TOWARD AN INTELLIGENT BANDWIDTH BROKER MODEL FOR RESOURCES MANAGEMENT IN DIFFSERV NETWORKS

R. Nassrallah, M. Lemercier, D. Gaïti
Institut of Computer Science and Engineering of Troyes (ISTIT), University of Technology of Troyes (UTT), 12 rue Marie Curie, BP 2060, Troyes cedex, France
(nassrallah,lemercier,gaiti)@utt.fr

Abstract: DiffServ model is known to be the most used model to handle QoS over IP networks. Moreover this model has the advantage to be appropriate for use on large network contrary to the IntServ model that suffers from scalability problem. However, the DiffServ model has two major difficulties: routers' configuration and resources' allocation problems. In this paper, we introduce a new approach based on customers' Service Level Agreements (SLA) declaration. The resource allocation is done by a federal entity called Bandwidth Broker implemented using Web-Services. Our proposal avoids the use of signaling protocol between the Bandwidth Broker and the core routers when establishing a new flow. Thus, core routers do not have the responsibility to store the customers' traffics information and therefore, we respect the DiffServ model philosophy. Our tool provides the admission control and resource allocation management using overbooking techniques which guarantees the performances of priority traffics.

Keywords: QoS, DiffServ, SLA, Bandwidth Broker, Intelligent agent, Web-Services

1. INTRODUCTION

The IETF proposed two models to handle the QoS over IP networks. The Intserv model adopts a per-flow approach, which means that each traffic flow is handled separately at each router. thus, resources can be allocated individually to each flow using RSVP (Reservation Protocol). It has been recognized that such a per-flow approach is affected by scalability problems which prevents from its applicability to large networks.

DiffServ is the second model[1] and it responds better to the QoS problem over IP networks. It aims at providing QoS on a per-aggregate basis. It offers services differentiation mechanisms, which allow packets classification. The DSCP (DiffServ Code Point) is a six-bit field in the IP packets header. It allows the classification of 64 different service classes[2]. The DiffServ model defines some standard service classes. The Premium service is suitable for real time applications (Voice over IP, Videoconference) that need lower transfer delay and jitter. The Assured Service suites for non real-time applications. It is characterized by its reduced packet loss rate and its reasonable transfer delay. The best effort service is a none-guaranteed service. Best effort packets are always accepted in the network and they do not affect higher priority packets. They are the first packets to be lost in case of congestion. Finally, eight values of DSCP were reserved to assure the compatibility with the previous TOS filed in IPv4. Those values constitute the CS (Class Selector) service.

To assure theses services, the DiffServ routers must support a set of predefined behaviors called PHB (Per Hop Behavior). The internal routers handle packets according to the PHB identifier, and do not distinguish the individual flows. Then a Premium service is assured by the EF (Expedite Forwarding) PHB[3]. Similarly, the Assured Services are handled by the AF (Assured Forwarding) PHB[4].

DiffServ defines the network architecture inside a DS domain. Each domain is a set of interior routers (core routers) enclosed by another set of boundary routers (edge router). The edge routers handle the packet classification and the traffic monitoring functions. They control the incoming traffic to see if the access contracts are respected (flow, peak rate, packets size, etc.). However, the core routers assure basic functions such as queuing and scheduling according to packets priorities without having to know the contracts characteristics. Consequently, the DiffServ model pushes back the network management complexity to the edge, leaving relatively simple tasks to core routers.

The DiffServ operation can be guaranteed only if the incoming traffics respect a set of predefined constraints. Thus, each flow has to state its characteristics to enable an optimal configuration of the network devices. It is therefore of primary importance to assure the DiffServ policing at the level of the edge routers. The policing control takes into consideration certain pre-stated parameters. Many algorithms were proposed, however this debate seems to be closed because the IETF selected and published algorithms for this function. The Token Bucket (TB) is a standardized mechanism that allows identifying non-conforming packets[5]. It has two parameters (token depth b, token rate r), a queue for packets and a bucket containing b tokens. Each token represents the right to emit a byte. Thus packets arriving at the TB are conform if their size is equal or lower than the number of available tokens. The rate of the outgoing flow is fixed by r. In addition, if the number of token in the bucket is sufficient a small amount of packets could be emitted at peak rate flow. A packet is declared to be non-conform if there are no sufficient tokens. In this case a second mechanism takes place and chooses the intervention method to be applied. Three methods exist: maker, shaper and dropper.

