

THE CHALLENGE OF QOS VERIFICATION

Jan de Meer

GMD-FOKUS Berlin, D10589, Germany

jdm@fokus.gmd.de

Son Vuong

University of British Columbia, Vancouver, B.C. Canada V6T 1Z4

vuong@cs.ubc.ca

Abstract Quality of Service Engineering deals with the management and control of distributed applications coping with multiple media. Controlling concludes the discriminating of qualities obtained and qualities specified. Thus, this is a kind of verification. Since qualities describe how things have to be done, the verification of qualities depend on the running environment. Conformance between a specification and its realization is necessary but not sufficient. In the OSI framework conformance verification methodologies - by testing - have almost maturely been developed. Quality verification methodologies - by controlling - are lacking. Thus, we have to investigate into the requirements of the new multimedia and high speed communication technology. These technologies are the means by which new systems and services will be built. The actual verification of the Quality of Service (QoS) depends from the embedding system and thus can only be checked during system operation time. In order to understand and to identify verification mechanisms and architectural observation or control points, a generic QoS model is envisaged. The new model is based on the basic concept of continuity respectively of processes in order to model continuous behaviour, i.e. streams adequately.

Keywords: Quality of Service, Multimedia Services, Conformance, QoS Verification and Testing, Continuity

1. INTRODUCTION

The liberalisation of the telecommunication market and the wide-spread availability of communication technology has emerged as an enormous mar-

ket for new various multimedia services to be realized and verified imposing adequate techniques and methods. The new services are expected to be easy deployable and customizable in order to match with users' flexibility. Furthermore, these services involve more and more continuous interaction using streams of data preferably in one direction and to be distributed to more than one customer. So, the advent of new technology will likely change the current paradigm of conformance verification from a discrete point-to-point view to a continuous multi-point view.

A new notion seems to be required to adopt the increase in dynamics and flexibility of communication. The new notion in mind is continuous behaviour and its verification, related to QoS testing.

The paradigm change might best be described by looking to the current trials of introducing flexibility to communication protocols. It begun with parameterization of the point-to-point network and transport protocols with so-called service classes. By such a service class, protocol service users are allowed to make - prior to the use - a selection on optional protocol functions, i.e. retransmission of lost protocol units, fixed throughput classes, etc. This is comparable to the invention of differentiated services to internet service users. Differentiated services provide service classes ranging from best effort to some type of guaranteed services.

In conventional transmission services time-consuming protocol functions like retransmission or time-outs have been possible, because transmission was not constrained by strong time requirements. Contrarily, for the new multimedia transportation services, timeliness and continuity became indispensable qualities. Hence, any appropriate QoS verification methodology has to go far beyond pure functional checks. Advanced transportation environments may include adaptation mechanisms to deal with QoS guarantees. Dynamic features like those, are based on permanent observation and control. Consequently, QoS verification, respectively testing integrates existing conformance testing techniques with those of continuous observation and controlling. Observation copes with sensors being introduced at appropriate component or object interfaces being fed-back to controllers. Controlling copes with service adaptation using specific traffic models or resource usage policies.

In the area of QoS, standardization bodies (i.e. ISO/IEC, ITU-T, ETSI) have dealt with three documents, the *Basic Framework of QoS* [QoS95], *Methods and Mechanisms* [QoS95] and *QoS in the ODP RM* [QoS98]. The basic framework document is restricted to the OSI basic reference model and thus, to a layered communication philosophy. Consequently it characterizes connection-oriented and connection-less transportation qualities. The identified QoS functions are assigned to layered entities. End-to-end control of stream-based distributed objects is devoted to the QoS ODP RM related document under way. The document *Methods and Mechanisms* provides with a precise description of QoS

negotiation mechanisms. In all provided documents the verification respectively testing of QoS is not really addressed. Initial work is provided by the ODP RM by which conformance reference points in systems architectures are identifiable.

2. THE CHALLENGE OF QOS VERIFICATION

The presented work is aiming at the notion *quality of service verification*. By the invention of massive and continuous service application the concepts of verification respectively testing are to be re-considered and readjusted.

