Network & Operating Systems Support for Digital Audio & Video

Thomas D.C. Little   Riccardo Gusella (Eds.)

Network and Operating Systems Support for Digital Audio and Video

5th International Workshop, NOSSDAV '95
Durham, New Hampshire, USA, April 19-21, 1995
Proceedings

Springer

Series Editors

Gerhard Goos, Karlsruhe University, Germany

Juris Hartmanis, Cornell University, NY, USA

Jan van Leeuwen, Utrecht University, The Netherlands

Volume Editors

Thomas D.C. Little
Multimedia Communications Laboratory, Boston University
02215 Boston, MA, USA

Riccardo Gusella
Hewlett-Packard Laboratories
1501 Page Mill Road, 94304 Palo Alto, CA, USA

Cataloging-in-Publication data applied for

Die Deutsche Bibliothek - CIP-Einheitsaufnahme

Network and operating system support for digital audio and video : 5th international workshop ; proceedings / NOSSDAV '95, Durham, New Hampshire, USA, April 19 - 21, 1995. Thomas Little ; Riccardo Gusella (ed.). - Berlin ; Heidelberg ; New York ; Barcelona ; Budapest ; Hong Kong ; London ; Milan ; Paris ; Tokyo : Springer, 1995
(Lecture notes in computer science ; Vol. 1018)
ISBN 3-540-60647-5
NE: Little, Thomas [Hrsg.]; NOSSDAV <5, 1995, Durham, NH>; GT

CR Subject Classification (1991): D.4, C.2, H.5.1, H.4.3

ISBN 3-540-60647-5 Springer-Verlag Berlin Heidelberg New York

Typesetting: Camera-ready by author
SPIN 10512287 06/3142 - 5 4 3 2 1 0 Printed on acid-free paper

Preface

The 5th International Workshop on Network and Operating Support for Digital Audio and Video commenced on April 19, 1995 with an audience of over 80 researchers from industry and academia.

The Program Committee selected 23 of the 101 submissions to form the basis of eight technical sessions. In each session, the Chairs dedicated a generous amount of time for discussion. Although there were some differences from session to session to accommodate the nature of the topics addressed and the preferences of the Chairs, in general the discussion centered on a round-table consisting of adjunct papers and various invited speakers.

While the technical program looked at the state of the art in networking and operating system support for multimedia, a ninth session took a broader look at the NOSSDAV workshop charter. Starting with what the workshop has achieved over the last five years from its origin at the International Computer Science Institute, this session explored possible directions for future NOSSDAV workshops.

Because of the limited attendance and the intimate setting of Durham, New Hampshire, the workshop proved to be an exciting and fruitful learning experience for all participants.

We are indebted to our Program Committee, and in particular to the Session Chairs for the work that went into the planning and realization of the workshop. Special thanks to Dinesh Venkatesh and Mary Hendrix for their invaluable support in workshop registration and Proceedings publication. We would also like to acknowledge the support of the IEEE Communications Society, the workshop sponsor, and ACM SIGCOMM, SIGGRAPH, SIGOPS, SIGMM, and SIGIR. Finally, our appreciation goes to Hewlett-Packard Laboratories and Boston University for the resources made available to us to realize the event.

April 1995 Thomas Little and Riccardo Gusella

Program Committee

Program Co-Chairs:

Riccardo Gusella	HP Labs USA
T.D.C. Little	Boston University USA

Program Committee:

Stavros Christodoulakis	Technical University of Crete
Domenico Ferrari	University of California Berkeley
Ralf Herrtwich	IBM Creative Multimedia Studios
Andy Hopper	Olivetti & University of Cambridge
Kevin Jeffay	University of North Carolina at Chapel Hill
Chuck Kalmanek	AT&T Bell Laboratories
Jim Kurose	University of Massachusetts at Amherst
Aurel Lazar	Columbia University
Derek McAuley	Cambridge University
Duane Northcutt	Sun Microsystems Laboratories
Guru Parulkar	Washington University
Steve Pink	Swedish Institute of Computer Science
Radu Popescu-Zeletin	GMD-FOKUS
K.K. Ramakrishnan	AT&T Bell Laboratories
P. Venkat Rangan	University of California San Diego
Jon Rosenberg	C|NET
Eve Schooler	California Institute of Technology
Doug Shepherd	Lancaster University
Cormac Sreenan	AT&T Bell Laboratories
Jean-Bernard Stefani	France Telecom/CNET
Daniel Swinehart	Xerox PARC
Hideyuki Tokuda	Carnegie Mellon University/Keio University
Jon Walpole	Oregon Graduate Institute of Science & Technology
Raj Yavatkar	University of Kentucky

