QoM: A New Quality of Experience Framework for Multimedia Services

Khalil ur Rehman Laghari, Thanh Tung Pham, Hang Nguyen, Noel Crespi
Institut Telecom, Telecom SudParis, 9 Rue Charles Fourier, 91011Evry France
khalil.laghari, pham.tung, Hang.Nguyen, noel.crespi@{it-sudparis.eu}

Abstract— **Quality of Experience (QoE) provides human centric assessment of multimedia quality. QoE of a multimedia service is affected by various application and network layer QoS parameters; content and business parameters. In this paper, we propose a new QoE framework for Multimedia services (named as QoM) for run time quality evaluation of video streaming services based on the influence of QoE factors, various network and application level QoS parameters. The newly proposed QoM framework monitors QoS and QoE data, evaluates it and moreover, in the event of decline in QoE, it can also send alert messages to the Administrator based on some policy rules. Our new QoM framework is being launched as an open-source QoE evaluation tool for the industry and research community. More realistic experimental work is also underway and we intend to conduct user tests to evaluate the performance of the proposed QoM framework in a context of real 4G WiMax wireless networks.**

Keywords- QoE; Video streaming; Multimedia services; QoS; QoE framework; QoM, monitoring tool

I. INTRODUCTION

The demand for the multimedia applications is rapidly increasing, and today video streaming is being widely used for video conferencing, video on demand, telemedicine and e-learning. But video streaming service has stringent quality requirements both from technological point of view (QoS) and user's point of view (QoE). There are many network-dependent, application-specific, content-based, business and context oriented factors which influence multimedia QoE.

Traditionally, technology centric approaches based on QoS parameters have been employed to analyze multimedia service quality. Existing QoS based solutions for internet are IP DiffServ, IP Intserv, they may guarantee resource allocation however, service guarantees alone are not sufficient to promise superior quality of experience [1]. Furthermore, QoS based quality assessments have often found to be grossly inaccurate at predicting user experience, and as such are not applicable in evaluating multimedia quality [2]. Therefore there is need to change the direction from technology centric QoS to human centric QoE approach. QoE provides an assessment of human expectations, feelings, perceptions, cognition and acceptance with respect to a particular product, service or application [3]. The International Telecommunication Union ITU-T defines QoE [4] as "The overall acceptability of an application or service, as perceived subjectively by the end-use".

In this era of competition, poor customer experience leads to a chain reaction of negative word of mouth, pushing customers into the arms of waiting competitors. Today, humans are quality meters and their expectations, perceptions and needs carry greater value.

The important challenge is to identify, collect, and measure such diverse set of QoE data, which is under the influence of different influencing factors. Thus the power is not found in the simple collection of customer profile information, network parameters and some SLA requirements, but rather in the means and methods by which the detailed quality of experience information is gathered, analyzed and applied to understand the performance at run time. Therefore, knowing what to collect, how to collect it, how to analyze it and subsequently understanding how to use QoE information for quality assurance and QoE aware service delivery is crucial.

But there is a big challenge to face i.e., human perceptions and feelings are very subjective and random in nature, how to capture and quantify human subjectivity. The one common solution is to conduct user studies and surveys to know customer opinions and feelings and then using some quantitive and qualitative techniques, subjective data is converted into usable information. In our previous work, we presented holistic model [5] to understand human requirements for multimedia services and then we conducted user study to evaluate the impact of QoS and content parameters over user QoE for video streaming service [6]. The data were analyzed based on both quantitive and qualitative methods. Learning from our previous works, we propose QoE framework for multimedia services in current paper. This framework is an attempt to provide a QoE evaluation tool to industry and research community. It is still at its infant stage but after launching it as an open-source tool to research community, we expect it to grow and get matured quickly.

This proposed QoM framework is intended to perform the functions of monitoring data, analyzing it and then reporting it to the administrator (admin). More realistic experimental work is also underway and we intend to conduct user tests to evaluate the performance of the proposed QoM framework in a context of real 4G WiMax wireless networks.

