Providing End-to-End guaranteed Quality of Service over the Internet: A survey on Bandwidth Broker Architecture for Differentiated Services Network

Priyadarsi Nanda and Andrew James Simmonds

Department of Computer Systems, Faculty of Information Technology
University of Technology, Sydney, Australia

E-mail: <pnanda,simmonds>@it.uts.edu.au

Abstract

The ongoing effort by the Internet Engineering Task Force (IETF) in recent years has resulted in two technologies for support of Quality of Services (QoS) in the Internet. They are Integrated Services Network and Differentiated Services Network, commonly known as int-serv and diff-serv. The scalability issue is considered as the major differentiator in the performance of these two technologies and has proven to be best in a diff-serv network. An agent-based architecture through the implementation of a broker between different networks is the central idea and is called Bandwidth Broker (BB) which can be used in such diff-serv networks. Where efforts are still going on to provide a proper QoS, this paper presents a review of various issues related to the implementation of a Bandwidth Broker and proposes a revised scheme for achieving end-to-end guarantees for better QoS over the Internet.

Keywords: diff-serv, QoS, Bandwidth Broker, BB

1. Introduction

Currently the Internet provides Best-Effort (BE) service to a wide variety of users where all transmissions are considered equal without any guarantee of delivery. The transition from traditional applications such as data, e-mail and file transfer to more sophisticated types of applications such as voice and video over the same Internet infrastructure has encouraged researchers to devise methods for providing QoS to these new applications in a better way. The framework for new Internet architectures has been developed with an eye to support two main category of applications: one requiring timely delivery of services called Delay Sensitive and the other requiring services without loss of any information called Loss Sensitive applications. Hence we consider delay and loss of information as the two prime factors to minimize within an information flow and maintain satisfactory QoS amongst different types of applications. Also, the transition to accommodate these new emerging applications has resulted in differentiation among various services. Such differentiation can well be achieved once the Internet starts classifying different user applications on the basis of their performance factors e.g. delay, loss, throughput, bandwidth, etc. As part of the ongoing research in the area of QoS, the Internet Engineering Task Force (IETF) has come up with certain models and they are defined as the Integrated Services Network (int-serv) and Differentiated Services Network (diff-serv). The IETF and various other research organizations are working towards standardization of a future Internet where different services will receive different treatment according to their requirements. Such a model is essential to support a wide range of services as per their requirements differentiating the service needs of one from the other and has been referred to as Differentiated Services (diff-serv) Network.

The central idea of implementing a diff-serv architecture comes from the use of the Type Of Services (TOS) field of IPv4 or service type field in IPv6, consisting of 1 byte. This 1 byte of information has been used to assign a particular value to different traffic types through which different priority level of services amongst them can be established. Currently 6 bits are used to assign 64 different priority services and 2 bits are unused. This 1 byte information in the IP header has been redefined as the Differentiated Services Code Point (DSCP) which defines a base set of packet forwarding treatments called Per-Hop-Behaviors or PHBs. Packets generated by different users can be marked in their DSCP field and several differentiated service classes can be created. Such a scheme of handling packets can be viewed as essentially a relative priority scheme. The Internet is basically organized as an interconnection of individual networks called Autonomous Systems (ASs), each one under separate administrative control. Initially hosts may request their ISPs for allocation of resources to send their traffic over the Internet. The ISPs in return check corresponding resources as requested by the hosts and, if the resources are available, inform the hosts with a positive response, otherwise the request from the hosts is simply denied. Once the request from the host is accepted, the hosts must supply their traffic according to the initial agreement. In case extra traffic is sent by the hosts, it may either be dropped at the entry point into the network or else renegotiation between the hosts and the ISP must be done to accommodate this extra traffic. Hence the devices situated at the entry point (Ingress routers) can be configured to perform the following functions:

1. Traffic classification
2. Traffic aggregation according to the DSCP set by the users
3. Mapping DSCP into their respective PHBs
4. Traffic policing
5. Signaling the ISPs for resource allocation

Resource management plays an important role in order to support a proper QoS in the Internet. The diff-serv architecture can be designed to handle the issue of resource management in different ways. A two-tier architecture has been implemented to allocate resources to the users in two levels [1]. They are, intra-domain resource management and inter-domain resource management schemes. In intra-domain resource management, each domain is responsible for allocating resources internally using a certain scheme. As suggested in [4], a logical entity called Subnet Bandwidth Manager (SBM) can be employed in doing this job inside a domain. Then inter-domain resource management enforces some agreement between different domains to allocate resource. A bandwidth broker is considered as the logical entity in charge of resource management for allocating inter-domain resources. Various works related to such resource management issues are reported in the past [1,5,6,7] for achieving a desired QoS. This paper identifies the basic requirements to achieve QoS from the users perspective and how to achieve them without too many complexities in implementation. In the next section we provide a summary of these methods of implementing Bandwidth Broker. Section 3 describes our proposed scheme and a prototype test bed to implement them. In section 4 we provide our future work and conclusions in section 5.