Additional Referees:

Milind Buddikot	Washington University
Zubin Dittia	Washington University
Anand Krishnamurthy	Boston University
K. Lakshman	University of Kentucky
Christos Papadopoulos	Washington University
Gopal Raman	Washington University
Srinivas Ramanathan	HP Labs
Dinesh Venkatesh	Boston University

Workshop Participants

Kevin C. Almeroth	Georgia Institute of Technology
Mostafa H. Ammar	Georgia Institute of Technology
P. N. Anirudhan	Texas Instruments, Inc.
Vivek Bansal	C&C Research Laboratories
Riccardo Bettati	University of California
Ernst Biersack	Institut Eurecom
Paul Bocheck	Columbia University
Jean-Chrysostome Bolot	INRIA
Milind M. Buddhikot	Washington University
John Buford	University of Massachusetts
Andrew Campbell	Lancaster University
Shanwei Cen	Oregon Graduate Institute of Science & Technology
Navin Chaddha	Stanford University
Shih-Fu Chang	Columbia University
Mikael Degermark	University of Lule
Luca Delgrossi	IBM European Networking Center
Alexandros Eleftheriadis	Columbia University
Julio Escobar	Bolt Beranek and Newman, Inc.
Kevin Fall	University of California
Domenico Ferrari	University of California
Valérie Gay	Université Paris VI
David Goodall	University of New South Wales
Pawan Goyal	University of Texas
Amit Gupta	University of California
Wendy Heffner	University of California
Mary Hendrix	Boston University
Ralph G. Herrtwich	IBM Eurocoordination Multimedia
Wen-Jane Hsieh	K200
Kevin Jeffay	University of North Carolina
Michael B. Jones	Microsoft Research
Chuck Kalmanek	AT&T Bell Laboratories
Kiyokuni Kawachiya	IBM Research
Srinivasan Keshav	AT&T Bell Laboratories
Hiroshi Kitamura	NEC Corporation
Edward W. Knightly	University of California
Jim Kurose	University of Massachusetts
K. Lakshman	University of Kentucky
Monica S. Lam	Sun Microsystems Laboratories
Simon S. Lam	University of Texas at Austin
Hugh C. Lauer	Mitsubishi Electric Research Labs
Aurel Lazar	Columbia University
Ian M. Leslie	University of Cambridge
Peter Leydekkers	Bellcore & PTT Research

Christopher J. Lindblad	Massachusetts Institute of Technology
T.D.C. Little	Boston University
Philip Lougher	Lancaster University
John C.S. Lui	Chinese University of Hong Kong
Derek McAuley	University of Cambridge
Shih-Min Mao	K200
Andreas Mauthe	Lancaster University
Darren C. Meyer	Worcester, USA
Mark Moran	University of California
Tatsuo Nakajima	Japan Advanced Institute of Science and Technology
Gerald Neufeld	University of British Columbia
J. Duane Northcutt	Sun Microsystems Laboratories
Max Ott	NEC
Seungyup Paek	Columbia University
Gurudatta M. Parulkar	Washington University
Stephen Pink	Swedish Institute of Computer Science
Kurt Rothermel	University of Stuttgart
Olav Sandsta	Norwegian Inst. of Technology, Trondheim-NTH Norway
Brian Schmidt	Stanford University
Henning Schulzrinne	GMD Fokus
Brian Smith	Cornell University
Ralf Steinmetz	IBM European Networking Center
Scott D. Stoller	Cornell University
Dan Swinehart	Xerox Palo Alto Research Center
Hideyuki Tokuda	Keio University
Dinesh Venkatesh	Boston University
Harrick M. Vin	University of Texas at Austin
Werner Vogels	Cornell University
Jonathan Walpole	Oregon Graduate Institute of Science and Technology
Johan Weilbach	South African Broadcasting Corp.
Ian D. Wilson	Olivetti Research Laboratory
Lars C. Wolf	IBM European Networking Center
Raj Yavatkar	University of Kentucky
Hui Zhang	Carnegie Mellon University

Table of Contents

Session I: Advance Reservation Systems

Chair: Kevin Jeffay, University of North Carolina at Chapel Hill

To date, much of the work on resource reservation in a network has considered only the problem of reserving resources for a network connection that is to be used immediately; the so-called "telephone call" model of network service. For uses that involve a larger number of participants, such as distributed meetings and conferences, or regularly scheduled, recurring events, such as classes, it is highly desirable to be able to reserve network resources well in advance of their actual use. The first session of the workshop considered the problem of supporting advance reservations of network resources.