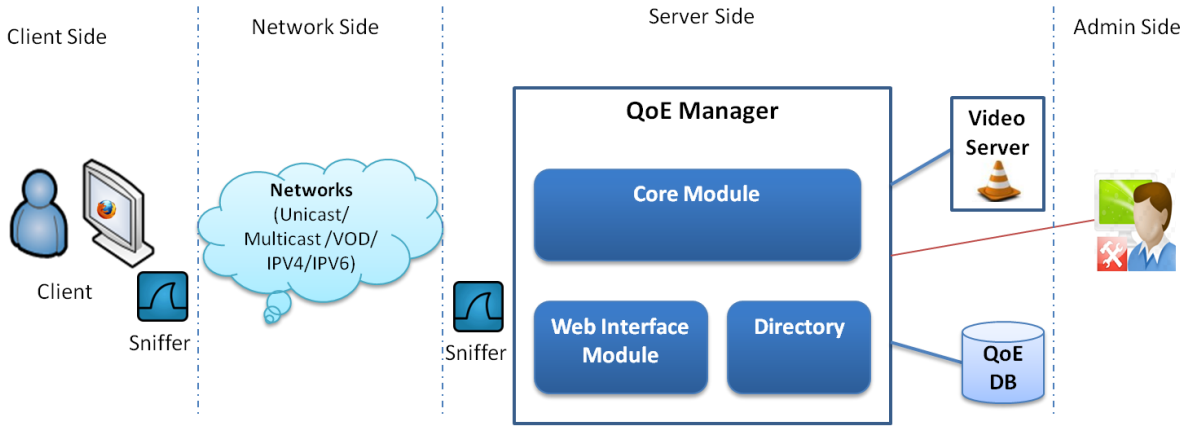

Figure 1. Architecture of the proposed QoE framewOrk for Multimedia services (named as QOM framwork)

This paper is organized as follows. In section II, we present brief background on video quality measurement techniques and frameworks. In section III, we present the architecture of our proposed QoM framework and in Section IV, we present the sequence diagram describing the working steps of the proposed QoM framework, finally, we conclude our work.

II. VIDEO QUALITY MEASUREMENT BACKGROUND

Accurate monitoring, evaluation and reporting of a video QoE are important requirements of multimedia service providers. The service provider needs to monitor the impact of various network layers (such as delay, jitter, and packet loss), application layer parameters (such as resolution, frame rate, encoding rate), and content types (such as slowing moving, fast moving video) over user QoE.

There are broadly two techniques to capture video quality, one is called subjective techniques and other is objective techniques. In subjective techniques, user studies, customer surveys and interviews are conducted to translate user feedback into statistical metrics; on the other hand, objective techniques infer QoE from collected network traffic or QoS data based on some estimation or prediction methods. ITU-T proposed a subjective method for video quality assessment in their recommendation P.910 [14]. For details about objective schemes, the work in [15] could also be referred.

Subjective methods are time consuming and expensive but they provide more accurate results, while objective methods are less time consuming but their accuracy is always dependent on the prediction method and moreover they are focused on QoS data only. Thus, they may also miss out the influence of other important information (such as context, business, and human psychology) over human behavior which could be gathered through surveys and user studies.

Some solutions for QoE measurement for multimedia services are being delivered by niche vendors but they are mostly focused on Objective QoS/QoE factors such as perceptual evaluation of speech quality (PESQ) technique peak signal to noise ratio (PSNR) [7,8].

In academia, there is also on-going effort to propose QoE monitoring and evaluation frameworks. One such work is [9]; authors proposed a model and framework to evaluate video quality based on application level QoS parameters. Another framework is subjective assessment framework named MintMos [2] which provides quantitative analysis of both network QoS (NQoS) and application level QoS (AQoS) parameters to predict QoE scores (MOS). Our framework captures both AQoS and NQoS parameters. It is based on subjective assessment and for data analysis; both quantitive and qualitative methods could be used. For quantitative analysis, descriptive statistics, multivariate regression techniques and rough set theory could be used and for qualitative data analysis, QOM supports CCA (categorize, catalog and Analyze) based data classification. For detail on CCA, and rough set theory based data analysis, readers are suggested to refer to work [6]. Our proposed framework is not only monitoring and analysis tool but it can also report alerts in the event of QoE degradation. The comparison of different frameworks is presented in following Table 1.

