analytics module and are used in the EP module.

This architecture is presented in an integrated way in order to facilitate the execution of the two processing modules simultaneously, although the Analytics module runs in Offline mode taking data from a traffic repository while the EP module is running in online mode taking the packets directly from the network. The advantage of this parallel execution is that it gets the update of the classification rules in the EP module as frequently as possible.

Regarding the internal design of the modules, it is important the decision to split the Analytics module into three processes: Rules Generator Process, Flow Characterization Process and User Characterization Process. These processes are executed in parallel and synchronously by updating of the flow clusters.

The management of the input data is performed through the components called "injectors". These components include the required logic to maintain the sequential model of data arrival (streaming) and the logic necessary to group these data and distribute them to the tasks.

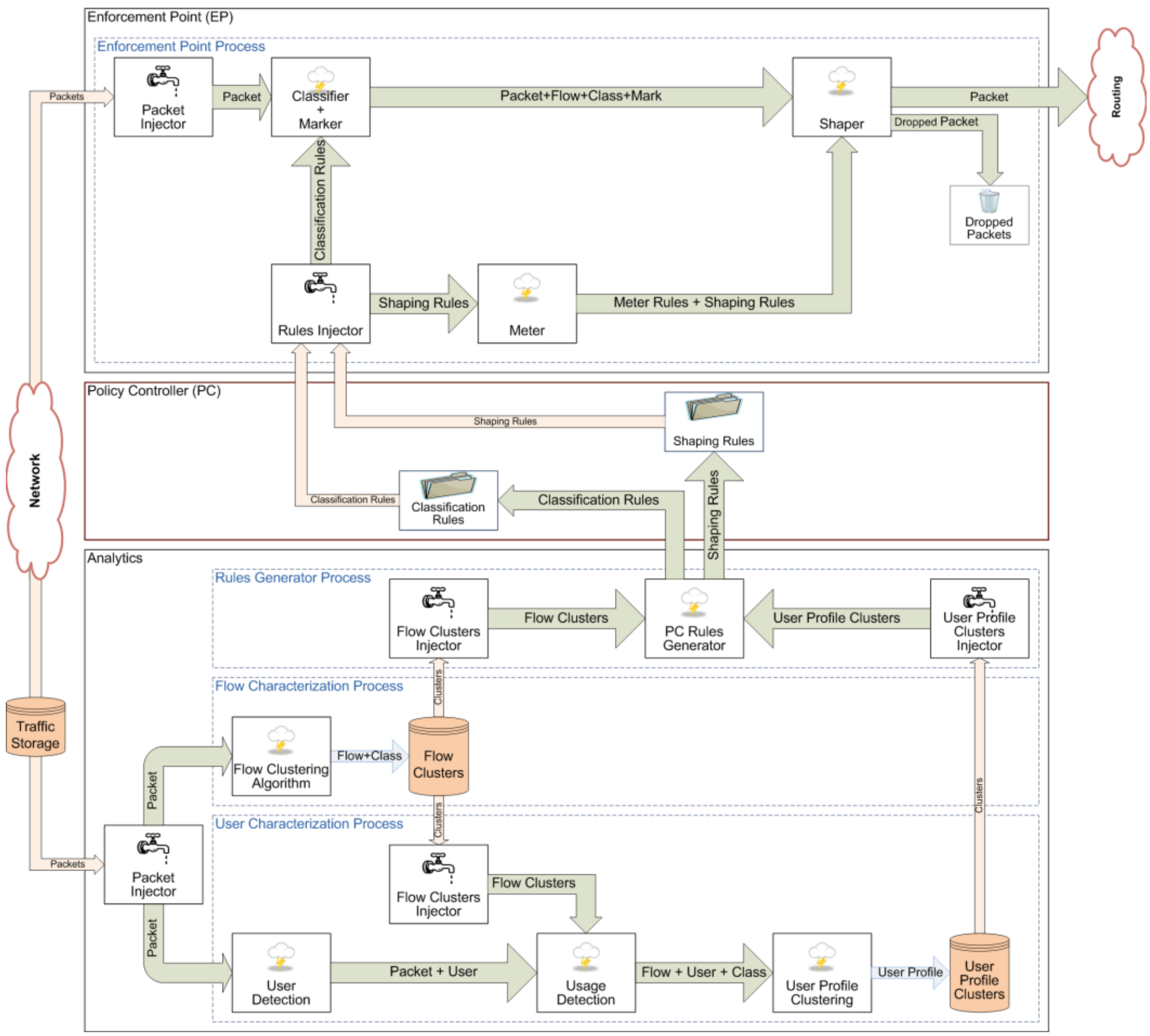


Figure 7.6: Architecture of Dynamic QoS scenario (modules and processes).