The first talk, by Lars Wolf from the IBM European Networking Center in Germany, presented on overview of the issues in advance reservation and presented a candidate architecture for supporting such reservations. The issues highlighted in Wolf's presentation included the need to specify the duration of a session that is reserved in advance to be specified and the handling of network failures that occur after an advance reservation is granted but prior to the use of the network. A key issue, and one that differentiated the subsequent presentations in this session, was whether or not network resources are a priori partitioned and reserved for advance reservations and immediate-use reservations. The trade-off is the familiar one between starvation avoidance and potential under-utilization of network resources. The conclusion of the authors was that either resources must be partitioned or the ability for one class of reservation to preempt the other must be provided.

The second talk, by Mikael Degermark from the University of Lulea in Sweden, presented an extension to the service model and admission control algorithm developed by Jamin et al. (presented at NOSSDAV '92 in San Diego, CA) for predicted service. The predicted service model was extended to require that all admission requests (including immediate-use requests) specify the duration of the session. Admission is then based on the resource requirements of reservations whose start times are greater than the present time and on the measured resource usage of active reservations. Simulations of the extended predicted service admission control algorithm showed that network utilization decreases modestly when advance reservations are supported. As expected, the decrease is proportional to the burstiness of the sessions reserved in advanced and the fraction of the overall capacity consumed by advance reservations. Moreover, the addition of advance reservations to the predicted service model preserves the model's property of providing higher network utilization than a guaranteed service model (also with advance reservation).

In the third and final talk, Amit Gupta from the University of California at Berkeley, presented an advance reservation scheme that is being implemented in Suite 2 of the Berkeley Tenet protocol suite. In Suite 2, reservations can be made for both immediate-use, indefinite length sessions, and advance, definite length sessions. It was conjectured that the important distinction to be made between resource reservations was not whether or not the starting time for a session was "now," but rather whether or not the duration of the session was known in advance. In Suite 2, network resources are partitioned into resources to be reserved for definite length sessions and resources to be reserved for indefinite length sessions (although for simplicity the case of indefinite reservations made in advance was not considered). The partitioning of resources is dynamic and is expected to change over time as the capacity required to satisfy reservations of one type dominates that of the other type. A distributed resource reservation mechanism based on a table of reservation intervals was also described. Simulations of the scheme showed that across a range of initial resource partitions, a larger number of sessions can be supported with advance reservations than without advance reservations.

A roundtable discussion following the three presentations focused on charging models for advance reservation systems, and the ability of the proposed Internet resource reservation protocol RSVP to support advance reservations. With respect to charging models, the "restaurant model," wherein reservations are typically respected with a high degree of certainty, and the "airline model" wherein over-allocation of resources is common, were proposed and discussed. It was clear that unknown factors such as how abundant or scarce network resources are likely to be in the future, will determine the outcome of this debate. With respect to RSVP, the discussion centered on the design mismatch between the apparent requirement for persistent state in the network to support advance reservation and RSVP's reliance on soft state.

Advance Reservations for Predictive Service[1]

Mikael Degermark[2], Torsten Köhler[2 3], Stephen Pink[2 3], and Olov Schelén[2]

[2]Dept. of Computer Science
Luleå University
S - 971 87 Luleå, Sweden
{micke,olov}@sm.luth.se

[3]Swedish Institute of Computer Science
PO box 1263,
S - 164 28 Kista, Sweden
{steve,tk}@sics.se

Abstract. We extend a measurement-based admission control algorithm suggested for predictive service to provide advance reservations for guaranteed and predictive service while keeping the attractive features of predictive service. The admission decision for advance reservations is based on information about flows that overlap in time. For flows that have not yet started, the requested values are used, and for those that have already started measurements are used. This allows us to estimate the network load accurately for the near future. To provide advance reservations we ask users to include durations in their requests. We provide simulation results to show that predictive service with advance reservations provides utilization levels significantly higher than those for guaranteed service.

