

Towards Better Support of Transaction Oriented Communication in Differentiated Services Networks

Roland Bless, Dirk Holzhausen, Klaus Wehrle

Institute of Telematics, Universität Karlsruhe (TH),
Zirkel 2, D-76128 Karlsruhe, Germany
Phone: +49 721 608 6411, Fax: +49 721 388097
{bless,holzhaus,wehrle}@tm.uka.de

Abstract. Applications using transaction oriented communication have special requirements. Packet forwarding of a client's request and a server's response should be as fast as possible and also highly reliable. This implies employment of a communication service supplying minimal end-to-end delay and minimal packet loss. Currently defined per-hop forwarding behaviors (PHBs) and per-domain behaviors (PDBs) within Differentiated Services networks do not support these requirements of bursty traffic well enough. This paper proposes a new PHB and a suitable PDB called "*Quick Forwarding*" to fulfill the previously described requirements. Results of simulations are presented for evaluation of the newly proposed behavior and show that it performs better than other behaviors based on Expedited Forwarding or Assured Forwarding. This is a first step towards better support for this important class of applications.

1 Introduction

There are many applications that would profit from using services that are based on Differentiated Services (DS) [3,1] mechanisms. For instance, Internet telephony applications, Virtual Private Networks or applications that transmit continuous streams requiring a low and bounded jitter will preferably use Per-Domain Behaviors (PDBs, cf. [8]), such as the "Virtual Wire" PDB [7], based on the Expedited Forwarding (EF) Per-Hop Behavior (PHB) [6]. More elastic applications may use PDBs (e. g., "Assured Rate" [9]) based on the proposed Assured Forwarding (AF) PHB classes [2]. While they get an assured minimal rate, additional resources may be used if available. The latter are usually shared (at least) between all users of the same AF PHB class, therefore allowing some degree of statistical multiplexing. But packets using these additional resources (i. e., they exceed a committed burst size [9]) are more likely to be discarded during congestion situations.

However, an important class of applications is currently not supported "well enough" by the so far proposed PDBs and PHBs. Such applications show a

certain typical property: a client sends a query message consisting of one or few more packets to a server which sends a (normally larger) reply message back to the client. Therefore, the resulting traffic shows a bursty behavior: a small series of packets is consecutively sent at peak rate (usually at link rate) to the peer (client or server respectively) after which usually a “silence period” follows during which no packet transmission occurs. In this paper, a message exchange with such properties is called *transaction oriented communication*, because of the query/reply (and possibly commit) exchange which is typical for database or directory queries and operations as well as for many signaling protocols, remote procedure calls (RPCs) or middleware infrastructures such as CORBA.

The requirements of this application class related to network communication services are mainly the following two: messages (that may consist of one or more packets) should be transmitted as fast as possible (all packets should possess a minimal end-to-end delay) and highly reliable. The first property is desirable since it shortens the overall transaction time which is very important for fast operations (e. g., RPCs). The second property serves the same purpose: if a packet gets lost it has to be retransmitted which adds additional delay. Currently proposed PDBs and PHBs do not support this application class “well enough” in the sense that transactions will last longer than necessary due to violation of one or both properties above. This is due to the fact that packets experience additional delay or have a higher drop probability than required for fast transactions.

Basically, Differentiated Services networks can constitute a good basis for enabling support for those applications. But existing per-hop behaviors, which principally determine how packets are treated while they are forwarded by a DS node, do not fulfill both properties very well.

The EF PHB [5,6] is a forwarding behavior for enabling “low delay, low jitter and low loss services”. In order to achieve these properties EF queues should ideally contain no packets at most times (especially to accomplish low and bounded jitter). Thus, an incoming EF packet should leave a DS node in principle directly after its arrival, i. e., it sees no packets in the respective EF queue. Empty queues can be reached by ensuring that “the service rate of EF packets on a given output interface exceeds their arrival rate at that interface over long and short time intervals, independent of the load of other (non-EF) traffic” [5]. In order to restrict jitter, bursts of EF packets arriving at a DS node are not desirable for this PHB. Eliminating or reducing packet bursts usually requires traffic shaping at least at DS domain ingresses [7]. This however imposes an additional delay for a burst of packets that form a transactional query or reply. For fast transaction processing it is desirable that all packets of such a transactional message are forwarded and arrive *en bloc*, i. e., as a consecutive sequence. Consequently, the bursty nature of transaction oriented communication contradicts with the objective of having nearly empty EF queues.