The phenomenon of mass service application changes the paradigm of communication from discrete message-based type to continuous stream-based type of interaction, from point-to-point distribution to multi-casting, from mono-media to multi-media. Conformance testing respectively verification techniques by which the quality of applied services shall be checked must take into account the effects of continuous processing of a stream and the mass of units transported by the stream.

By considering the above mentioned interconnection protocols, prior to deployment the protocols are to be *verified* by various conformance testing methodologies. Confidence about the protocol functionality is gained by the set of accepted test runs. Conformance Statements about the mass transfer of transported data units in a multicasting environment however, is not captured by the conformance testing approaches. In a massive and continuously communicating environment, quality constraints will replace discrete event assertions. Additionally, multicasting communication topologies is a second constraint which makes the adaptation of available verification and testing techniques to mass and continuity requirements indispensable.

The term *verification* is rather overloaded by many research activities in the telecommunication domain. Verification is mainly realized by testing, monitoring and system management techniques. Even so, it does not exclude verification techniques provided by formal models. In this work however, we would like to adopt the view on verification of QoS from the concept of permanent observation of systems in operation. In case of deviations from contracted quality during service application, a modification action has to be taken, which either tunes the service to an allowed degradation quality or provides with sufficient resources to maintain a given level of quality.

Consequently, a fundamental step needs to be taken from protocol conformance testing towards QoS verification. The observation and controlling techniques and concepts for individual message-based systems are discussed in comparison with the features of systems and services serving for mass communication. Testing and QoS verification concepts become built-in features of the envisaged QoS-aware systems. The effects of timeliness, which one of the

major factor to determine the service quality, is analysed by this work. Finally, a synthesis is offered which integrates some of the identified factors of influence into a unified model of QoS. From the integrated model, language concepts can be derived to describe mass effects, their dynamic qualities and techniques for their permanent observation and control. The integrated model will serve for the analysis and simulation of the end-to-end control mechanism of the service quality with respect to perception, mass distribution, timely delivery, safety and security. It provides means to check the effects of advanced quality control and adaptation mechanisms.

3. CONFORMANCE CONTROL FUNCTIONS

Since protocol conformance testing is a *prior-to-deployment* verification technique the expected behaviour in terms of events and responses compared to a specification is tested. Systems which rely on stream-based communication between autonomous components are interfered by many factors like the variation of background traffic. Hence, the quality of operation is to be controlled continuously during application of a service. Any observation and controlling activity with respect to QoS conformance must thus be performed on-line.

In figure 1 one can see that conformance testing of event-based systems may involve streams, but there are important difference to continuously operating systems. The streams in event-based systems do not continuously flow, they are bound to a very limited number of allowed units to be transported. The openness of a stream, i.e. the determination of the beginning of the stream and his end is - in event-based systems- restricted to a countable and fixed number of discrete events allowed to happen, respectively transportation units allowed to flow. In the flow control protocol component the units which can flow are determined by the size of a so-called window. The window is opened or closed by acknowledging or unacknowledging messages received. The controlling and decision-taking mechanism is implemented at the protocol server side and consists of a flow control buffer *FCB* and a component which decides on the information gained from the connected protocol client.

For the purpose of conformance testing, in figure 1, the role of the protocol client is played by the conformance tester. Notice, here we refrain from explaining the variations of the conformance testing architecture which provide with lower and upper tester roles, depending on the interface of the implementation observed. Instead, we restrict ourselves to the interactions between the protocol server and client which are observable at the so-called *lower interface*, where protocol operations are made observable. The tester connected to is thus also called *lower tester*. In order to check the flow control mechanism the tester can send and receive streams of a length up to the pre-defined size of the window. To the units transferred a sequence number is added to determine losses,

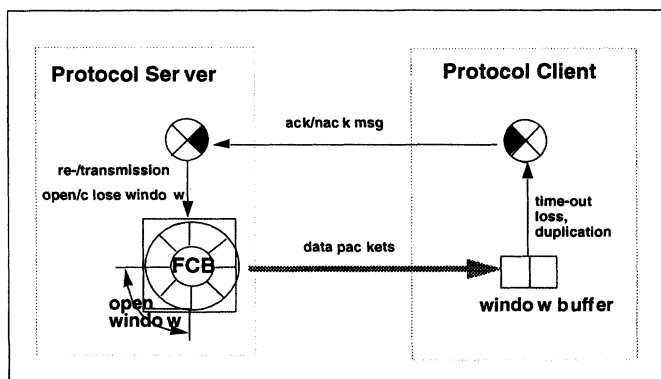


Figure 1 Discrete Event-based Interaction

duplications or misorders. Misorders and duplications can easily be resolved at the receiver's side by applying reordering the sequence or just pruning the duplicated units. Units are lost, if they do not arrive in fixed limit of time, a retransmission must be requested.

