Quality of service management for distributed multimedia applications

Gregor v. Bochmann
School of Information Technology and Engineering (SITE)
University of Ottawa, Canada

Contribution to the discussion at the Dachstuhl Seminar
"QoS in Networks and Distributed Systems"

May 2000

Abstract

The notion of "quality of service" (QoS) was first introduced to qualify properties of the service provided communication networks, such as loss rate, delay and jitter. However, it turns out that the "quality" experienced by users of communication services, such as teleconferencing, access to remote multimedia databases or video-on-demand, depends also largely on the performance properties of the end systems (client workstations and servers) that are involved in the application. In order to globally manage the QoS for distributed multimedia applications, it is therefore necessary to consider all the system components involved in the application (the communication network being only one of them). It is also important to consider the user preferences, since different optimization criteria may be used, including the trade-off between quality and cost.

The talk will give an overview of the lessons learned from two research projects, one on QoS management for applications involving distributed multimedia databases, and one on multimedia multicasting to large numbers of users (e.g. teleteaching applications). The issues related to the management, at the application level, of the user-perceived qualities are discussed (including cost) and avenues for future research are indicated.
QoS considerations within the overall system architecture

QoS considerations concern performance properties influencing the quality of multimedia presentations, as well as aspects of reliability and dependability, such as:

• **End-user wishes and requirements**: sound quality, video quality, colour rendering, and also **cost**

• **Client system**: limitations due to screen size and precision, audio equipment, operating system’s real-time response, available decoding software, etc.

• **Server system (continuous media file server)**: number of users, overall throughput limitations, access delay and jitter

• **Stored MM document**: encoded information structure, possibly scalable encodings (or several versions)

• **Network**: throughput, delay, jitter

General approach -- Issues considered

**Management aspects**: QoS negotiation should be seen in relation with distributed systems management, in particular resource allocation.

**Dynamic renegotiation of QoS parameters**: There are different reasons why the QoS parameters may have to be renegotiated

– changing user requirements

– changing network service availability (e.g. congestion)

– changing DB utilisation patterns,

– changing system configuration during group communication, e.g. people joining or leaving the group
Steps of the Negotiation Process

- **Defining the user requirements**
  Step 1: The QoS requirements are defined in terms of user-level parameters, e.g. CD quality (e.g. stored “user profile”). Note: This must be translated into the corresponding internal QoS parameter values in preparation for the subsequent phase.

- **Selecting a physical configuration for the functional units of the application**
  Step 2: allocation of functional units to physical system components (possibly consider several distribution alternatives)
  Step 3: negotiate and optimize the resource allocations
  Step 4: resource commitment

**Backtracking:** At each step of the negotiation process, backtracking should be performed if the user requirements cannot be obtained, possibly leading to negotiation with the user about the requirements.

Information for QoS Negotiation

- **Static management Information (used for choice of distribution architecture)**
  - Example: static information relative to the network component
    - maximum packet size, maximum speed supported, charging schemes, type of guarantees supported, network type, support of constant bit-rate and/or variable bit-rate, support of multicast or/and broadcast

- **Dynamic management information (used during resource allocation phase, and obtained through continuous application monitoring)**
  - e.g. current state of system components, load and congestion status, network path performance, reliability measures, accounting management, etc.

**Both types of information should be available from a local or distributed management information base**
Typical scenario (with QoS negotiation)

(a) Search of database to find MM document of interest
(b) Request of a specific MM document
   (1) Request for QoS parameters for specific document
   (2) Server sends QoS offer for each version of the MM document
(c) Verification of QoS compatibility and reservation of resources
   (3) Evaluating different configurations (based on static information),
       considering the local workstation constraints and the user’s wishes (as
defined by the active user profile); select best alternative
   If there is no suitable alternative, return to user for renegotiation (change
user profile, select a different document, or abandon session).
   (4) Activate corresponding database server (if congestion, backtrack to next
alternative)
   (5) Reserve transport connections (if congestion, backtrack to next
alternative)
(d) Display of MM document

Adaptation and renegotiation

... during the display of the MM document. Possible reasons:
• User-initiated renegotiation
  - to increase the presentation quality or reduce the cost
• QoS violation by the system
  - may be detected
    » by automatically monitoring the performance of certain system
      components (e.g. the network)
    » through the notification of a component indicating its inability to
      continue according to the agreed QoS (e.g. overload condition, failure)
  - may be recuperated by automatic adaptation (e.g. by selecting the next
    alternative distribution architecture)
  - otherwise the following alternatives are available
    » continue the ongoing session (with or without notifying the user about
      the problem)
    » renegotiate with the user (agreeing on lower quality or higher cost) and
      do the corresponding system reconfiguration
    » other application-dependent adaptation (including the final possibility:
      abandoning the session)
Another application configuration:

Multicasting

Example configuration: Two video qualities
A target application: “teleteaching”

Characterized by:

- Multimedia communication
- Very large number of receivers, only one sender
- Multicasting, possibly without restriction on number of receivers
- Different quality of service (QoS) requirements by different receivers
  - quality of video (frame rate, size, etc.)
  - quality of audio (e.g. telephone or CD quality)
  - cost (depending on selected presentation quality)
- Sender should not negotiate quality parameters with each receiver individually
- Possibly access to multimedia databases

Varying QoS requirements

Different receivers may have different QoS requirements, depending on

- hardware/software resources available at the workstation/terminal
- user preferences (cost vs quality, relative importance of video vs voice, etc.)
- available bandwidth and loss rate for transmission to the particular user (consider low-speed modem, wireless connections, etc.)
Demonstration prototype: Architecture

Use of standard protocols, if possible

- **Transport**: IP, UDP
- **IP multicasting**: the prototype uses the MBone system
- **Network performance monitoring**: RTP/RTCP
- **Session management**
  - knowledge about the participating users: RTCP
  - channel management for the different multimedia streams: Delivery Multimedia Integration Framework (DMIF) of MPEG-4 (with some small modification/extension)
- **Media encoding**: various standard voice and video coding schemes, plus extensions dealing with packet loss
**MPEG-4 and the DMIF**

- **MPEG-4**
  - low data rate
  - multiple media streams, e.g. one video stream for each moving object in a virtual environment
  - variable shape of video surface (depending on object shape, not only rectangular)

- **Delivery Multimedia Integration Framework (DMIF)**
  - hides delivery technology (e.g. from disk, over network) through the DMIF Application Interface (DAI)
  - in case of network delivery:
    - session-level protocol (a control channel, identified by a URL + several data channels, allocated to different media streams)
    - management of network QoS
    - lower interface to the network: DMIF-Network Interface (DNI)
    - Version 2 (under development) deals with multicasting
Ongoing research avenues

• Coding schemes for becoming more robust against packet losses
• Adaptive transmission rate control (as in TCP) for avoiding network congestion
• Experimenting with different policies
  – at the sender: deciding which kinds of stream variants should be sent (with fixed or adaptive rate control)
  – at the receiver: selecting the “best” available stream variant based on the high-level quality preferences of the user
• Experimenting in realistic Internet environments, including experimental differentiated services

Persuasion

• Example: a2 requests r4 to switch from quality 2 to quality 1
  – because only r4 receives quality 2 in the subtree of a2
  – offering a special (lower) price
• Simulation results (MSc thesis by L. Mekouar, U of Montreal)
Some other issues

• Scalability of resource sharing algorithms
• Resource sharing considering user preferences and benefits
• Different perspectives on optimization

Scalability for accessing electronic commerce catalogues

• Considering a very large number of ... clients ( n ) and servers ( m << n )
• Selection of server based on available QoS
  – QoS depends on: server load status, network connectivity and congestion status
  – Where is this choice made ?
    » by the server (may forward an incoming request to another server)
    » by an extended DNS (directory name server) which keeps information on status of servers)
    » by the client’s workstation (which gets up-to-date QoS information from the different servers)
• Complexity of distributed algorithm
  – number of messages per request (constant ?)
  – number of messages per QoS information update (linear in the number of servers, clients ?? )
A principle of resource sharing

- Principle: “Use limited resources to maximize the benefit”
  - how is “benefit” defined (different perspectives: users, network operators, service providers)
  - different priorities for users?

• Example

![Diagram showing resource sharing](image)

Benefits: Different perspectives

- User’s perspective
  - Given user’s benefit function: benefit (in $) as function of QoS obtained (this boils down to the question: what price is the user willing to pay?)
  - Optimize relative benefit: obtained benefit minus price to pay
    » based on tariff structure of networks and servers
    » also depending on adaptation possibilities of the application software and hardware

- Network perspective
  - Optimize profit = income minus operating costs
    » This leads to the network tariff structure (based on assumptions about number of users and the user’s benefit functions)