The execution components are each of the tasks defined within a process (Flow Clustering, User Detection, Classifier + Marker, etc.) and it work synchronously driven by input data coming either from an injector or from another task.

Finally, once the structure of processes and tasks has been designed, the development environment will allow a deployment configuration which will provide the scalability that the system needs. This configuration will allow the deployment of the same process on different nodes of the cluster available and, within each process, will allow the specification of a structure of threads which execute the tasks in parallel following different models which may vary from a simple load balancing (using round robin) up to the generation of data sets and the distribution of them to the different threads.

As an example for the deployment of the "Rules Generator" process (as shown in 7.7) we can define two different threads for the two data injectors (Flows injector and User Profiles injector) followed for sets of new parallel threads that process the input data from its injector (with an scalability according to the arrival rate of the data). Thus, if the system knew the number of clusters in each time, it could create the same number of threads for "rule generation" task, or otherwise, performing a static configuration of threads in accordance with the characteristics of the execution nodes. Moreover, taking into account the possibilities of data grouping provided by the environment, the entire process could be replicated in a dynamic cluster in which new process nodes could be added by implementing of a new performance monitoring process.

## 7.5 Conclusions and Future Work

In this paper we have seen that, along the evolution of Internet, different QoS manage- ment techniques have been used to face the demand of network resources. Since the beginning of Internet, a classification of traffic has always been considered, as IP Type of Service field shows and, later on, IntServ and DiffServ architectures have shown. In mobile communications it is also important to differentiate among the services that in- frastructures are holding. However, the classification of traffic has never been dynamic, and nowadays it is difficult to place the growing number of services in the current classes for current QoS management techniques.

Machine Learning brings the chance to create dynamic classification and character- ization of traffic in substitution of traditional techniques such as Deep Packet Inspection or port-based classification. In addition, Quality of Experience gives a new understanding of customers' needs and satisfaction by giving preference to what is important to user, not what user has