2. Bandwidth Broker and resource management

A Bandwidth Broker (BB) is an agent designed to negotiate resource allocation between a user network and ISP, and resource negotiation between different ASs in support of QoS. Figure-1 shows different situations of the bandwidth broker in an interconnected networking system.

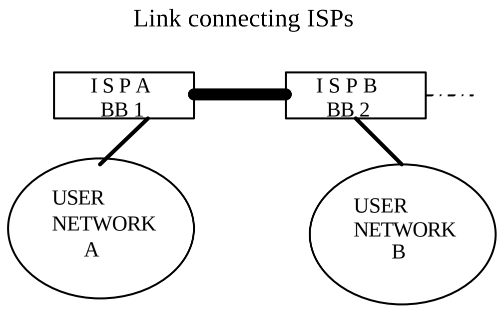

Figure-1. Negotiation of Service Level Agreement (SLA)

As shown above, a host in user network A wants to send information to a host in user network B. The user networks are directly connected to their respective service providers (ISPs) and the ISPs are connected to each other. Each ISP has an entity called a BB situated within it and these are the elements which resolve the following issues of the users in the network:

1. Management of internal resource distribution to the user network
2. Negotiation for resources with neighboring ISP

Also as shown in the figure above, there are two different types of negotiations taking place in order to provide the connectivity for flow of information between these two user networks. According to the diff-serv terminology, all the user networks and the ISPs are treated as diff-serv domains and these domains include the following components for proper flow of traffic within them as shown in figure-2:

a. Ingress router: The ingress router encounters the traffic flows entering a domain and this is the point where major control functions (shaping, policing, filtering etc.) are executed on the flows depending on the Service Level Agreement (SLA) of the flow specs.

b. Egress router: This is the point where traffic leaves the network and it may be necessary also to execute the same or a part of the control activities as are performed at the ingress router.

c. Core router: core routers are the intermediate ones between the ingress and the egress ones and are generally configured to have a greater capacity than the routers discussed above. Their basic activities are to collect and forward the traffic streams on the basis of the destination address in the packet header. In terms of functionality, apart from forwarding the traffic, necessary consideration must be given not to overburden the core routers, otherwise the QoS of time sensitive applications would be degraded further.

Figure-2 describes the location of these routers and their main activities in a diff-serv domain:

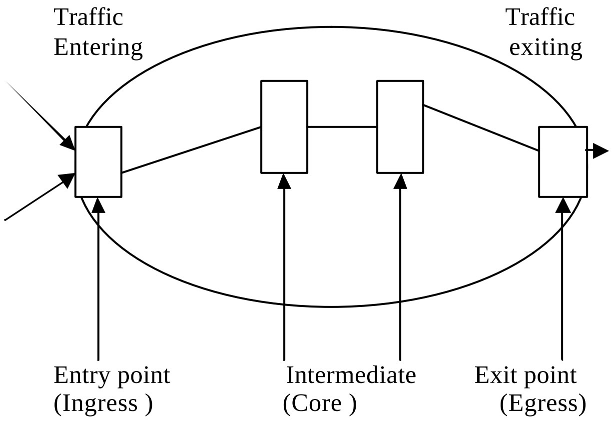

Figure 2. Location of different routers in a diff-serv domain

The diff-serv architecture relies on aggregated flow rather than individual per-flow traffic (as opposed to int-serv) passing through the routers. Such an aggregation is based on the DSCP as defined in the IP packet header. A database consisting of the traffic profiles and corresponding mapping to their DSCP may be maintained in the ingress router at the entry point into a diff-serv domain. This is also the location where after assigning the six bit code point to the packets of different traffic streams, the router does the job of aggregating (grouping) similar traffic flows and performs other jobs such as policing, traffic shaping and other related activities so as to ensure the flow agreement between the user and ISP is kept. After this the next router in the path, the core router, simply forwards the packets across the link until they finally exit through the egress router. In terms of functional loads, the core routers must be designed to handle less function in order to carry more traffic. If we overburden the core router by assigning the job of policing and shaping into it, it makes the whole process quite complex and ultimately degrades the QoS of the flows. Hence, it is always necessary to keep the design of the core router simple without overburdening it with such functions.