Table 1. Comparison of QOM with other Video Quality tools

QoE Frameworks	***MintMos [2]***	***Taichi Kawano et al [9]***	***Niche Vendors [7,8]***	***QOM Frame work***
Parameters	NQoS and AQoS	AQoS: Video Blur and Blocking	PESQ, PSNR, VQM	NQoS and AQoS
Monitoring Support	Yes	Yes	Yes	Yes
Analysis Support	Quantitative	Quantitative	Quantitative	Quantitative and Qualitative
Reporting	No	No	Yes	Yes
Remarks	Subjective Evaluation	Subjective Evaluation	Objective Evaluation	Subjective Evaluation

III. ARCHITECTURE OF THE PROPOSED QOM FRAMEWORK

In this paper, we created a new QoE framewOrk for Multimedia services (named here as QoM). It is built on J2EE model [13]. The important components of this newly proposed QoM framework are QoE Manager, QoE database, Video streaming server, Web-based client interface, Web-based admin interface. Figure 1 shows the detail architecture of the newly proposed QoE framewOrk for Multimedia services (QoM). We deploy sniffer tool at two sides of the service model, one at server side and one at client side. With this deployment, we can have all the information of service providing process. QoE manager performs the analysis and management functions on captured data. VLC streaming server [10] provides the VoD service over the client web browser. All the objects of the proposed QoM framework are synchronized with the MySQL (QoE) database. We discuss one by one each of these components of the newly proposed QoM framework in next subsections.

A. Proposed QoM framework's Client Interface

Figure 2 describes the proposed QoM framework's Client Interface. A web-based client interface is developed to facilitate users to watch videos online and give their QoE ratings. Client interface consists of user profile information, QoE rating tab and Video. User profile information consists of user demographic information (age, gender, profession, country, and location). QoE rating consists of quantitive scales of 5-point scale, bi-nominal ratings and qualitative comments. Different QoE scales provide users a facility to give their response about quality based on these scales. Video part of client interface permits users to watch any of given video content (e.g., News, Football, and Container) by using any of two resolutions (360x240 or 640x480).

We have used the latest Firefox browser for current version of framework but our client interface also supports Google chrome. Client side machine should have latest version of Firefox or Google Chrome with VLC plug-in utility. As client interface is based on web browsers, it can run on any operating system which supports Firefox and Google chrome. Secondly client machine should contain a Wireshark utility in their machine so that client side data capturing could be done. Each time, a user wants to watch video, s/he will be allotted a unique random session number. S/he has to insert client IP address before processing video test. A client comes to the VoD service web page and chooses the video contents and its resolution. After watching demanded video, client will provide his personal information such as name, age, profession, gender, country, place, like or not the video content, comments, resolution selected. However client interface is not dumb terminal, it has ability to calculate client side QoS information such as Jitter, one-way delay, number of packet received, and resolution. Upon clicking submit button on client interface, the information will be transferred to QoE manager for further processing.