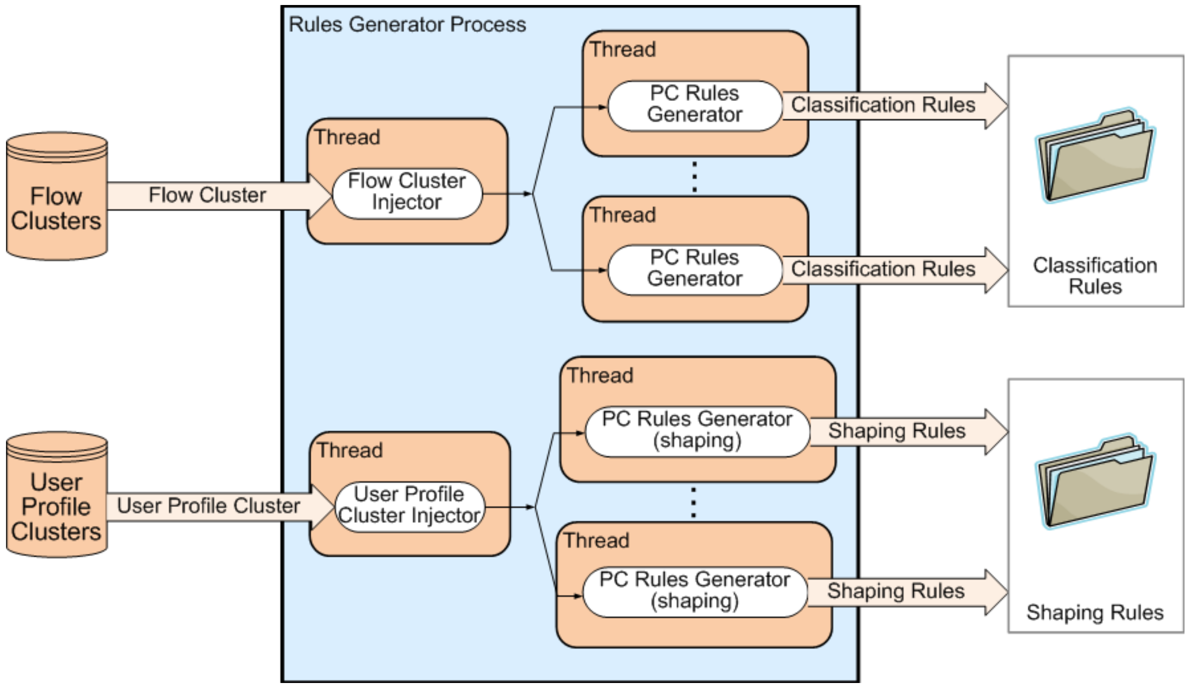


Figure 7.7: Example of scalability for the module ”Rules Generator”.

contracted with the ISP.

The proposed framework in this paper aims to handle Big Data ISP traffic and apply Machine Learning techniques in order to smartly manage Quality of Service. This framework will contain an Analytics framework that will study traffic in order to achieve the optimal classification both of the profile of the customers and the charac- teristics of the flows. With this knowledge, the Policy Controller module will create rules that may help Enforcement Point to make intelligent decisions about how traffic should enter to the network in different situations. Finally, this Enforcement Point will handle with Big Data traffic making use of parallelization to best process the huge amount of data (about 200000 packets per second) that will need to be treated.

In order to achieve the proposed goals for this framework, ONTIC project will focus his efforts in: (1) making a selection of which ML algorithms best fit the needs of classification and characterization of the traffic, as well as the input parameters for these algorithms (they must be related to flow features); (2) study how to best classify users per profile and per the usage they do of each cluster generated; and (3) create policies that will help to generate both shaping and classifying rules. A further step in case 3 would be to generate an automation process that would generate these rules without human supervision.

In addition, ONTIC project wills to contribute in the development of the areas covering Machine Learning and Big Data, so the results of the investigation will be incorporated to tools such as Traffic Identification Engine (TIE) [38] in order to join efforts with other communities.

## 7.6 Acknowledgment

The authors would like to thank ERICSSON Spain R&I department for their collaboration in the definition of the high level architecture and the work done related to Quality of Experience and Business Support System.

# Bibliography


[1] T. Nguyen and G. Armitage, "A Survey of Techniques for Internet Traffic Classification Using Machine Learning", IEEE Commun. Surveys & Tutorials, vol. 10, no. 4, 2008, pp. 56–76.

[2] A.C. Callado, C.A. Kamienski, G. Szabo, B.P. Gero, J. Kelner, S.F.L. Fernandes, and D.F.H. Sadok, "A Survey on Internet Traffic Identification", IEEE Comm. Surveys and Tutorials, vol. 11, no. 3, pp. 37–52, 2009.

[3] J. Postel. "Internet protocol". RFC 791. Sept. 1981. Available: http://www.rfc-editor.org/search/rfc_search.php

[4] D. C. S. S. R. Braden, "Integrated services in the internet architecture: An overview", RFC 1633. Sept. 1994. [Online]. Available: http://www.rfc-editor.org/search/rfc_search.php

[5] K. Nichols et al., "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", RFC 2474, Dec. 1998. [Online]. Available: http://www.rfc-editor.org/search/rfc_search.php

[6] OpenFLow Consortium. "OpenFlow switch specification". Can be accessed at http://openflowswitch.org/documents.php.

[7] ETSI TS 123 107 V11.0.0 (2012-11) Digital cellular telecommunications system (Phase 2+); Universal Mobile Telecommunications System (UMTS); LTE; Quality of Service (QoS) concept and architecture (3GPP TS 23.107 version 11.0.0 Release 11) ETSI Technical Specification, 2012.

[8] B.M. Leiner et al., "A brief history of the Internet", Internet Society, Reston, VA; http://www.isoc.org/internet/history/brief.shtml.

[9] Sullivan, Mark, Surveys: Internet Traffic Touched by Youtube, Light Reading, http://www.lightreading.com/document.asp?doc_id=115816 , January 2007.

[10] Kenjiro, C., Kenshue, F., Hiroshi, E., Akira, K., "The Impact and Implications of the Growth in Residential User-to-User Traffic", Proceedings of ACM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications SIGCOMM, pp. 207–218, 2006.