As we have stated earlier, resource management is one important aspect in order to achieve desired QoS in a heterogeneous environment thereby providing good scalability to any network architecture. The diff-serv architecture is designed to handle traffic flows on aggregated basis as opposed to per-flow in int-serv. This relieves the routers by maintaining less number of flow information within them. Hence diff-serv is the right candidate for this. Below we show how bandwidth resources can be well distributed amongst different candidates in the path of the traffic flows.

Consider a network connecting users serviced by an ISP for handling traffic flows in the Internet. The ISP containing our logical model of a BB basically monitors all the usage of bandwidth. Hence any request for initial bandwidth allocation and subsequently any request for extra bandwidth are only to be dealt with between the BB and the device requesting them. In this case the requesters are the two routers situated at the two extreme ends. Assume the total bandwidth available within the ISP for allocating to different user networks be M megabits. Assuming n different networks to be serviced by this ISP then gives a fair share as M/n Mbps. We have stated here the initial condition of allocating the bandwidth available within the ISP equally amongst those networks. However it may be worth mentioning at this point that in reality ISPs only allocate bandwidth as and when required by the users, resulting in an un-equal distribution, which is different from the one we are considering at the moment. Hence in our case each network gets an amount of bandwidth equal to M/n Mbps. For maximum flows, this is the amount the egress router always confirms on the flow leaving the network. For simplicity, consider the network to support only two classes of traffic streams, the Explicit Forward (EF) type and the traditional Best Effort (BE) type; this M/n Mbps is divided between them. Assume further that, x% of the total resource is reserved for EF type with highest priority, then BE gets a bandwidth of (100– x)% of the total allocated resource. Such a bandwidth distribution amongst the applications is considered as a necessary but not sufficient condition for resource management. In order to define the sufficient condition, one needs to look at the traffic profile and characteristics of various traffic encountered in the process. Below we present a review on different techniques for implementing a Bandwidth Broker.

In [1] a mechanism that dynamically adjusts the level of inter-domain allocated resources according to the level of traffic crossing domain boundaries is described. Initially the BB situated in each of the domains allocates a pre-specified amount of bandwidth to their respective egress routers. If there is an increased demand for bandwidth encountered by the egress router from its domain, the egress router informs its BB. In response to this the BB talks to the neighboring domain BB and finally this BB checks with its Ingress router about the extra bandwidth. If the ingress router in the neighboring domain after checking its internal resources, gives a positive response to its BB and in reply this BB replies positively to the BB of the requesting domain, the Egress router in that domain can go ahead accommodating the extra traffic; otherwise a denial message is sent to the source. Likewise a series of agreements between the Bandwidth Brokers of different domains will be required in an end-to-end basis.

In [5] the granularity of the agreement between the Bandwidth Brokers of different domains has been considered as a key issue of the overall performance evaluation of diff-serv through signaling techniques. Here the authors present three different techniques through which the negotiation process between the BBs can be handled. They are described as No notification, End-to-End notification and Limited notification. No notification also called the coarse-grained approach, reserves resources between the Bandwidth brokers situated in the domain and its neighboring domain and no further notification is propagated. Such a scheme does not allow end-to-end guarantees, which circumvents the scalability issue. However it is suggested that the ISPs would measure the diff-serv traffic and overprovision their networks and their agreements with the adjacent networks to ensure adequate end-to-end performance. This is called the adaptive reservation scenario. End-to-end-notification overcomes this which is similar to described in [1] but encounters the same complexity issues as int-serv and suffer from the same scalability problem. Limited notification takes the approach to decrease the granularity of no notifications. This is achieved by not providing each flow or change in an aggregated flow which triggers notifications and suggest only limited notifications to be sent to further brokers. A comparison between these three techniques has made through simulation and it is stated that the Limited notification approach to the bandwidth broker signaling is a favorable option.