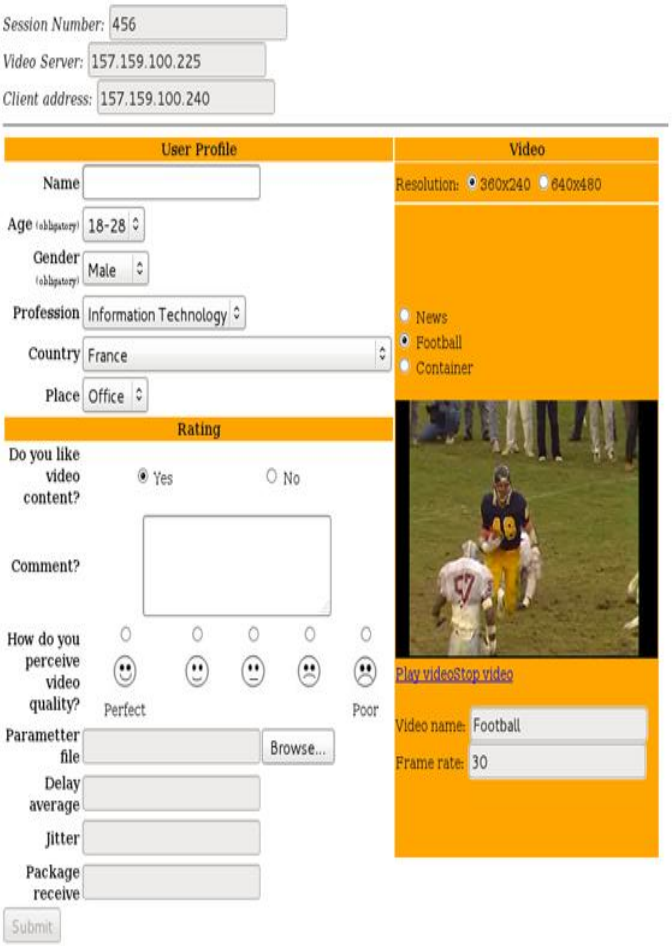

Figure 2. Proposed QoM framework's Client Interface

B. Proposed QoM framework's Sniffers

In our QoM framework, TShark is used as sniffer. TShark is a terminal oriented version of Wireshark designed for capturing and displaying packets when an interactive user interface isn't necessary or available [11]. Using TShark, packet data can be captured from a live network, or read packets from a previously saved capture file, either printing a decoded form of those packets to the standard output or writing the packets to a file [12]. TShark's native capture file format is "libpcap" format, which is also the format used by "tcpdump" and various other tools.

We use sniffers both at client side and server side. Sniffer at client side is used to sniff the packet data (UDP) transmitted by client and received from server. The captured data will be written into a file in the client's terminal, so that after VoD session, client's terminal can provide the information such as the number of packet received, delay and jitter to QoE Manager.

Sniffer at server side will be triggered at the time, when user chooses to start the video service test. Client needs to run wire shark manually and s/he will be asked to turn off sniffer at the end of session, while sniffer at server side will start automatically and it will turn off after an elapsed time t (i.e., 3

min). Two Wireshark processes should be configured with the same filtering parameters such as client's IP address, server's IP address, and UDP protocol.

C. Proposed QoM framework's QoE Manager

Proposed QoE Manager is the heart of the QoM framework and it consists of three important modules: Core module, Web-interface module, Log directory. Both Core module and web-interface module run on Glassfish Application server. Web interface module consists of JSP, and Java Servlet. JSP is used for displaying information. Java servlet is responsible for interconnection between Core and client/admin interface. Log directory stores logs of of every video streaming session.

Core Module is based on Java beans and it is responsible for taking all necessary management actions. It comprises of three main component functions i.e., object definition, operations and DB processing. Core module defines four categories of objects such as user, session, QoS and video. Operation part of Core module is responsible for conducting all analysis, processing and management functions of this framework. DB (database) processing signifies the operation processing between Core module, QoE- DB and log directory.

Beside the core module, we created the web interface module. This module is based on JSP and Java servlet to facilitate communication between core module and web-based client/admin interface module. Web interface module receives the request from client and/or admin interface and transfers them to Core module for further action. Core module processes data and send output data to web interface module to enable client/admin terminals to display/access data.

Directory is responsible for storing data log file captured from sniffer at server side. All the packets coming in each side will be reported into log files.

Inside the J2EE platform, we had to build some function that connect to the outside sniffing module therefore, aost all the step capturing the packet, writing into sessions' log files, analyzing the log files, calculating the QoS parameters and displaying the information to administrator and client are automatically processed.

Inside the framework, each session has its own session id, and for each session, the sniffer at server side captures information which is stored in Log directory. And, QoE Manager can obtain the parameters like Delay, Jitter, Packet received from the client side. The QoE Manager then uses batch processing to update all the session parameters and calculate packet loss, predicted QoE.

D. Proposed QoM framework's QoE Database

MySQL server is used to create database tables for the framework. It is connected to QoE manager through JDBC (Java DataBase Connector). Figure 3 describes the proposed QoM framework's Structure of Data Tables. There are four tables, the Session table, QoS, Video and User table. The Session table stores the session's id and foreign key of other tables QoS, Video and Users.