The RFC 2698[6] proposes an algorithm "A Two Rate Three Color Marker" that handle three levels of policing control. The Two Rate Three Color Marker meters an IP packet stream and marks its packets green, yellow or red. A packet is marked red if it exceeds the peak rate. Otherwise it is marked either yellow or green depending on whether it exceeds or doesn't exceed the TB rate (r). The TRTCM is useful, for example, for ingress policing of a service, where a peak rate needs to be enforced separately from a committed rate.

In the core routers, packets can be buffered into different queues according to the DSCP. The queues being of limited size, a packet is rejected when it overflows. A mechanism of congestion control can be applied for each one of these queues. Several mechanisms were proposed to anticipate congestion problem: RED (Random Early Detection), WRED (Weighted RED), etc.

Each router of the network uses a scheduling policy to determine in which order the packets will be transmitted. There are several algorithms which aim at solving this problem. PQ (Priority Queuing), CBWFQ (Class-Based Weighted Fair Queuing), WRR (Weighted Round Robin), GDR (Deficit Round Robin), WFQ (Weighted Fair Queuing), WF2Q (Worst-Case Fair Weighted Fair Queuing), etc.

Some algorithms provide a static scheduling function, which implies that resources can not be freed if they are not totally used, to be then reallocated to other classes.

Actually, algorithms allowing a dynamic allocation of the resources are privileged in DiffServ networks.

We presented the main functions of a DiffServ router. The IETF recommendations leave many questions about monitoring and scheduling functions without any response. How to configure in an optimal way the policing control algorithm parameters for a given customer using several types of traffic? How to choose the intervention method in case of traffic in excess? As for core routers which will be the configuration allowing the management of EF, AF, CS and BE PHBs? How to define the solution if it is based on a combination of different scheduling mechanisms (WFQ + PQ)? One of the limits of this system is the difficulty of configuring DiffServ routers. Indeed, there are multiple ways of managing the differentiation of services according to the classes.

The monitoring mechanisms (TB and TRTRM) give the required parameters to define the source traffic envelope in a DiffServ domain. However, other indications are needed to ensure the whole network functionalities. These indications should enable edge routers to ensure traffics classification and to specify the desired QoS constraints of each traffic. It is thus, necessary to define a contract between the customer and the service provider. Section 2 presents the required elements for the DiffServ domain configuration. In section 3 we define our Bandwidth Broker model and the allocation strategy. Section 4 presents the implementation of our model using Web-Services and our future work. The conclusion is in section 5.

2. RESOURCES MANAGEMENT AND DIFFSERV CONFIGURATION

2.1 Per-Domain Behavior

The informational RFC 3086[7] defines Per-Domain Behavior (PDB) as the expected edge-to-edge treatment that an aggregate flow will receive within a DS Domain. A PDB sonsists of one or more PHBs and traffic conditioning requirements. Contrary to PHBs, a PDB defines a particular combination of DiffServ components that can be used inside a domain to offer a quantifiable QoS. Five PDBs have been defined.

1 A Best Effort (BE) PDB.
2 The Virtual Wire (VW) PDB.
3 The Assured Rate AR PDB.
4 The one-to-any Assured Rate PDB.
5 A Lower Effort (LE).

For the moment, the definitions of PDBs seem to us insufficient to characterize with precision the edge-to-edge behavior within a DS domain. We need a complete description of the mechanisms to be used within the DS domain that handle the policing, queuing and scheduling functions. The PDB definition is done by the operator of the DS domain. It characterizes the behavior and the operation of his network. That's why it seems to us that it is not necessary to have a normalized PDB.

2.2 SLA - SLS

A Service Level Agreement (SLA) is a contract that specifies forwarding services a customer can expect to receive from a provider. The SLA can cover one or more services a provider can offer to a customer. Each service is technically described in a Service Level Specification (SLS). The SLS contains typically a description of the allowed traffic envelope (peak flow, mean flow, etc.). It's on this basis that a provider can check the traffic conformity and decides which policy to apply for the traffic in excess. The Tequila project gives a more detailed description of the SLS parameters [8].