In all these cases conformance testing prior to deployment is still possible because the events are limited in time, even the event of a data unit loss virtually occurs by the elapsing of a timer, independent of the message eventually will be received after.

3.1 QOS CONTROL FUNCTIONS

In the ODP reference model it is said that events just happen which means that their beginning and ending are observable. Streams however, cannot be simply cut-off by time-outs or limited to a pre-defined length. The receiver must continuously be able to consume the stream of units over time. Furthermore, system inherent interference quantities will influence the quality of continuously ongoing behaviour. Continuous behaviour is represented by processes behaving over time and thus represented by the symbol $q(t)$. Since streams have also a direction and transport a mass of units, any assessment operation of streams must deal with all three properties, timeliness, direction and mass of transportation.

Streams are dynamic processes which behaviour varies over time and hence cannot be tested as simple events. Instead they must be observed and assessed continuously during operation. Consequently, conformance testing approaches for QoS must be developed towards a process-oriented quality assessment strategy, which means the extension of current conformance testing methodologies from single event observation to continuous process observation.

The operational quality of a system or a system component depends on many factors related to its real-time environment, like the current traffic in the network, or the actual utilization of resources. By verification prior to deployment the correct parameterization of the set of possible interactions and the causality relations between the interactions are tested. Verification after deployment, i.e. during operation, must deal with certain interference quantities like traffic, load conditions, buffer over or underflows etc. One can say pure conformance testing provides with structural and relational inter-operability while the quality of service verification provides with inter-operability under varying embedding conditions.

As it is illustrated in figure 2, the QoS control mechanism, or in other words the protocol of a stream-based system can in case of figure 2 be influenced from a single service client. In other cases this influence will be issued by the load which represents the a group of clients sharing e.g. the stream server. The stream server comprises a multimedia store and may serve the requests of many clients. In order to avoid overloading with requests, the stream server has a built-in admission control mechanism. The admission controller balances the capacity of the server by the number of out-streams requested for several clients. For this purpose the admission controller is able to vary the deadlines of the out-streaming units. In order to decide which deadlines of which stream are to be changed the admission controller needs further information from his clients. For example the urgency of a stream to be served. Thus, the deadlines of less urgent streams can be changed to a lower frequency. Or, clients may have problems in presenting the in-streaming units, thus wishing a degradation of a stream unit rate. This might be signalled to the server's admission controller, which must decide very rapidly in order to avoid losses of stream units.

A feedback link from the client's in-stream buffer to the server's admission controller might thus be realized to signal decision-making information. The continuous nature of stream-based systems induces the cooperation between the clients and their servers.

At each site, at the server's and at the client's site, locally functioning control mechanisms are installed. At the server's site the accuracy of the stream units' deadlines must be under permanent control. The accuracy of the server's deadlines influences the quality of the inter and intra-stream synchronisation at the clients' sites for example. There, the in-streaming units must be buffered prior to presentation to accommodate the differences in speed between the incoming streams and the presentation capabilities of the multimedia devices. Points of observation are installed to observe the possible skipping of stream units or their pausing. These observations are translated into a frame rate changing signal to the server, which must either speed-up or slow-down the individual streams.

