A Software Framework for User Level QoS in Unreliable Internet Connection

[1]Ratna Wardani, [2]Dr. Lukito Edi Nugroho, [3]Dr. Ahmad Ashari, [4]Dr. Risanuri Hidayat
[1]Dept. of Electronic Engineering Education, Faculty of Engineering
Yogyakarta State University, Yogyakarta Indonesia
[2]Department of Electrical Engineering and Information Technology
Faculty of Engineering, Gadjah Mada University Yogyakarta Indonesia
[3]Department of Physics, Faculty of Mathematics and Natural Sciences, Gadjah Mada University, Yogyakarta Indonesia
[4]Department of Electrical Engineering and Information Technology
Faculty of Engineering Gadjah Mada University Yogyakarta Indonesia

Abstract
This paper describes a QoS framework that provides support for user Quality of Service (QoS) specification and service enforcement in low-quality connection environment using user-oriented approach. The QoS framework has been designed with the aim of solving the limitation Internet access in unreliable Internet connection. This framework provide user with the flexibility in the specification subjective QoS requirements and give an alternative access arrangement if resource availability in the system is limited. User specifies his/her subjective preference to the framework and it would check the resource availability, then compare to the user preferences. In the case resource availability is lower than user preferences, the framework switch access mechanism to another option as determined by user requirements. This paper also describes the design of QoS framework in detail and presents a prototype implementation to analyze the applicable of its design.

Keywords
User QoS, user-oriented approach. QoS framework, unreliable Internet connection.

I. Introduction

Given the low-quality connection, we realized that the model for accessing Internet that exists today is not compatible with the poor communication infrastructure. Until recently the Internet application such as World Wide Web (WWW) and the associated browser have provided no support for accessing Internet in low-quality connection environment. They are designed for high-bandwidth, high-connectivity environments [5]. That is, they optimize for speed, assuming that the users can quickly look through the result and immediately run a second, modified their request if they are unhappy with the results of their access. This tight feedback loop between the users and the browser is inappropriate for low-quality connection environment. We therefore need a new model that can provide support for accessing Internet in low-quality connection environment.

This research is concerned with the study of mechanism for the provision of quality of service guarantees for Internet access in low-quality connection. The research aim to propose the QoS framework for the specification of user's access and allow the users to specify their subjective preferences through the Quality of Service parameters. The framework supports a dynamic access model that provides users with more flexibility in controlling access behavior. This model provides the alternative option for user access if resource availability in the system is limited. The user is given opportunity to define their access and determine the parameter for each application which they are chosen. The system will check the resource availability and then compare to the user preferences. In the case resource availability is lower than user preferences, the system can move to access alternative as determined by user requirements.

II. Quality of Service

This section presents a discussion about the main concepts in the area of Quality of Service, including subjective QoS and QoS specification.

A. Term and definition

Quality of Service is very popular and overloaded term that is very often looked at from different perspectives by the networking and the application-development communities. Quality of Service was primarily used by the communications and networking areas to describe the ability to measure and guarantee transmission rates over networks [1]. In more broadly vision, Quality of Service can be defined as a relation between server and client. The server provides services with a specific quality level whereas the client requests a service with a desired quality.

Growing usage and diversity of applications on the Internet makes Quality of Service increasingly critical. To date, the majority of research on Quality of Service is systems oriented, focusing on traffic analysis, scheduling, and routing.

The requirements for Quality of Service (QoS) of Internet access are traditionally expressed in terms of network oriented or systems oriented parameters. The term QoS refers to a set of performance metrics that provide an objective measurement of a user in a given network. Most of the researches in the provision of QoS have occurred in the context of network-oriented QoS. Those researches have focused on providing suitable traffic models and service. Many concepts have evolved to define and provide an improved QoS.

The general definition of QoS provided by the International Telecommunication Union (ITU) [4] is that QoS is: “the collective effect of service performance, which determines the degree of satisfaction of a user of the service”. Different user and communities can interpret QoS differently. This research is using QoS definition in the user's perspective.

B. User level QoS

There are two main aspects of QoS: subjective QoS (user level QoS) and Objective QoS (application and system level QoS) [7]. Subjective QoS is the user's overall perception of service quality. It is the user's opinion whether a service is working satisfactorily or not. Subjective QoS is difficult to be specified with objective measures; therefore user-perceived quality is often expressed non-technically [2]. Objective QoS refers to the technical aspects

 ISSN : 2229-4333(Print) | ISSN : 0976-8491(Online)

of QoS, so it can be specified with quantitative measures. The different scopes of QoS are described in Fig. 1.