[11] A. Dainotti, A. Pescape, and K. C. Claffy, "Issues and future directions in traffic classification", IEEE Network, vol. 26, no. 1, pp. 35–40, 2012.

[12] Stewart, L.; Armitage, G.; Branch, P.; Zander, S., "An Architecture for Automated Network Control of QoS over Consumer Broadband Links", IEEE Region 10 TENCON 2005, pp. 1–6, 2005.

[13] Roughan, M., Sen, S., Spatscheck, O., and Duffield, N., "Class-of-service mapping for QoS: a statistical signature-based approach to IP traffic classification". In Proceedings of the 4th ACM SIGCOMM conference on Internet measurement (IMC 04), pp. 135–148, 2004.

[14] Jelinskis, J., Lauks, G., "Data Mining for Managing Intrinsic Quality of Service in MPLS". Electronics and Electrical Engineering, 2008, No.5, pp.33-36. ISSN 1392-1215.

[15] Cardoso, J., "Applying Data Mining Algorithms to Calculate the Quality of Service of Workflow Processes". In Intelligent Techniques and Tools for Novel System Architectures, pages 3-18, Springer Berlin / Heidelberg, Studies in Computational Intelligence, Vol. 109, ISBN: 978-3-540-77621-5, 2008.

[16] Bashar, A., Parr, G., McClean, S., Scotney, B., Nauck, D., "Machine learning based Call Admission Control approaches: A comparative study", International Conference on Network and Service Management (CNSM) – 2010, pp. 431–434, 2010.

[17] Menkovski, V., Exarchakos, G., Liotta, A., Cuadra Sánchez A., "Estimations and Remedies for Quality of Experience in Multimedia Streaming". In Proceedings of Third International Conference on Advances in Human-Oriented and Personalized Mechanisms, Technologies and Services (CENTRIC 10). IEEE Computer Society, Washington, DC, USA, pp. 11–15. DOI=10.1109/CENTRIC.2010.14

[18] Finamore, A., Mellia, M., Meo, M., "Mining unclassified traffic using automatic clustering techniques". In Proceedings of the Third international conference on Traffic monitoring and analysis (TMA 11), Jordi Domingo-Pascual, Yuval Shavitt, and Steve Uhlig (Eds.), Springer-Verlag, Berlin, Heidelberg, pp. 150–163, 2011.

[19] Erman, J., Arlitt, M., Mahanti, A., "Traffic classification using clustering algorithms", In Proceedings of Workshop on Mining Network Data (MineNet 06), pp. 281–286, 2006.

[20] Wang, Y., Xiang, Y., Zhang, J., Yu, S., "A novel semi-supervised approach for network traffic clustering", 5th International Conference on Network and System Security (NSS), pp. 169–175, 2011.

[21] Grimaudo, L., Mellia, M., Baralis, E., Keralapura, R., "Self-Learning Classifier for Internet traffic", IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), pp.423–428, 2013.

[22] Andrew, W, Moore and Zuev, D., "Internet traffic classification using bayesian analysis techniques". In Proceedings of the ACM International conference on Measurement and modeling of computer systems (SIGMETRICS 05), pp. 50–60, 2005.

[23] Zhang, J., Xiang, Y., Wang, Y., Zhou, W., Xiang, Y., Guan, Y., "Network Traffic Classification Using Correlation Information", IEEE Transactions on Parallel and Distributed Systems, vol.24, no.1, pp. 104–117, 2013.

[24] Grimaudo, L., Mellia, M., Baralis, E., "Hierarchical learning for fine grained internet traffic classification", Proceedings of 8th International Wireless Communications and Mobile Computing Conference (IWCMC), pp.463 -468, 2012.

[25] Wang, Y., Yu, S., "Supervised Learning Real-Time Traffic Classifiers", Journal of Networks, vol. 4, no. 7, 2009.

[26] Paredes-Oliva, I., Castell-Uroz, I., Barlet-Ros, P., Xenofontas D., Solé-Pareta, J., "Practical anomaly detection based on classifying frequent traffic patterns", Proceedings of IEEE INFOCOM Workshop 2012, pp. 49–54, 2012.