2.3 Admission Control

The admission control allows the acceptance or the refusal of a new traffic. This could be done by using multi-criteria traffic filters. Only conform traffics with the definite filters are authorized on the network. However, in case of a dynamic service (dynamic SLS) the implementation of an admission control supposes that edge router can access a centralized resources database.

The implementation of an admission control based on per-flow signalization raises significant scalability problem. Thus, a best approach consists in using a minimal description of each flow (SLS) coupled with an admissibility condition. The admissibility criterion determines whether to accept or not a new flow. A criterion can be a threshold of available bandwidth. We made the choice to store in the Bandwidth Broker central database the whole reservations on each link of the DS domain.

A Bandwidth Broker (BB) is a central element in an Autonomous System (AS) that manages network resources within its domain. It also cooperates with other BBs in the neighboring domains to manage the In/out inter-domain communications. The BB gathers and monitors the state of QoS resources within its domain. It uses that information to decide whether to accept or not new traffics. The BB follows the client/server model. It can use COPS protocol to communicate with edge routers within its domain or with adjacent BBs from neighboring domains.

2.4 Policy Based Network

A PDP (Policy Decision Point) is a process that makes decisions based on policy rules and the state of the services those policies manage. The PDP is responsible of the policy interpretation and initiating deployment. In certain cases it transforms and/or passes the policy rules and data into a form of syntax that the PEP (Policy Enforcement Point) can accept. PEP is an agent running on a device (edge router) that enforce a policy decision and/or makes a configuration change. In the DiffServ model, the BB is a PDP.

Signaling protocols are commonly used in policy based network. Policy communication protocols (COPS) enable reading/writing data from a policy repository (SLS database) and communication between PEP and PDP. COPS stands for Common Open Policy Service, is a client/server model that support policy control over QoS signaling protocol[9]. It uses TCP protocol for messages exchange. Many COPS extensions exist like the outsourcing model, the provisioning model etc. In the outsourcing model, PEP can send request, update and delete messages to remote PDP and the PDP returns back its decision to the PEP. The provisioning model (COPS-PR) is used to "push" decisions from PDP to PEP, policy data is described by Policy Information Base (PIB).

According to COPS-PR protocol, the network devices identify their capabilities to PDP using PIB model. PDP can then take into account characteristics of the device, when handling request and/or translating policy rules into PIB parameters. Other extensions like COPS-ODRA for DiffServ, COPS-SLS[10] that supports dynamic SLS were proposed. We thus, believe that COPS is not yet a steady protocol.

3. OUR BANDWIDTH BROKER APPROACH

This section introduces our proposal of an intelligent Bandwidth Broker, its architecture and how it operates. Our goal is to build a tool that handles admission control function and support dynamic SLA. Contrary to other approaches[11] we don't use RAR (Resource Allocation Request) messages because it leads to a two-level resource negotiation. Indeed, the use of RAR can be done only after a previous negotiation of the corresponding SLA.

3.1 DS Domain strategy for resources allocation

The major difficulty that faces a DS domain operator is its capacity to offer the required QoS by supplying in an optimal way the needed resources. Moreover, it is of high importance that a provider can manage easily its domain. We propose a resource allocation strategy of bandwidth based on traffic classes. Each provider has to define the set of traffic classes (DSCP) to use within its domain. For example, he can define four classes: voice, critical, normal and best effort. In this strategy we assume that:

1 All the classes have higher priority compared to the best effort class. In addition, the packets belonging to these priority classes can not be rejected in the core of the domain. This constraint implies that the bandwidth allocated to these classes should be limited and less than the network capacity.

2 The BE traffics are always admitted into the network because they share the remaining bandwidth. The BE class do not offer any guarantees.

3 The remaining bandwidth of a class can be reused by another class.

4 The network operator specifies the DSCP values for each class. In our example we affect EF DSCP to voice class, the AF to critical, the CS to normal and the BE to best effort class.