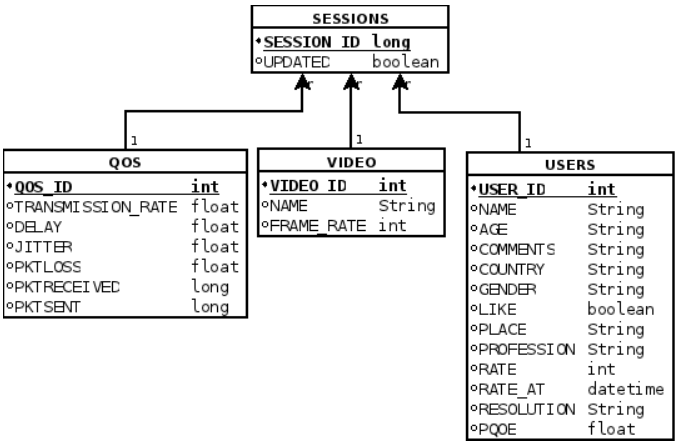

Figure 3. Proposed QoM's Structure of Data Tables

The QoS table stores the information of all the network parameters. It includes Transmission Rate, Delay, Jitter, Packet loss, Packet received, Packet sent. The Video table stores application level QoS parameters such as video name, frame rate, resolution. All the videos are stored in a specific folder of the server. We store video's information such as video name and video frame rate in the database so that this information will be shown to the client when they watch them. The User table stores all the personal information and the rating information of sessions. After watching the video clip, client will be asked to provide their information, rating and comments. User information will be stored in the database for further analysis.

E. Proposed QoM framework's Admin Tool

Proposed QoM framework's Admin Tool is a web-based administrator (admin) interface. It performs three main functions: view data records, update data records by initiating batch processing and receive QoE decline alerts. The administrator (Admin) at any time can view the available data records (QoS and QoE status) of a VoD service. The administrator (Admin) can get updated information of various sessions by initiating batch processing (Admin Clicks Batch-processing button on his interface).

QoE manager performs batch processing to analyze data and provide updated records. QoE manager also periodically checks data records. If it finds any anomaly (e.g., QoE ratings are less than 3), it will report to Admin for further investigations and actions. Based on the idea of the framework that the level of satisfaction of client is very important to the service (business), the server will update all the rating information from client's session.

If there is any session in which the client gives the bad rating, the server will display an alert message on the administrator's screen as shown in Figure 4.

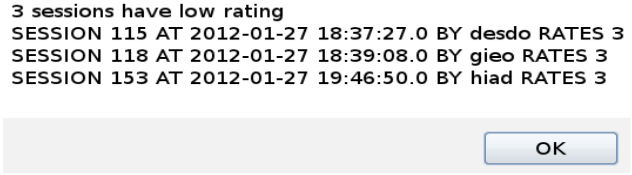

Figure.4 Proposed QoM's Screen shot for Alert Report

For our proposed QoM framework, we have made a simple policy rule, however many different policies and rules can be developed based on the requirement of service and admin.

Example: Policy for Reporting Alert

Reload Admin's screen with updated QoE ratings every 5 seconds

if (session is new (not updated) && rating of session is equal or less than 3) then

put session's information of rating, user's information into the alert message;

session is set updated;

end if

IV. Proposed QoM framework's sequence diagram

In Figure 5, we produce sequence diagram to explain how our newly proposed QoM framework operates to produce QoE evaluation for multimedia services. We present three actions.

- At the start of the session, user manually starts Wireshark at his side and then opens web based client interface to watch video clips. Sniffer at server side will be triggered at the time, when user chooses to run the VoD service. User request for video content will be routed to VLC server through QoE Manager. VLC start streaming specific video content to a client. At the end of video clip, user gives his rating and uploads client side data log file. Web-based client interface calculates number of packet received, delay and Jitter and sends them to QoE Manager. Upon the reception of client side data, QoE Manager analyzes log data to compute packet loss, basic statistics on data (Mean etc) and predicted QoE (based on Regression). Once data analysis process completes, the attribute file is stored in QoE database. Furthermore updated records will be sent to Admin.
- Admin can also view the records of available data and for this purpose, QoE Manager is contacted, which gets attribute records from database and sends them to Admin.
- QoE Manager also periodically verifies the records, for this purpose, after every 5 seconds, he gets updated records. Furthermore QoE Manager takes decision based on some policy rules, for instance, if QoE rating is found to be equal to or less than 3, it generates alerts for Admin to investigate the decline of QoE.