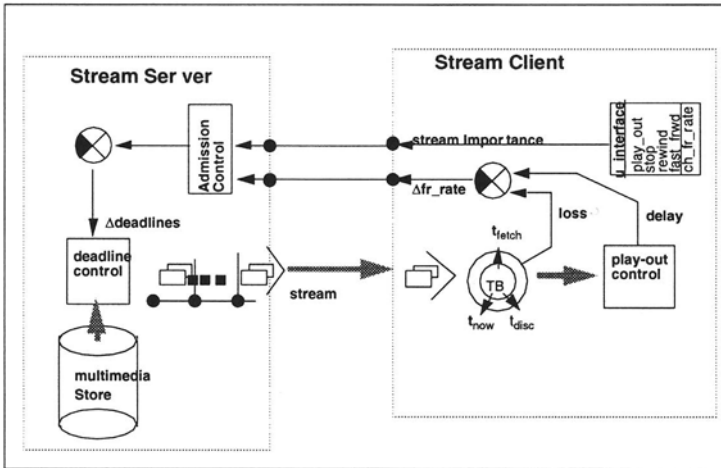


Figure 2 Continuous Stream-based Interaction

In such a self-controlling environment, the QoS specification is provided by a special specification, called the QoS contract. It contains all the values, thresholds, limits and bounds of QoS parameters agreed among a server and of every one of his clients. In case of figure 2 the QoS parameters comprise the deadlines set for the out-stream, the accuracy of the presentation device of each client or the minimum QoS of presentation to be achieved, etc. From a conformance point of view these dynamic processes must be tested. Since, the architecture provides with a closed loop the testing components must be inserted into the loop such that controlling or measurement signals can be generated. This will be the case when replacing the so-called discriminators of figure 2 by testing components which are able to take over the discriminators' roles.

In opposition to the event-based testing methodologies applied to configurations like those of figure 1, where testing components replace peer-entities of a considered communication protocol, in stream-based systems, all the communicating peer-entities will not be replaced. The system under test will not be changed, instead it must provide with interfaces at special points of interest internally to the system under test, e.g. the measurement points at the clients, or the steering and admission control points at the single server. QoS-aware system components are tightly coupled. Thus testing is not done by simply playing the role of peer communicating components but by a general purpose external signal generator, that intrudes signals in order to change global behaviour of communicating components. In practical terms the discriminator must provide with a second interface at which the externally generating testing signals will be accepted.

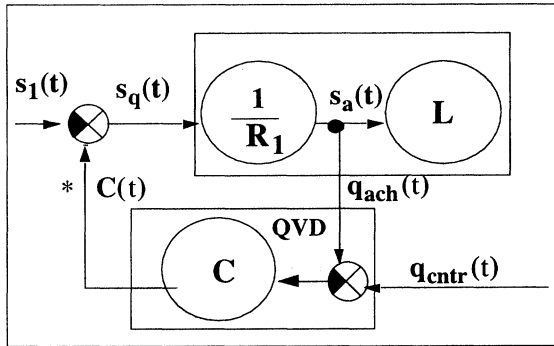


Figure 3 Integrated QoS Model

4. CONTINUOUS VERIFICATION MODEL

To achieve time-independent quality of presentation and performance of competing processes, it is necessary to scale, to customize and to coordinate the access to and usage of global available resources. Service users and service providers together with the underlying transportation service, need all the negotiation of their QoS requirements during connection set-up. And once the connection is established a facility for controlling the contracted QoS is required. Therefore conformance verification techniques and an architecture, that supports actively the control of QoS are needed in order to guarantee the negotiated level of quality during service application to all participating players. In the model, the streams flow - so to say - through the resource object R with a time delay but get interfered by factors, which may influence the stream quality. At the client site the achieved quality is compared with the contracted one and the comparison is fed into a controlling object C . This process informs a control mechanism at the server site and advises him which changes are to be taken to maintain the contracted quality. In figure 3 the resultant basic architecture is modelled. The streaming processes are indicated by lower case letters. E.g. s_1 and s_2 stand for two concurrent streams from the server to several clients via some transportation network. q_{ach} , q_{cntr} stand for the achieved and contracted qualities at the client's site. The resource characteristics of the streamed-through objects in the model are indicated by upper case letters. R and L stand for some resource providing objects; C is the characteristics of the controlling object being able to influence in-streams by providing a certain amount, proportional to C , of less or more resources.

Assume, at an observation point at the client's site, the achieved quality of the stream $q_{ach}(t)$ is gained. Then the observed quality must permanently be discriminated with the quality contracted $q_{cntr}(t)$, i.e. $q_{ach}(t) - q_{cntr}(t)$.