[27] Paredes-Oliva, I., Xenofontas, D., Molina, M., Barlet-Ros, P., Brauckhoff, D., "Automating root-cause analysis of network anomalies using frequent itemset mining', SIGCOMM ACM Conference, Editor(s) Kalyanaraman, Shivkumar and Padmanabhan, Venkata N. and Ramakrishnan, K. K. and Shorey, Rajeev and Voelker, Geoffrey M, pp. 467–468, 2010.

[28] Tstat Homepage, http://tstat.tlc.polito.it.

[29] Cisco Netflow, http://www.cisco.com/c/en/us/products/ios-nx-os-software/ios-netflow/index.html

[30] Apache Hadoop, http://hadoop.apache.org/

[31] Li, H., Wang, Y., Zhang, D., Zhang, M., Chang, E., "Pfp: parallel fp-growth for query recommendation". In RecSys'08, pp. 107–114, 2008.

[32] Moens, S., Aksehirli, E., Goethals D., "Frequent itemset mining for big data". In IEEE Workshop on Scalable Machine Learning SML: BigData, 2013.

[33] Debatty, T., Michiardi, P., Mees, W., Thonnard, O., "Determining the k in k-means with MapReduce". 17th International Conference on Database Theory EDBT–ICDT 2014, Athens, Greece, 2014.

[34] Bahmani, B., Moseley, B., Vattani, A., Kumar, R., Vassilvitskii, S. “Scalable k-means++”, Proceedings of VLDB Endow, pp. 622–633, 2012.

[35] Lloyd, S., “Least squares quantization in PCM”, IEEE Transactions on Information Theory, vol. 28, no. 2, pp. 129–136, 1982.

[36] Ferreira, R., Caetano, J., Machado, A., López, J., Kang, U., Faloutsos, C., “Clustering very large multi-dimensional datasets with MapReduce”, Proceedings of the 17th ACM SIGKDD International conference on Knowledge discovery and data mining (KDD 11), pp. 690–698, 2011.

[37] Panda, B., Herbach, J., Basu, S., Bayardo, R. “PLANET: massively parallel learning of tree ensembles with MapReduce”, Proceedings of VLDB Endow 2, pp. 1426–1437, 2012.

[38] Shaikh, J., Fiedler, M., Collange, C., “Quality of experience from user and network perspectives”, Annals of Telecommunications, vol. 65, no.1–2, pp. 47–57, 2010.

[39] Fiedler, M., Hossfeld, T., Tran-Gia, P., “A generic quantitative relationship between quality of experience and quality of service”, IEEE Network, vol. 24, no. 2, pp. 36–41, 2010.

[40] Hossfeld, T., Schatz, R., et al., “Internet Video Delivery in YouTube: From Traffic Measurements to Quality of Experience”, Data Traffic Monitoring and Analysis, E. Biersack, C. Callegari, and M. Matijasevic, Eds. Springer Berlin Heidelberg, pp. 264–301, 2013.

[41] Wang, T., Pervez, A., Zou, H., “VQM-Based QoS/QoE Mapping for Streamming Video”, Proceedings of 3rd IEEE International Conference on Broadband Network and Multimedia Technology (IC-BNMT), pp. 807–812, 2010.

[42] Pinson, M., Wolf, S., “A New Standardized Method for Objectively Measuring Video Quality”, IEEE Transactions on Broadcasting, vol. 50, pp. 312–322, 2004.

[43] Agboma F., Liotta, A., "QoE-aware QoS management", Proceedings of the 6th International Conference on Advances in Mobile Computing and Multimedia, pp. 111–116, 2008.

[44] Fechner, G., Boring, E., Adler, H., Howes, E., "Elements of psychophysics", New York:Holt, Rinehart and Winston, 1996.

[45] Menkovski, V., Oredope, A., Liotta, A., Cuadra Sánchez, A., "Predicting Quality of Experience in Multimedia Streaming", Proceedings of the 7th International Conference on Advances in Mobile Computing and Multimedia, pp. 52–59, 2009.

[46] Quinlan, J., "Induction of Decision Trees", Machine Learning, vol. 1, no. 1, pp. 81–106, 1986.

[47] Smola, A., Scholkopf, B., "A tutorial on support vector regression", Statistics and Computing, vol. 14, no. 3, pp. 199–222, 2004.

[48] Mushtaq, M, Augustin, B., Mellouk, A., "Empirical study based on machine learning approach to assess the QoS/QoE correlation", Proceddings of 17th European Conference on Networks and Optical Communications (NOC), (2012)