The provider should specify the maximal allocated bandwidth threshold of each class, allowing the reuse of unused resource by the other classes. We propose to allow resource overbooking of certain classes. This overall strategy for resources allocation is depicted in table 1.

Table 1. Global bandwidth allocation strategy (classes table)

Traffic	Max Allocated BW	Overbooking
Voice (VO)	15%	1.0
Critical (CR)	30%	1.5
Normal (NO)	10%	2.0
Best Effort (BE)	45%	8.0

The provider must limit the EF traffic to guarantee the performance of his DiffServ domain. We thus decided not to authorize EF class overbooking. The other overbooking coefficients are chosen by the operator according to his marketing strategy.

The only constraint imposed by our model is that even with overbooking, the allocated resources of the different priority classes (VO, CR and NO) must remain lower than 100% (15×1 + $30\times$ 1.5 + $10\times$ 2 = 80%). Consequently, the operator guarantees a very low packet loss rate even if all the traffics are active at the same time obviously by degrading the available resources for best effort. However, it is clear that the main priority of an operator is to sell at a higher price the available bandwidth. We presented a simple model for global resource allocation strategy. This approach does not inhibit the operator from adding new classes of traffic.

3.2 Managing resources with Bandwidth Broker

To ensure the admission control function, our Bandwidth Broker requires the reservations statistics within its DS domain. The reservation statistics are grouped by class on each link of the network. The tender of a new SLA implies two important tasks to be done by the Bandwidth Broker. The first one consists of a mapping between the required QoS constraints and a per-domain behavior PDB. This mapping leads to the PHB (or DSCP) assigned to this traffic. At the end of this task the Bandwidth Broker knows the required bandwidth (traffic_throughput in bits/s) and the class of traffic (traffic_class).

The second one consists in determining if the new traffic can be accepted or not in the network. For this reason our BB has the forwarding information of its domain and it knows the whole possible routes between the Ingress and Egress routers. Each route is defined by a set of links ($route_1 = r_{1,1}, r_{1,2} \ldots r_{1,n}$). On each link we have the reservations that were carried out corresponding to the demanded traffic_class. It is then possible to deduce the available bandwidth on this link by the given formula:

$$BW_{avalaible} = BW_{Link} \times (BW_{class\%} \times BW_{overbooking}) - BW_{aLink,class} \quad (1)$$

1 BW_{Link}: BW of the link
2 $BW_{class\%}$: Max allocated BW of this class
3 $BW_{overbooking}$: Overbooking parameter of this class
4 $BW_{aLink,class}$: Sum of the allocated BW on this link for this class.

Example: Supposing that we accepted on a 10Mbit/s link two AF traffics (1.2 Mbits/s and 2.5 Mbits/s). Then the available bandwidth for a new AF traffic is equal to: 10Mbits/s $\times$ (30% $\times$ 1.5) - (1.2 + 2.5) = 0.8Mbits/s.

This computation must be done for each link. We can then deduce the available bandwidth on the route by taking the lowest available bandwidth value on the route's links. If we have more than a possible route, the BB selects the one having the highest available bandwidth. This solution allows a good use of the network resources by selecting the less loaded routes. In case of non point-to-point traffic the BB has to take into account the state of the reservations on the whole set of routes (broadcast). If the allocation of the demanded resources fails on one of these routes, the BB rejects the client's SLA.

3.3 Intelligent Bandwidth Broker Services

Our approach allows the proposition of intelligent services at the same time for customers and the DiffServ operator. Our BB implementation relies on agent technology and Web-Services. An agent is an autonomous program having a goal to attend. In our Bandwidth Broker architecture, agents are Java programs offering services that can be reached using Web-Services. Some of the main functions are the following:

1 A bandwidth allocation request can be rejected because of insufficient resources availability. In this case, an agent can reach the available information in the database and proposes a degraded service by indicating the maximum available bandwidth.
2 Our BB is able to compute the available bandwidth per class of service. If there are no sufficient resource for the required traffic. It can thus, determine if this traffic can be assured by another class. Knowing that clients, do not need to have any information about the operator classes, a software agent can calculate the possible QoS values of this class and propose a new SLA based on new degraded QoS criteria.
3 It is also possible that agents carry out periodic analyses that aim to inform the operator about the overall state of resources within its domain. Thus, a per-class analysis gives the resources allocation rate (i.e. 5% of the overall 30% of AF) and can alert the operator about the less used classes. This alarm means that the operator has to change its business model, either by offering these less used classes at lower prices or simply to stop offering them.