Since the consumption of resources will influence the quality of a stream, the discrimination process must act proportionally to the change of discriminated quality, i.e. $C * q(t)$. A stream passing a component, i.e. a network, at least consumes time which is considered to be a resource in terms of the model. Consuming time however, means between source and sink there is a certain amount of stream units in transition. The amount of units in transit can be calculated by measuring (testing?) the difference between the stream $s_q(t)$ fed into the network and the stream $s_a(t)$ received from the network at the client's site at some later time. The network must thus be able to store all the units of a stream during an interval of time e.g. $[0 : T]$. If the total quantity of stream units kept by a capacitive component would be stable, no jitter of transition delay per unit would occur. Since this is unrealistic, the amount of stored units depends from the system's respectively from the component's dynamic characteristic. It is a variable which must thus be kept permanently under control. In order to keep the achievable quality in acceptable limits, it may not be allowable to vary the size of stored units too much. For example, if the stream considered is a video stream to be presented with a certain contracted quality the play-out buffer must not be emptied less than a given lower limit. So possible delays occurring during decoding can be equated. From another point of view, such a system parameter can be said to describe the state of a system respectively of a component. The state is then the difference between the controlled stream $s_q(t) = Cq(t)$ and the achieved stream $s_a(t)$ having passed the system respectively the component considered:

$$\frac{1}{R} \int_0^T (Cq(t) - s_a(t)) dt$$

In the formula above, R represents the capability of a component to store a certain amount of stream units. The quotient of the storing capability R and the difference of the in- and outstreaming quantities for a period of time T describes the dynamically changing state of the component. The componental state is the result of two dynamic processes, i.e. the instreaming process and the out-streaming one. The quantification and hence the measurability of the state is expressed by the integral.

In a further step the QoS verification model will be improved by another state variable modelling delay and jitter. The quantity of a transmitted stream is effected by a time relocation of T time units which is proportional to the time of transporting a quantity z over a distance L at a speed v . Since, there are the dependable variables L and t , delay is modelled by the partial differential equation:

$$\frac{d}{dt} z(L, t) = v \frac{d}{dL} z(L, t)$$

In the integrated model of figure 3 delay is represented by a component with the characteristic L which represents a distance. The bigger the distance modelled the greater the delay. Notice, a stream z passes unchanged the component that obeys certain delaying characteristics. One can say the in-stream and out-stream processes are tightly coupled by so-to-say a "conveyor" of length L . Contrary, a capacitive component couples in- and out-streams only loosely, i.e. both processes may obey behaviour which is independent from each other.

5. CONCLUSIONS

New approaches of QoS verification and testing, must be applied to the emerging technology of communication including multi media and multi casting. The verification techniques for protocols based on discrete message passing must be adapted to the needs of continuous interaction relaying to streams. The major challenges are the system inherent properties of continuity and quantity. In continuous systems inter-action is not only a question of correct operation but also a question of the quality of operation. That is because quality depends mainly from resource consuming streams passing through components, respectively from loads of components. Hence, verification or testing of inter-operation is twofold. Firstly, inter-operation generally is interfered by the embedding conditions and secondly, it is a real-time matters. Prior-to-deployment checks are restricted to static and functional correctness, for which conformance testing techniques are infact appropriate. Quantitative aspects play just a minor role. Contrarily, in continuous systems, quantitative aspects, i.e. resource consumption cannot any more be neglected.

In this paper we have presented a model that is based on the concept of stream processing. A stream processing object is characterized by the properties of the resources it contains. The consumption or use of these resources is not constant over time. So, certain quality constraints to the system can not be verified without observing the consumption of these resources. Quantitative constraints expressed by limits, thresholds, bounds etc., instead of sequences of events are the basic elements of testing. Thus, testing translates into a permanent process of taking probes and comparing them with limits. In the former sections, the variability of componental states is expressed in terms of integrals and differential quotients. The integral represents the variation in accumulation, i.e. the summation of transported stream units and the quotient represents the variation in transportation, i.e. the delay. Since both characteristics are variable over time they are out of scope of discrete testing procedures and are thus matters of continuously taken measurements and evaluations. To this continuous activities we would like to apply the notion of QoS verification.