In contrast to EF, the proposed AF PHB [2] allows bursts, but normally assigns a higher drop precedence to packets in bursts (it actually depends on the concrete PDB, but it will usually be the case when the committed information rate and burst size is exceeded [9]). An AF PHB class comprises a group of

three AF PHBs that have at least two different drop precedences. The objective of assured forwarding is that transfers get at least their committed information rate, but when there are additional resources available, they can be used for faster transmission. Packets using those additional resources are marked with an AF PHB of higher drop precedence (e. g., allowing TCP to reduce its sending rate and to adapt to the available shared bandwidth). This would increase the drop probability of a burst’s packets thereby violating the second property (“nearly no loss”) from above.

In this paper a new behavior is described that yields a better support for the previously described transaction oriented communication. Especially, time critical applications such as banking and brokerage applications or real-time process control applications will profit from this better support.

The paper is organized as follows: the properties of a new forwarding behavior are defined in the following section 2. Section 3 describes an evaluation by simulation of the proposed behavior. The paper closes with concluding remarks and an outlook on future work in section 4.

2 A Quick Forwarding Behavior

We define a *Quick Forwarding* behavior that consists of a PHB and an appropriate PDB, i. e., (among other things) appropriate conditioning functions and assured quality of service attributes. Since the quick forwarding behavior should be “burst friendly” and should also provide a fast forwarding of packets as well as a low loss guarantee, the following conditions apply:

- The QF PHB uses own resources, i. e., a separate queue. Packets of higher priority, e. g., EF packets should always be served before any QF packets, but any unused EF bandwidth will be used by QF.
- It assures transport of packets that are sent with a mean rate, actually defined by the mean rate r and a maximum tolerable burst size b that is sent at peak rate p (commonly link rate). Therefore, after transmission of a burst, the sender must pause sending packets for a while.
- A DS node should forward all packets of a single burst as fast as possible, i. e., with maximum link output rate. Usually bursts will cumulate to even larger bursts in (output) queues. Thus, in order to empty the queue fast enough, the ratio of the quick forwarding behavior aggregate bandwidth to overall link bandwidth should be small.
- The queue should not grow over longer periods, since this would inevitably lead to packet loss. Thus, the queue should be emptied fast enough. Hence, the effective departure rate of a QF aggregate must be greater than its maximum arrival rate, i. e., the total rate of all incoming QF aggregates flowing into that aggregate. This condition has to be verified by admission control with respect to the given mean rates.

The suitable traffic conditioning actions [3] are as follows:

- Classifying and initial marking of traffic at the first boundary node (first-hop router) of a DS domain.
- Metering in the first-hop router by using a token bucket that has fill rate r and size $B := b(1 - r/p)$.
- Unconditional dropping of non-conforming packets (i.e., when the token bucket contains no or too few tokens).
- No traffic shaping at the domain ingress, since it would add additional delay to packets of a burst.

In addition to that, some resource management aspects have to be considered. *Admission control* has to be performed (e.g., by some DS management components such as Bandwidth Brokers) in order to guarantee transmission of data at the specified mean rate and to prevent packet loss. Both mean rate and maximum burst size have to be considered in admission control tests in order to avoid buffer overflows. Due to the fact that all injected bursts may be cumulated in the worst case, queues with large capacities have to be provided. Since the overall amount of traffic in the QF aggregate is limited by admission control and link rates should be much higher than the configured QF rate, those queues are also emptied quickly.

At a first glance it may be non-intuitive why QF could forward packets quicker than EF, because EF packets should very rarely see any packets before them in their output queue. But one important property of EF is to restrict jitter, thus bursts or filled queues are not desirable by definition [5,6]. QF avoids any traffic shaping and always serves output queues at link rate, thereby allowing a faster forwarding of packets as illustrated and confirmed in the following section.

3 Evaluation

To show that using QF is indeed faster than using EF for the considered transaction oriented communication resp. bursty traffic, several simulations were performed. A modular QoS simulation suite called *simulatedKIDS* [13] was used for this purpose. It is a very flexible construction set of basic building blocks for QoS mechanisms that can be combined to build nearly arbitrary QoS behavior. *SimulatedKIDS* is based on the freely available discrete event simulation system *OMNeT++* [11]. Various scenarios with different traffic generators and topologies have been examined, but due to space limitations, only two of these scenarios are described here.

The implementation of QF was evaluated using two different variants of scheduling algorithms: one with simple priority scheduling (cf. Fig. 1) and one with a modified weighted fair queueing (WFQ) combined with a simple priority scheduling (to let EF always have priority). The latter implementation could only show significant better results than EF if there was unused bandwidth of any other PHBs (including EF) left. This is due to the fact that WFQ limits the bandwidth that QF can use, thus bursts may be delayed until QF gets its time slice again to serve queued packets. For the rest of the paper only the implementation based on simple priority scheduling is used.