V. Limitations and Future work

Firefox is very powerful browser; it supports many kinds of plug-ins and service from Java, PHP, and JSP to video, voice. In addition, it is very secure that it does not allow the script such as Java script to run and execute the client's application or service. That's the reason the client who want to use the service, he must activate the capturing service manually first, and after that the service will work well. This dependency on manual switching on sniffers is important challenge to address in next version of the framework.

Our newly proposed QoM framework permits users to watch video and give their feedback in textual format (comments). The qualitative analysis will be included in the next step of our work.

VI. Conclusion

In this paper, we have presented a new framework for the monitoring, evaluation and management of video streaming service based on quality of experience. The proposed QoE framewOrk for Multimedia services (named as QoM framework) captures network and application layer QoS data, qualitative QoE data and quantitive user ratings and content information. Using descriptive statistics and multiple linear regressions, QoE is evaluated. In case of any decline in QoE, an alert message is conveyed to the administrator (Admin) for further investigation. Our newly proposed QoM framework is being launched as an open-source QoE evaluation tool for the industry and research community. For next step, we intend to improve this framework by solving the limitations mentioned in the previous section. More realistic experimental work is also underway and we intend to conduct extensive user tests to evaluate the performance of the proposed QoM framework in a context of real 4G WiMax wireless networks.

References

[1] Cisco , "Whitepaper: DiffServ-The Scalable End-to-End Quality of Service Model ," Cisco , 2005.

[2] Venkataraman M, Chatterjee M, and Chattopadhyay S, "Lighweight, real-time, no-reference framework for inferring subjective-QoE," in IEEE Globecom, 2009.

[3] Khalil Ur Rehman Laghari, N. Crespi, B. Molina, and C.E Palau, "QoE Aware Service Delivery in Distributed Environment," in Advanced Information Networking and Applications (WAINA), 2011 IEEE Workshops of AINA Conference , Singapore, 22-25 March 2011, pp. pp.837-842.

[4] ITU-T, "Definition of Quality of Experience (QoE), ," International Telecommunication Union, Liaison Statement Ref.: TD 109rev2 (PLEN/12, 2007.

[5] Khalil Ur Rehman Laghari, N. Crespi, and K Connelly, " Towards Total Quality of Experience : A QoE for multimedia services in comunication ecosystem," *IEEE Communication Magazine*, April 2012.

[6] Khalil Ur Rehman Laghari, Imran Khan, and N. Crespi, "Quantitative and Qualitative Assessment of QoE for Multimedia Services in Wireless Enviornment"," in *MoVid Workshop, ACM Multimediasys Conference*, Chapel Hill NC USA, 22-24 Feburary 2012.

[7] www.witbe.net,

[8] www.qoesystems.com

[9] Kawano Taichi, Yamagishi Kazuhisa, Watanabe Keishiro, and Okamoto Jun, "NO REFERENCE VIDEO-QUALITY-ASSESSMENT MODEL," in *Proceedings of 2010 IEEE 18th International Packet Video Workshop*, Hong Kong, 2010.Video Lan VLC player. [Online]. http://www.videolan.org/vlc/

[10] Online Tutorial D.2. tshark: Terminal-based Wireshark.

[11] Tshark description. http://www.wireshark.org/docs/man-pages/tshark.html.

[12] J2EE Model tutorial. [Online]. http://docs.oracle.com/javaee/1.4/tutorial/doc/

[13] ITU-T, "Subjective video quality assessment methods for multimedia applications," International Telecommunication Union, Geneva, Switzerland, Recommendation P.910, 2008.