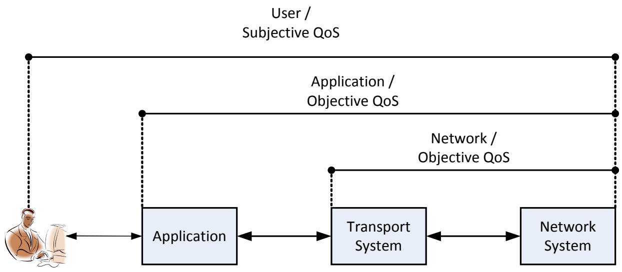

Fig. 1: Scope of QoS [8]

Subjective QoS represents two aspects: user's perception and user-level QoS requirements [8]. The user has a high-level perspective over Quality of Service of the application. It is difficult for users to express their subjective QoS in network QoS parameters, such as bandwidth, delay, jitter and packet loss. Therefore, in the context of user's perception we need terms that describe user-perceivable QoS, rather than an in-depth conception of the underlying implementation and operation of the network service [8].

C. QoS specification

While there is work in the general area of QoS, there is limited research in the specific area of user-oriented QoS for accessing Internet over low-quality connection. It is remarkable that research in QoS have been focusing mainly on network QoS(e.g. IntServ, DiffServ and RSVP) and multimedia application [1]. Many concepts have evolved to define and provide an improved QoS. The QoS concept is referred to Policy-based Networking. It is lets the network managers define service policies that govern how much bandwidth goes to specific applications and end users [6]. Another concept refers to QoS as network ability to provide service guarantees appropriate for various applications while at the same time making efficient use of network resources [9]. More specifically, QoS refers to a set of metrics performance that provides an objective measurement.

Specification of Quality of Service is vital to realizing quality guarantees. The specification can be done at various levels of system (i. e. network, application and user). Network level Quality of Service specification states the degree of resource commitment required to maintain performance guarantees. In this layer, the specification of the Quality of Service is made in quantitative aspects (i.e. delay, jitter, throughput, and bandwidth). Application level Quality of Service specification describes the application-specific Quality of Service requirements. Since different applications have different Quality of Service requirement, each application should specify its requirements to a network in order to achieve the desired Quality of Service. If there are no requirements given, the network will take for granted that any level of service is acceptable, and therefore can provide any level of networks support. User level Quality of Service specification reflects the user-perceptive quality of the application quality in the subjective criteria.

Most researchers in the field of user level Quality of Service agree that user Quality of Service specification must not include technical details in describing Quality of Service as perceived by the user. They also agree that there is a lot of subjectivity and context relevance associated with the user's perception of the Quality of Service. For instance, user Quality of Service can be described in term of user perceived characteristics of service performance. It is expressed as a number of parameters.

III. A QoS framework

This section will describe a framework for implementing the user level QoS mechanism as the platform for Internet access in unreliable Internet connection.

A. Parameter of user level QoS

The user has a high-level perspective of QoS application. It is difficult for users to express their subjective QoS in network QoS parameters, such as bandwidth, delay, jitter and packet loss.

In this paper, we proposed three parameters to set QoS specification that expected by users. These parameters will be an attribute that characterized the modeling of Internet access based on user-oriented QoS, as shown in Table 1.

Table 1: Parameter of user level QoS

Parameter	Attribute	Description
t	Time_access	Time required to receive a response from the service requested
s	Successful_ access	Availability of the service requested
c	Content_ relevance	Degree of matching between output/ response and the service requested

B. A model for user level QoS

Fig. 2 shows the functional block diagram of the software framework.