In [6] it is stated that resource provisioning in terms of bandwidth to high priority diff-serv traffic such as EF type can be done by assigning 99% of the available bandwidth of the slowest link in the end-to-end path. Relating this scheme to the Bandwidth Broker implementation, the BB can configure the edge routers properly so that QoS can be achieved for different traffic. Such a scheme can guarantee QoS for the high priority traffic streams.

Similar work on Bandwidth Broker implementation and resource management has been discussed in [7-10]. [12] Considers a different approach to BB for resource allocation and propose a hierarchical structure called Clearing House (CH) architecture. It is a distributed architecture where each CH node monitors traffic and network statistics continuously, adapts aggregate reservations within and across multiple domains, maintains intra or inter-domain reservation status and performs admission control. Taking such a distributed approach reduces the complexity within the routers and a better QoS for different traffic streams can be achieved.

All the above-mentioned schemes in some way or other does not truly satisfy the QoS requirements of different traffic streams with their implementation. As suggested in [1,3,4], the schemes are statically implemented and hence do not scale well. [4] Proposes a SBM, but our belief is that by combining the functions of both SBM and BB into a single entity can simplify the scheme and result in a scalable, extendable hierarchy. In the next section we propose our prototype test bed for resource management through sophisticated BB mechanism in a diff-serv environment.

3.Proposed Bandwidth Broker implementation

This section of the paper presents our proposed prototype test bed developed for implementing a single tier Bandwidth Broker architecture and tries to investigate its feasibility in a global scenario. To make our scheme as simple as it can be, we propose to implement a hierarchical Bandwidth Broker [l] with a limited notification scheme [5].

Figure-3 gives the structure of our testbed. Initially we consider all the traffic generated by users is grouped into two PHB classes. They are EF and BE as discussed before. The BB situated in each domain initially allocates *x*% of the available bandwidth to EF class and (100-*x*)% to the BE class. The hosts generate different traffic and send them to their attached routers, which in our case is the Ingress router. Looking at the traffic profile, the Ingress router marks the traffic into two classes and groups it to forward across the core routers. Finally the Egress router situated at the other end of the domain forwards the traffic outside the domain to its destination. The BB maintains a client-server relationship with the edge routers in a domain. When extra bandwidth is requested by the host to send their traffic across the network, the Egress router requests its BB which monitors resource utilization within its own domain. This BB then talks to its immediate neighbor BB for the extra resource. If successful, the BB re-configures its domain routers and then traffic flows can be maintained across the network. In case the request is unsuccessful, the edge routers simply drop the extra traffic. The ingress router A in domain X and A1 in domain Y do the following activities:

1. Traffic differentiation (marking) by looking at the traffic profile contained in the IP header
2. Traffic aggregation
3. Traffic policing
4. Communicating with BB and Egress router

Routers B, C in domain X and B1 in domain Y are the core routers, which do the job of simply forwarding the traffics across the path. Router D in domain X and C1 in domain Y are the Egress routers responsible for sending the traffic outside their own domain. Also these are the routers who request the BB for any extra resources. The routers in each domain run Interior Gateway Routing Protocol (IGRP) for maintaining routing activities whereas the two domains communicate with each other by running Border Gateway Protocol (BGP). We have considered these two protocols for implementing in our model because of high network convergence. All the routers are initially taken as Cisco 2500 series routers, though we have a plan to upgrade our test bed to Cisco 7200 series after successfully implementing the first version of our BB algorithm. The BB will maintain an active database of the following and monitor resource usage:

1. Initial Bandwidth allocation to the routers
2. A table containing available resource and its location
3. A list of domains requested for additional resources

Also, as shown in the diagram, outside its own domain the BB only talks to the BB of its neighboring domain. This is as opposed to communicating with the egress router of the neighboring domain as given in [1]. The Ingress router in each domain separates the traffic streams into EF type (solid one way arrow) and BE type (dotted one way arrow). Also the Ingress and Egress routers in each domain communicate with their BB (two way arrow).