1 Introduction

Real time multimedia applications will share future networks with traditional data applications. To provide quality-of-service (QoS) for real time applications, it is likely that resource reservations will have to be made in the network. Current resource reservation protocols allocate resources just before communication begins, e.g., ST-2 [7] and various ATM signaling protocols reserve resources during connection establishment. This model of communication may not fit the needs of future network users, [6] pp. 44–45.

Resource reservations should be optional and decoupled from the starting time of the session. One should be able to reserve resources prior to or during a network session depending on when a specific service is needed. Users may know far in advance of their needs and would like to plan their activities by making advance reservations to ensure that they are not blocked by the network's admission control mechanism. Imagine some users with busy schedules in different time zones who want to have an important teleconference on a resource-limited network at an agreed time in the near future. They should be allowed to make an advance reservation given that they know when and for what duration their teleconference will take place.

[1] This work was supported by a grant from the Center for Distance Spanning Technology (CDT), Luleå, Sweden

In this paper we will look at an important candidate for an admission control algorithm originally proposed at this workshop some years ago [2] and later refined in [5], for a new kind of service called predictive service [1]. Predictive service provides quality of service for applications that can tolerate some loss such as real time digital audio and video applications that can adjust their playback points in response to jitter in the network. The efficiency gain of predictive service comes from allowing more flows into the network than guaranteed service, thus providing more sharing and lower cost. The architecture described in [1] supports guaranteed and predictive service, but not advance reservations.

The possibility of making advance resource reservations should be a part of a communication architecture to provide better service to the users. Whether advance reservations are actually needed depends on future resource scarcity. Where resources are plentiful, not even immediate reservations may be necessary, but where resources are scarce enough to justify reservations at all, it makes sense to be able to make them in advance. In this paper we will show that advance reservations can be provided by the network with little overhead.

2 Framework

The service model and the admission control algorithm suggested in this paper are extensions of those presented in [1] and [5]. In [1], the proposed network service interface offers guaranteed service, predictive service and best-effort (ASAP) service. The service interface is simple and relies on token bucket traffic shaping; the source specifies the bucket size b and the token generation rate r. Guaranteed service provides a minimum transmission rate and the queuing delay bound becomes the bucket size divided by the rate. Predictive service provides K different service classes with widely spaced target delay bounds D_i and it is suggested that the target bounds are spaced by an order of magnitude. The bounded quantity is the queuing delay per hop, so it is necessary to add up the target delay bounds at each hop to find the upper bound on the total queuing delay.

To support this service interface a scheduling algorithm is presented in [1]. The guaranteed service traffic is scheduled with weighted fair queuing (WFQ) [9] so that each guaranteed service client has a separate WFQ flow. All the predictive service flows and ASAP traffic share the spare bandwidth in a pseudo–WFQ flow, called flow 0. The available bandwidth for flow 0 is therefore $\mu - \hat{\nu}_G$ where μ is the link bandwidth and $\hat{\nu}_G$ is the measured bandwidth usage for all guaranteed flows over the link. Inside flow 0, there are a number of strict priority classes: one class for each target delay bound and ASAP traffic at the lowest priority. The strict priority scheme implies that queuing delay experienced by higher priority classes will be conveyed to lower priority classes.

Admission control is performed in each switch along the path of a flow. Admission requests will be carried to the switches by an end-to-end resource reservation protocol such as RSVP [8].

Note that our use of the term "guaranteed service" in this paper is adopted from [1]. There are other ways to provide guaranteed service which may give good utilization, e.g., jitter-EDD [3].

3 Duration Intervals

To achieve an efficient scheme for advance reservations we ask that each request includes a duration interval: $I = [t_s, t_e]$, where t_s is when the requested service will start and t_e when that service will end. The intervals are necessary to determine which requests overlap and when the reserved resources will be released.

We have extended the service interface so that each admission request includes a duration interval. Requests for immediate admission will specify *now* as their starting time. If a requested duration is too short, it should be possible to renegotiate the request by calling the admission algorithm again. If this request is rejected the session may continue but not necessarily with the same service quality.