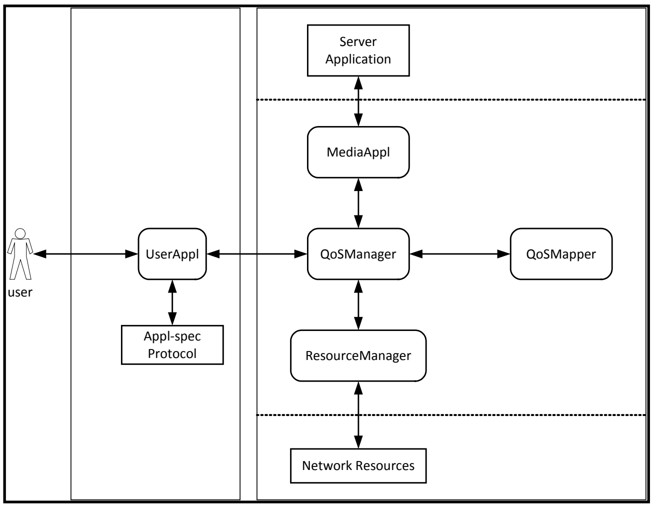

Fig. 2: Block diagram of QoS framework

Users specify the QoS requirements and preferences using the application interface. The user's requirement may be specified for one or more subjective QoS parameters. QoSManager coordinates and performs the mechanism on behalf of the interacting components. In order to decide on the solution (i.e. selection of appropriate service based on user's preferences), the QoSManager has to make a reference to: (i) user's Quality of Service requirements and preferences, (ii) available resource conditions, and (iii) the operational point of application media. For this purpose, ResourceManager informs the QoSManager regarding the state of the resources. Mapper would convert high-level user QoS specifications to a set of resource requirements. QoS parameters have to be translated between different levels of abstraction to be meaningful for the mechanism present at a particular level. Finally, ApplicationMedia performs the available media and the parameters related to the application that requested by users. The framework is based on a reactive mechanism. This mechanism can be represented by the following scheme:

Si : { pre:(Spre Λ e[guard]) [V (Spre Λ e[guard])]* | action:(ai , qexp) | post: Spost [V Spost]* }

where pre is a predicate which denote a precondition that defines one or more previous states Spre and a transition e containing a guard. The transition e[guard] represents an event that triggers a state in which a process action: (ai,qexp) is executed. The action is an ongoing activity that is performed as long as the model element is in the state or until the computation specified by the action expression is completed. Finally post is a predicate which denote a postcondition that defines one or more states that can possibly be reached from Si. How a possible state is selected is defined by the result of parameter evaluation (i.e., specified as a guard of a next possible state). The state (Si) is triggered by the transition e[guard]. The guards in the transition e indicate the conditions that determine which state will be executed, meaning if the condition is true then the specified state will be executed and contrary if the condition is false then the process will go to another state. In the state Si, pre-condition Spre is a requirement for the action (ai,[qexp]) and post-condition post: Spost [□ Spost]* showed the next state that will be occur after the action is completed. The parameter in the pre-condition predicate contains the evaluation value of the condition (guard) in the transition e. This value determines what action will be processed in the state Si.

C. Architectural components

The QoS framework is designed based on reactive model for specification of subjective QoS. It has the following interaction among its components: (i) The user should be able to specify his/her requirements; (ii) The QoSMapper would convert high-level user QoS specification to a set of resource requirements; (iii) The ResourceManager would indicate resource availability and inform to QoSManager; (iv) Based on these the QoSManager would make a decision which would be then activated MediaAppl to perform the available application.

QoS framework architecture is shown in Fig. 3.

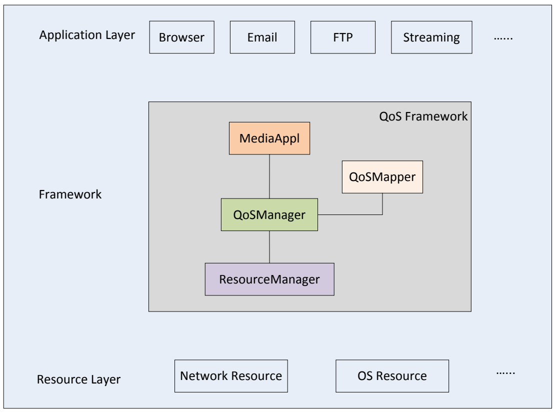

Fig. 3: Components architecture of QoS framework

The QoS framework works as an intermediate layer connecting the existing mechanism at the application layer with the mechanism in the resource layer or as an integrated modul in the aplication. Through this architecture, Internet applications can take advantage of the QoS framework mechanism to change the behavior of applications so that applications can provide access to quality services for users in the low-quality connection environments.

IV. Implementation

We have designed and implemented a software framework with the intent of addressing the user level QoS requirement in unreliable Internet connection. We use application scenario in order to explain how the model will be implemented and how to evaluate the mechanism.

Scenario:

1. A user wants to download a file "eMule-installer.exe" from the location http://sourceforge.net with a response time parameter (t_user). The user specifies that he does not want to wait for longer than 10 seconds.
2. If the requirement cannot be satisfied due to some network problems, the user specifies an alternative:

a. The download process is to be retried.

b. If the requirement still cannot be met, then the download process is put on background and the downloaded file is emailed to a specific address.