4 After a certain time, the network resources will be consumed. An agent can analyze the levels of available bandwidth on each link, which makes it able to report periodically the load ratio per link (value of BW higher than a threshold). At the issue of this analysis the agent can suggest a network upgrading strategy (to add a new link between two routers).

4. WEB-SERVICES APPROACH

The dynamic SLA management and the interactions between adjacent BB require important exchanges of information. Currently, it seems to us that COPS still unstable and unnecessarily complicates the implementations of Bandwidth Broker. The BB follows the client/server model and that's why we preferred to implement it by using Web-Services.

A Web-Service[12–13] is a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format (specifically WSDL). Other systems interact with the Web Service in a manner prescribed by its description using SOAP messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards. This section presents the Web-Services technology and our actual Bandwidth Broker architecture.

4.1 Our Bandwidth Broker architecture

The required data for the BB are stored in a relational database management system (RDBMS). From the BB specification presented in section 3, we deduce the tables of the MySQL database.

1 SLA table: contains information about active SLA within the DiffServ domain.
2 Classes table: hold the list of classes that characterize the provider resource allocation policy within a given domain.
3 Routers table: is the set of routers within the DiffServ domain.
4 Links table: is the set of links within the DiffServ domain.
5 Reservations table: lists per link and per class reservations within the DiffServ domain.
6 Route table: can be deduced from the routers, links and SLA tables. However for optimization issues, we decided to pre-calculate all possible routes within the DiffServ domain. This is possible from a technical point of view and corresponds to the Cisco approach (there routers can memorize 600 000 possible routes).

The SQL language allows the management of the database tables and data filtering (i.e. SQL queries). We use Java programs (BASIC Java Service Layer) and the JDBC package to access the data and to implement the basic services. These services allow adding a bandwidth reservation on a link, for all the links of a route, to calculate the available resources on a route for a given class, etc. The Intelligent Agents Layer consists of autonomous agents written in Java. Currently, Madkit is our agents' platform. Several multi-agent platforms have been proposed allowing the development of complex system with the help of agents. The main insufficiency of these approaches is the lack of an organizational structure for the agents. Some researchers[14] have proposed a multi-agent platform named MadKit based on three concepts: agent, group and role. The generic development of a multi-agent system and the agents' organization constitute the central proposal of this platform[15]. Two structure levels are proposed: the group and the role. An agent belongs to one or several groups, and inside a group an agent can play one or several roles. A role can be seen as a particular function of an agent. From the agents' cooperation point of view, these organization concepts allow to structure dialogues between agents. An agent can communicate directly with other agent identified by its address or can broadcast the same message to each agent with a given role in a group.

Finally, the last layer (Web-Services Layer) gathers the whole services accessible from outside and allows the communication with the edges routers and other Bandwidth Broker.

4.2 Previous Bandwidth Broker models

There have been numerous undertakings to propose a Bandwidth Broker model for use within a DiffServ environment, the most notable being the following[16]:

1. CANARIE ANA: Implementation of a basic BB that handles differentiated services. This model uses the BBTP (Bandwidth Broker Transfer Protocol) for the Client/BB communications.
2. University of Kansas Research Group: Implementation of a BB that can handle internal and external differentiated services. This model uses the RAR messages and BBTP for message exchange protocol.
3. Merit: Proposition of a multidomain Bandwidth Broker that support the VLL (Virtual Leased Line). This model focuses on the role of authorizing and establishing one type of service (i.e. VLL).
4. Novel: This model separates the QoS control from core routers. It relies on virtual time reference system for QoS abstraction from the data plane.

4.3 Futures works

Our proposal avoids the use of signaling protocol between the BB and the core routers when establishing a new flow. Thus, the core routers do not have the responsibility to store the customers' traffics information and therefore, we respect the DiffServ model philosophy.