In an adequate model for QoS, architectural concepts like feed-back links, suitable points of observations and control and characteristics of resources like delaying or accumulation become new elements of consideration of QoS testing. This paradigm shift from conventional conformance testing towards QoS testing has been motivated by comparing the well-known scenario of a point-to-point message transportation protocols with the advanced scenario of continuously communicating components. Whereas in the massaging scenario flow control can be evaluated by a limited set of tests based on probes and observing reactions, in the continuous communicating scenario flow control can only be evaluated by permanent observation including quantitative measurement methods.

However, a complete QoS model has not yet been developed. We have started with an evaluation of testing methods for message passing systems with the belief that there is an evolution from the current event-based techniques towards the more sophisticated stream-based ones. For the development of new verification methods including QoS verification an extension of the anticipated QoS reference model in standardisation is envisaged. By means of this generic reference model, basically quantification of resources, which is much more than just accumulating or delaying as shown in the paper will be captured. The enhancements will deal with encoding patterns, relations between two or more streams, adaptation strategies on resource usage, negotiations among groups of applicants, etc. This requires sophisticated language and modelling concepts for which we gave a motivation.

While conformance testing usually is considered an off-line activity and which is executed prior to the service or system deployment, most work of quality of service verification must be done during the operation of an application or use of a service. An overall system is decomposed into the composite application and its supporting transportation environment. Both the transportation and service application environment comprise resources, which are under the control of the *Quality Verification Discriminator (QVD)* (see figure 3). This structure has been outlined by our early RACE2 project R2088 TOPIC [R2088]. A comparable approach has also been made available early by [FraHav94] and is known as the *Performability Manager*. The *QVD* component of the proposed model communicates with the observers that measure the resource usage, averages the values over the defined period and raises an alarm signal if the value falls below a specified threshold. In existing networks, whenever packets that flow between some application objects need more resources originally allocated to the transportation, then violate the QoS requirements, and thus will be discarded or must be retransmitted. In continuous systems those failures can be avoided by setting appropriate QoS thresholds and by implementing controlling and adaptation techniques in order to react on violation of QoS border lines.

Work is underway to set-up verification experiments with conferencing, joint editing, news-on-demand services, group communication techniques etc. Qualities will be evaluated on models and checked by measurements at implementations. The information gained will be used to improve the stream process model and to develop more advanced adaptation and decision strategies for quality verification.

References

- [Blair95] Blair, Coulson Lancaster University U.K.; Stefani, Horn, Hazard CNET France: Supporting the real-time requirements of continuous media in open distributed processing. CNIS Issue No. 27 1995.
- [CTMF94] ISO/IEC 9646, 1-9. IT- OSI - Conformance Testing Methodology and Framework 1994.
- [FMCT95] ISO/IEC P1.21.54.1,2 Revised Working Draft - Framework on Formal Methods in Conformance Testing, Guidelines on test generation methods from formal descriptions, 1995.
- [FraHav94] Leonard Franken, Boudewijn Haverkort: The Performability Manager. IEEE Network Reprint Vol.8, No.1, January 1994.
- [HdM95] Herman de Meer, University of Hamburg: Modelling and Management of Responsive Systems. The Quality of Service for multimedia Applications in High-speed Networks as an Example. Stifterverband fuer die Deutsche Wissenschaft.
- [ODP95] ISO/IEC 10918, 1-4: ODP Reference Model Part I - IV, 1995
- [Lazar93] Aurel A. Lazar, Columbia University: Quality of Service Control and Management for Broadband Networks. SICON93 Tutorial, Singapore.
- [NakTez94] Tatsuo Nakajima, Hiroshi Tezuka, Japan Advanced Institute of Science and Technology: A Continuous Media Application supporting Dynamic QoS Control on Real Time Mach, 1994 ACM.
- [R2088] RACE2 R2088 Project Toolset for Protocol and Advanced Service Verification in IBC Environments 1992 - 1994. Project Deliverables <ftp://ftp.fokus.gmd.de/pub/race/topic>.
- [QoS98] ISO/IEC JTC1/SC7 N1996 CD15936 IT ODP RM - Quality of Service, 1998-10-16.
- [QoS95] ISO/IEC JTC1.21.57.2/.3, January 1995. Quality of Service Basic Framework, 1995. Methods and Mechanisms, 1995.