A complete specification for the above scenario is described as follows

S1	:	{(Initial(), e1[nil]) | (Get("eMule-installer.exe", http://sourceforge.net, t_user ≤ 10)) | (S2 V S3)}
S2	:	{(S1 Λ e2[t_process ≤ t_user]) V (S3 Λ e4[t_process ≤ t_user]) | (NormalDL("eMule-installer.exe", http://sourceforge.net, t_user ≤ 10)) | (S4)}
S3	:	{(S1 Λ e3[t_process ≤ t_user]) | ((Retry("eMule-installer.exe", http://sourceforge.net, t_user ≤ 10)) | (S2 V S5)}
S4	:	{(S2 Λ e5[true]) | (SaveFile("eMule-installer.exe", MyDirectory, true)) | (End)}
S5	:	{(S3 Λ e6[t_process > t_user]) | (BackgroundDL("eMule-installer.exe", http://sourceforge.net, t_user > 10)) | (S6)}
S6	:	{(S5 Λ e7[true] / act := S5) | (SendEmail("eMule-installer.exe", ratna@uny.ac.id, true)) | (End)}

A graphical illustration of the specification is given by Fig. 4.

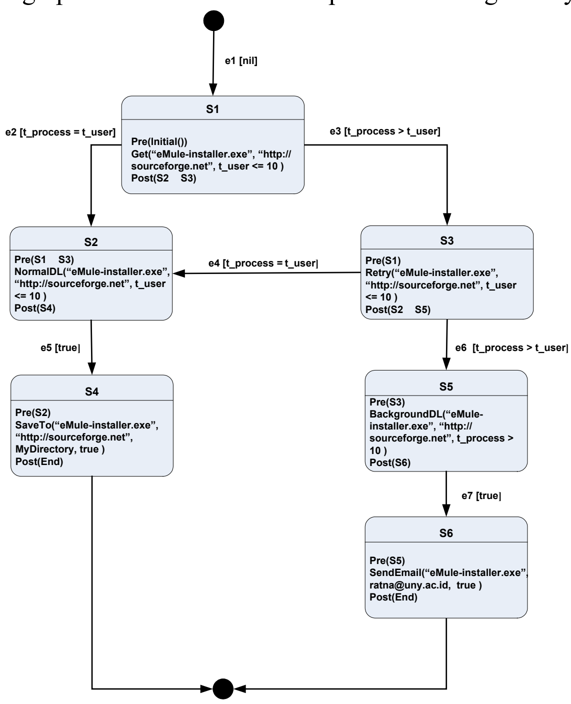

Fig. 4: An illustration of state-transition model for application scenario.

Important conclusion can be reached based on the testing of the application scenario and on the result of testing QoS parameters executed with it.

Testing parameter(s) :

time_response (t)

ISSN : 2229-4333(Print) | ISSN : 0976-8491(Online)

Application scenario :

1. A user wants to download a file "eMule-installer.exe" from the location http://sourceforge.net with a response time parameter (t_user). The user specifies that he does not want to wait for longer than 10 seconds.
2. If the requirement cannot be satisfied due to some network problems, the user specifies an alternative:

a. The download process is to be retried.
b. If the requirement still cannot be met, then the download process is put on background and the downloaded file is emailed to a specific address.

The processing application scenario is given by Table 2.

Table 2: An illustration of processing application scenario

State	Description
Pre-Condition	Based on user specification, the pre-condition would be in one of these condition: 1. User's QoS specification : Initial(t_user ≤ 10 seconds) 2. ((resourcecAvail == true) && (serverAppl == true) && (t_user ≤ t_process)) 3. ((resourcecAvail == true) && (serverAppl == true) && (t_user ≥ t_process)) 4. ((resourcecAvail == false) && (serverAppl == true) || ((resourcecAvail == true) && (serverAppl == false) || ((resourcecAvail == false) && (serverAppl == false)
Action	One of these action will be taken according to the pre-condition: 1. On the pre-condition #1: Get("eMule-installer.exe", http://sourceforge.net, t_user ≤ 10) Check(resourceAvail, serverAppl) Compute(t_process) Compare(t_user, t_process) 2. On the pre-condition #2: NormalDL("eMule-installer.exe", http://sourceforge.net, t_user ≤ 10)) (SaveFile("eMule-installer.exe", MyDirectory, true)) 3. On the pre-condition #3: ((Retry("eMule-installer.exe", http://sourceforge.net, t_user ≤ 10)) 4. On the pre-condition #4: (BackgroundDL("eMule-installer.exe", http://sourceforge.net, t_user > 10)) (SendEmail("eMule-installer.exe", ratna@uny.ac.id, true))
Post-Condition	The post-condition consist of one or two possible state. How a possible post-condition is selected is defined by a result of parameter evaluation: 1. If (t_user ≤ t_process) then NormalDL("eMule-installer.exe", http://sourceforge.net, t_user ≤ 10)) Or ((Retry("eMule-installer.exe", http://sourceforge.net, t_user ≤ 10)) 2. If t_user ≥ t_process) then (BackgroundDL("eMule-installer.exe", http://sourceforge.net, t_user > 10))