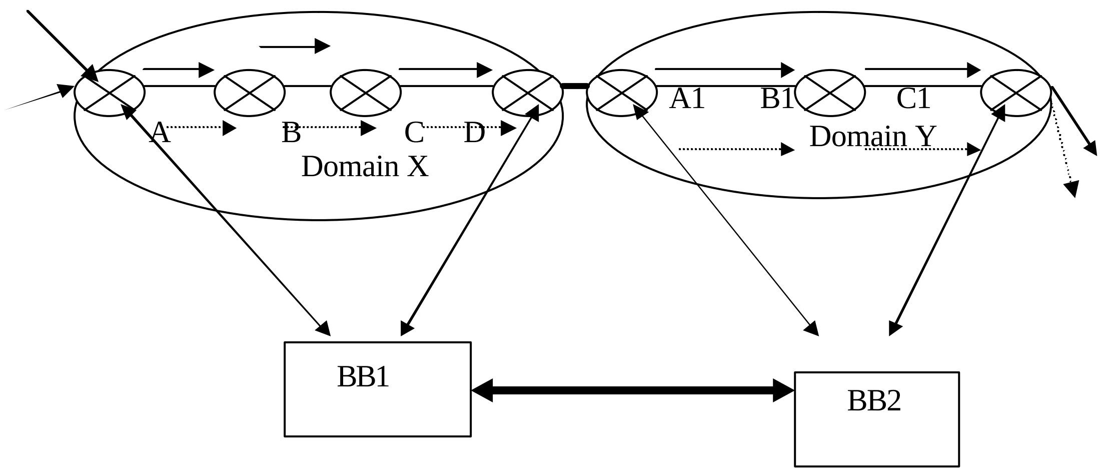

Figure 3. Prototype model

Though initial results of our implementation are expected, we are optimistic about providing a better end-to-end QoS through this single-tier BB implementation.

4.Future work

For the future we propose to carry out experiments with a wide variety of applications having a large number of new PHB as opposed only to the EF and BE type. Since the BB should maintain relevant information about all the routers in a domain, a suitable mechanism and protocol could be utilized to provide secure communication with the routers and a local database could be maintained for storing such information.

5. Conclusion

We have presented a summary of Bandwidth Broker and its relevance in resource allocation in a diff-serv architecture. Also, we have proposed our prototype model and BB architecture to provide an end-to-end QoS for different applications. Such end-to-end QoS management is essential for the future Internet that will carry a wide variety of user applications. Our solution is dynamic and we have simplified the BB scheme such that there is no SBM. We believe that our BB scheme can provide a scalable end-to-end QoS solution with a proper design and implementation.

Reference

[1] A.Terzis, L.Wang, J. Ogawa, L.Zhang, A Two-tier resource management model for the Internet, Proceedings of Global Internet, December 1999

[2] Xipeng Xiao and Lionel M. Ni, Internet QoS: A big picture, IEEE Network, March/April 1999

[3] Raju Rajan, Dinesh Verma, Sanjay Kamat,Eyal Felstaine, Shai Herzog, A policy Framework for Integrated and Differentiated Services in the Internet, IEEE Network, September/October 1999

[4] Raj Yavatkar, Don Hoffman, Yoram Bernet, Fred Baker, Michael Speer, SBM: A protocol for RSVP-based Admission Control over IEEE 802-style networks, IETF, Internet draft, January -2000

[5]Manuel Gunter, Torsten Braun, Evaluation of Bandwidth Broker Signaling, 7th International conference on Network Protocols (ICNP), October, 1999

[6] Volker Sander, Ian Foster, Alain Roy, Linda Winkler, A Differentiated services implementation for high-performance TCP flows, Computer Networks 34 (2000) 915-929, Elsevier Science

[7] M. Listanti, F. Ricciato, S. Salsano, Delivering end-to-end statistical QoS guarantees for expedited forwarding, Computer Communications 24(2000) 822-832, Elsevier Science

[8] Prashant Chandra, Allan L. Fisher, Corey Kosak and Peter Steenkiste, Network Support for Application-Oriented QoS, Proceedings of 6th International Workshop on QoS, pages 187-195, Napa, California, USA, May-1998, IEEE

[9] A. Bouch, M.A. Sasse, H.DeMeer, Of packets and People: A User-centered Approach to QoS, 2000 IEEE

[10] Venkata N. Padmanabhan, Using Differentiated Services Mechanism to Improve Network Protocol, Microsoft Research

[11] Zhi-Li Zhang, Zhenhai Duan, Lixin Gao and Yiwei Thomas Hou, Decoupling QoS control from Core Routers: A Novel Bandwidth Broker Architecture for Scalable Support of Guaranteed Services, ACM SIGCOMM '00, Stockholm, Sweden

[12]Chen-Nee Chuah, Lakshminarayan Subramaniam, Randy H. Katz and Anthony D. Joseph, Resource Provisioning Using A Clearing House Architecture, ACM SIGCOMM 2000-Poster Session, August 2000.