At this stage we have considered that the reservations are active starting from the SLA acceptance date and until its expiry date. Thus, our SLA table contains only the active flows. Therefore, it is possible to enhance this model by adding another table to store non-active flows.

Also it is necessary to consider scheduled SLA with several active/passive phases. Thus, the reservation requests have a start/end time. Consequently, it complicates the computation of the available resources and it is necessary to build the exact reservations state within the time interval of the demanded service. It is thus, necessary to identify all resources requests and releases intervals of this new traffic, and to re-evaluate the whole allocation status per interval. We are evaluating this approach.

For the inter-domain traffics, the BB must preliminary contact the other BB in the adjacent domain to propose an end to end QoS service.

5. CONCLUSION

In this article, we presented the needed functionalities for the operation of a DiffServ network. Additional research on Per-Domain Behavior and signaling protocols seems important to specify the overall behaviors of the devices in an IP network offering QoS. The role of centralized equipment was emphasized in order to provide the admission control function. Our approach allows the exchange of information between the edge-routers and the Bandwidth Broker or between the Bandwidth Broker of different adjacent domains without using signalization between the BB and the core-routers.

The management of an IP network that support QoS and integrate heterogeneous approaches and protocols (IntServ, DiffServ, MPLS, RSVP, COPS, etc) is complex. Many researches have proposed the use of a traditional IP network on the management plan in order to ensure the administrative functionalities. From this point of view, we think that the use of signaling protocols like COPS or RSVP does not benefit from the progress made in the distributed applications domain. We thus built our Bandwidth Broker architecture by using Web-Services concepts.

REFERENCES

1. S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss. An architecture for differentiated services. *IETF, RFC 2475*, December 1998.

2. K. Nichols, S. Blake, F. Baker, and D. Black. Definition of the differentiated services field (ds field) in the ipv4 and ipv6 headers. *IETF, RFC 2474*, December 1999.

3. V. Jacobson, K. Nichols, and K. Poduri. An expedited forwarding phb. *IETF, RFC 2598*, June 1999.

4. J. Heinanen, F. Baker, W. Weiss, and J. Wroclawski. Assured forwarding phb group. *IETF, RFC 2597*, June 1999.

5. S. Shenker and J. Wroclawski. General characterization parameters for integrated service network elements. *IETF, RCF 2215*, September 1997.

6. J. Heinanen and R. Guerin. A two rate three color marker. *IETF, RFC 2698*, September 1999.

7. D. Goderis, Y. T'Joens, C. Jacquenet, G. Memenios, G. Pavlou, R. Egan, D. Griffin, P. Georgatsos, L. Georgiadis, and P.V. Heuven. Service level specification semantics and parameters. *draft-tequila-sls-01.txt*, June 2001. Work in progress.

8. K. Nichols and B. Carpenter. Definition of differentiated services per domain behaviors and rules for their specification. *IETF, RFC 3086*, April 2001.

9. D. Durham, J. Boyle, R. Cohen, S. Herzog, R. Rajan, and A. Sastry. The cops (common open policy service) protocol. *IETF, RFC 2748*, January 2000.

10. T.M.T. Nguyen, N. Boukhatem, Y.G. Doudane, and G.Pujolle. Cops-sls: A service level negotiation protocol for internet. *IEEE Communications Magazine*, 40(5):158–165, May 2002.

11. P Chimento and al. Qbone signaling design team. *Final Report*, July 2002. http:qos.internet2.eduwgdocuments-informational20020709-chimento-etal-qbone-signaling.

12. J. Ferber. Multiagent systems for telecommunications: from objects to societies of agents. *networking 2000*, 2000. Paris.

13. Madkit. official web site. *last visited*, June 2004. http:www.madkit.org.

14. S. Sohail and S. Jha. The survey of bandwidth broker. *Technical report UNSW CSE TR 0206*, May 2002. School of Computer Science and Engineering, University of New South Wales.

15. J. McGovern, S. Tyagi, M. Stevens, and S. Mathew. Java web-services architecture. *Morgan Kaufmann*, May 2003.

16. W3C. Web-services architecture. *W3C website*, February 2004. http:www.w3.orgTRws-arch.