V. Conclusion

Implementation of QoS framework on Internet applications and a series of tests show that the design of QoS framework is:

a. Complete, the methods and parameters defined in the framework are sufficient for all required specification of interaction.
b. Realistic: it is possible to implement the QoS framework on the application.
c. Flexible: QoS framework can be implemented as a mediated layer or integrated into the application.
d. Orthogonality: QoS framework capable of handling dynamic access behavior by applying the alternative access mechanism using different application. This is possible because mechanisms of interaction between the QoS framework with an application protocol defined in one component. The addition of new components or properties within the framework to handle different application scenarios as well as new applications that want to use the mechanisms provided by the QoS framework can be done easily.

References

[1] Babulak, E. (2002). "The IT Network Quality of Service Provision Analysis in Light of The User's Perception and Expectation". Proceedings of Third International Symposium on Communication Systems Networks and Digital Signal Processing, Sheffield Hallam Univ. Press Learning Centre. pp. 5-8. [Online] Available: http://www.soc.staffs.ac.uk

[2] Bouch, A.; Kuchinsky, A.; Bhatti, N. (2000). "Quality is in The Eye of Beholder: Meeting user's requirements for Internet Quality of Service". Proceedings of the SIGCHI conference on Human factors in computing systems. [Online] Available: www.hpl.hp.com/techreports/2000/HPL-2000-4.pdf

[3] El-Khatib, K. M. (2005). "A QoS Content Adaptation Framework for Nomadic User". Ph.D. Thesis, School of Information Technology and Engineering University of Ottawa, Ontario, Canada.

[4] ITU-T Recommendation E.800. (1994). "Term and definition Related to Quality of Service and Network Performance Including Dependability".

[5] Prevost, J. (2001). "A Reliable Low-Bandwidth Email-Based Communication Protocol". Master's Thesis, Massachusetts Institute of Technology 2001.

[6] Ray, G. (2000). "Quality of Service in Data Networks: Products". [Online] Available: www.ohio-state/~jain/cis788-99/QoS_products/index.html

[7] Scaefer, C.; Enderes, T.; Ritter, H.; Zitterbart, M. (2002). "Subjective Quality for Multiplayer Real-Time Games". Proc. First Workshop on Network and System Support for Games, Braunschweig, Germany. pp. 74-78.

[8] Sutinen, T.; Ojala, T. "Case Study in Assessing Subjective QoS of a Mobile Multimedia Web Service in a Real Multi-Access Network". Proceeding Thirteenth International Workshop on Quality of Service, Passau, Germany. pp. 298-312.

[9] Venkateswaran, R.. (2002). "Network QoS and IP Telephony".

[10] Ye, N.; Chen, Y.; Farley, T. (2003). "QoS Requirements of etwork Applications on the Internet". Department of Industrial Engineering, Arizona State University, Tempe, AZ, USA.

Ratna Wardani is Ph.D student in Department of Electrical Engineering at Gadjah Mada University. She received a B.S. degree in Computer Science from Gadjah Mada University in 1995. From the same university she received her M.S. degree in Information Technology in 2003. Currently she is working in Department of Electronic Engineering at Yogyakarta State University as a lecturer.

Dr. Lukito Edi Nugroho is a lecturer at Department of Electrical Engineering and Information Technology, Faculty of Engineering, Gadjah Mada University, Indonesia. He holds a Ph.D. degree in mobile computing from school of Computer Science and Software Engineering at Monash University Australia, a M.S. degree in computer science from James Cook University of North Queensland Australia, and a B.S. degree in electrical engineering from Gadjah Mada University Indonesia. His research interests are in mobile computing, software engineering and information systems.

Dr. Ahmad Ashari is a lecturer at Department of Physics, Faculty of Mathematics and Natural Sciences, Gadjah Mada University, Indonesia. He holds a Ph.D. degree from Vienna University of Technology Austria, a M.S. degree from University of Indonesia, and a B.S. degree from Gadjah Mada University Indonesia. His research interests are in data communication and computer network, Internet and World Wide Web, and distributed system and parallel computation.

Dr. Risanuri Hidayat is a lecturer at Department of Electrical Engineering and Information Technology, Faculty of Engineering, Gadjah Mada University, Indonesia. He holds a Ph.D. degree from Chulalongkorn University Thailand, a M.S. degree from Chulalongkorn University Thailand, and a B.S. degree from Gadjah Mada University Indonesia. His research interests are in signal processing, telecommunication and computer network.