If a requested duration is longer than needed, resources are over-reserved. This reduces the chances to grant admission to other advance reservations. Fortunately, immediate reservations can be granted to a large extent anyway. This is because the measurement procedure of predictive service automatically detects unused capacity once a flow is active. Therefore, over-reservation has little impact on the total utilization as long as there are some immediate requests for admission. In addition, there is an option for clients to explicitly close the requested service before the duration expires.

4 Admission Control Decision for Advance Reservations

The admission decision for predictive service is based on requested rates for flows that have not yet started and on measured rates for currently active flows. If there are no advance reservations and a request for immediate admission arrives, our extended conditions give the same result as the conditions stated in [5].

Figure 1 is a snap-shot of admitted flows in a time/bandwidth diagram. Flows a, b and c are currently active and we have measurements of their rates and maximum delays which are used as predictions of their future behavior. When a new admission request arrives, admission is granted if the new flow would not cause any delay bounds to be violated or bandwidth limits to be exceeded. The admission conditions only consider flows that overlap with the new flow (b,c,d,e,g,h), using measured bandwidth if they have started or, otherwise, bandwidth requests; we call this the *estimated* bandwidth. The conditions are checked at all points where new flows begin (t_s, t_x and t_y).

For reservations in the distant future the number of currently active overlapping flows is small and admission decisions are based mainly on requested rates. In the near future the number of currently active overlapping flows is probably

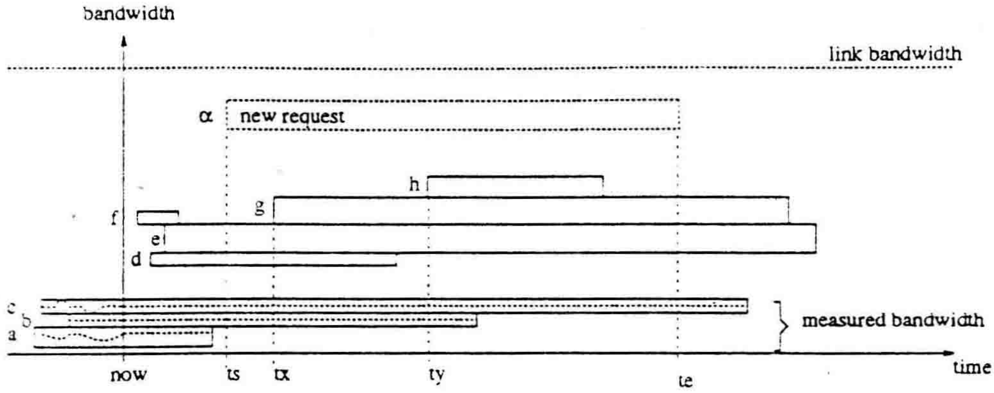

Fig. 1. Snap-shot of reservations

large and admission decisions are based mainly on measured values. So, in the distant future, the admission criteria are conservative, but as time proceeds more overlapping flows will become active and we get better estimates of bandwidth usage. Thus, as we get closer in time to the point at which a flow with an advance reservation is to begin, we have a more accurate knowledge of the network load and more flows can be admitted. Requests for immediate reservation can fill up the remaining bandwidth.

4.1 Admission Criteria

A client may request admission for predictive service in one of the classes 1 to K (where class K packets are scheduled at the lowest priority level), or for guaranteed service. The following notation will be used in the formulas[2] describing our admission criteria:

- $\nu_{G(t)}$ estimated bandwidth for guaranteed flows at time t
- $\nu_{P(t)}$ estimated bandwidth for predictive flows at time t
- $\nu_{P_i(t)}$ estimated bandwidth for flows in predictive class i at time t
- $R_{G(t)}$ requested bandwidth for guaranteed flows at time t
- $\hat{D}_j$ measured delay in predictive class j
- $B_j(t)$ bucket size sum for not yet started flows in predictive class j.

Predictive service: When a client requests service in predictive class k for a flow α, shaped by token bucket filter $(r_k^\alpha, b_k^\alpha, I^\alpha)$, the admission control algorithm performs the following checks:

– Determine if the bandwidth usage, after adding the new load r_k^α, will exceed the available link capacity $v\mu$ during the requested interval I^α:

$$v\mu > \max_{t \in I^\alpha} \left(r_k^\alpha + \nu_{G(t)} + \nu_{P(t)} \right) \qquad (1)$$

The available link capacity, $v\mu$, is determined by the link capacity μ and the link utilization target v, that is tunable.

[2] These formulas are extensions of those presented in [5]