[14] Hamid R. Sheikh and Alan C. Bovik Zhou Wang, "Objective Video Quality Assessment," in *The Handbook of Video Databases: Design and Applications*.: CRC Press, September 2003, ch. 41, pp. 1041-1078.

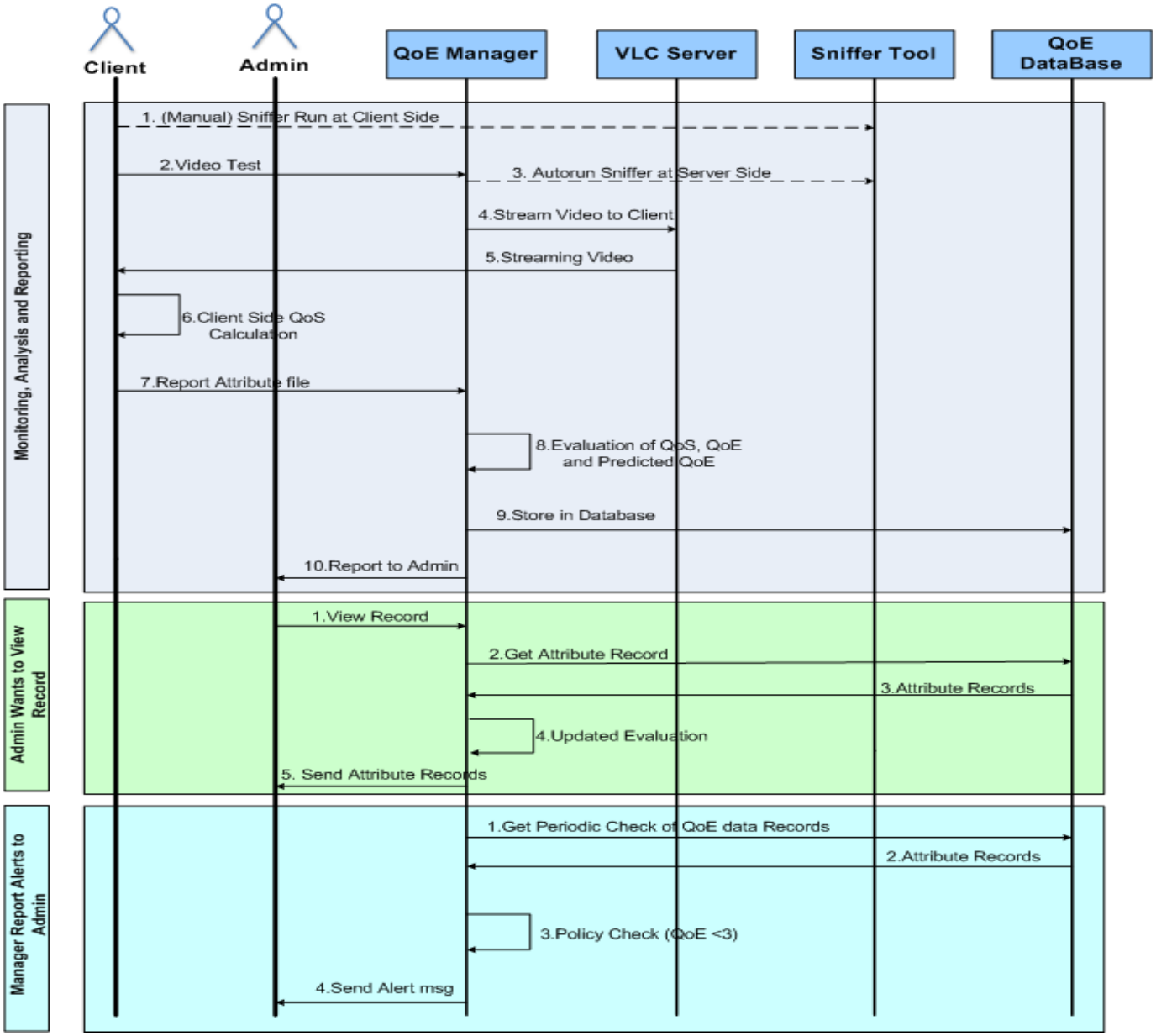

Figure 5. Proposed QoM Framework's Sequence Diagram