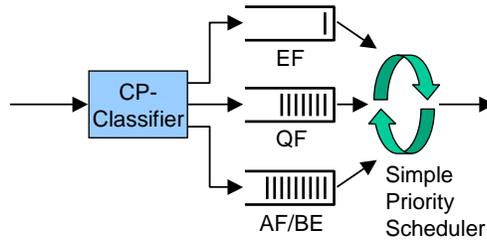


Fig. 1. DS mechanisms on an output interface of an interior router

3.1 Scenario 1

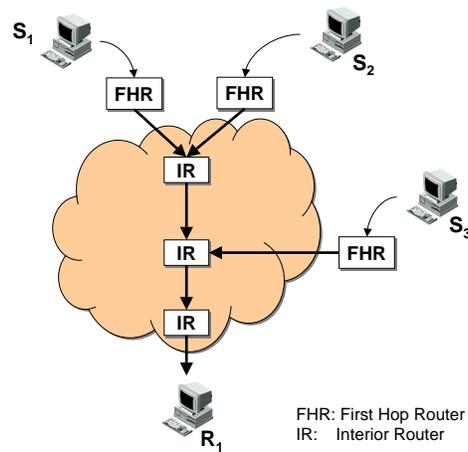


Fig. 2. A small topology

The small DS domain that is depicted in Fig. 2 was used for validation of the implementation in a first simple scenario. Due to its low loss property and expedited packet forwarding at highest priority, setting up a “Virtual Wire” path [7] for transmitting important transaction oriented data would have been an alternative to using QF. Thus, a first comparison between EF and QF was done. Bursty traffic was generated by simple On/Off sources at each of the senders S_1 , S_2 and S_3 and was sent towards the receiver R_1 . The traffic mean rate was set to 10 Mbit/s and the burst length to 40 kbyte. All packets had a fixed length of 1250 byte. The first-hop routers (FHR) had a link rate of 100 Mbit/s and interior routers (IR) a link rate of 150 Mbit/s. All queues were large enough so that no packet losses occurred due to completely filled queues. The configured share of EF and QF were each 20 Mbit/s at FHRs and 60 Mbit/s at IRs, thus the

bandwidth shares of EF and QF were identical. Links were additionally filled up (saturated) by best-effort traffic. Loss of QF or EF packets did not occur during the simulation.

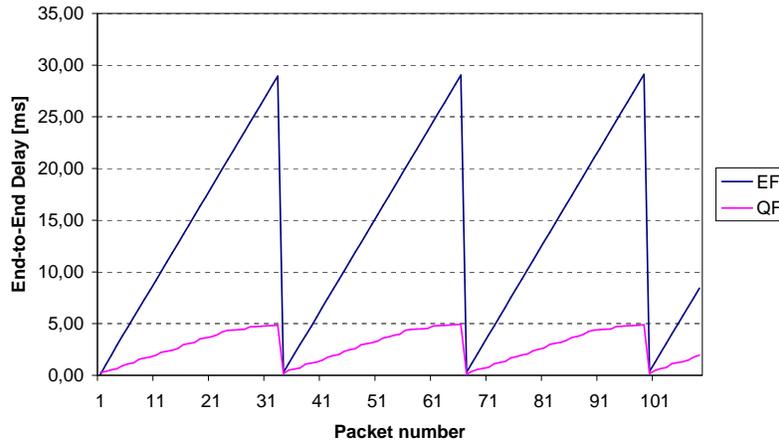


Fig. 3. End-to-End delay of packets received at R_1 (minimum: 0.067 ms)

The advantage of using QF instead of EF for such bursty traffic is illustrated in Fig. 3. Traffic generators are sending a burst of packets at link rate (“On” period) and then pause for a while to comply with the configured mean rate (“Off” period). This results in the depicted sawtooth curves, because a packet within a burst has to wait in the queue until all preceding packets of the burst have been transmitted. Traffic shapers release EF packets at the configured rate after the first router, resulting in a continuous data flow without bursts. But due to the artificial delay inserted by the traffic shaper, a stronger increasing end-to-end delay for adjacent EF packets during a burst can be observed as expected.

3.2 Scenario 2

The second scenario presented here is more complex and is illustrated in Fig. 4. In addition to the previous scenario, an AF class (PHB group) with two different drop precedence levels is also simulated. The default PHB for best-effort traffic is mapped to the AF PHB with highest drop precedence AF13, while packets with medium and low drop precedence are treated identically. Therefore, AF and BE share the same queue as depicted in Fig. 1. This configuration lets AF excess traffic share the residual bandwidth with BE traffic (cf. first example in (b) of section 7 in [2]).

There are 9 access IP subnets that are connected via first-hop routers to the DS domain. The FHRs have a link bandwidth of 100 Mbit/s while interior routers

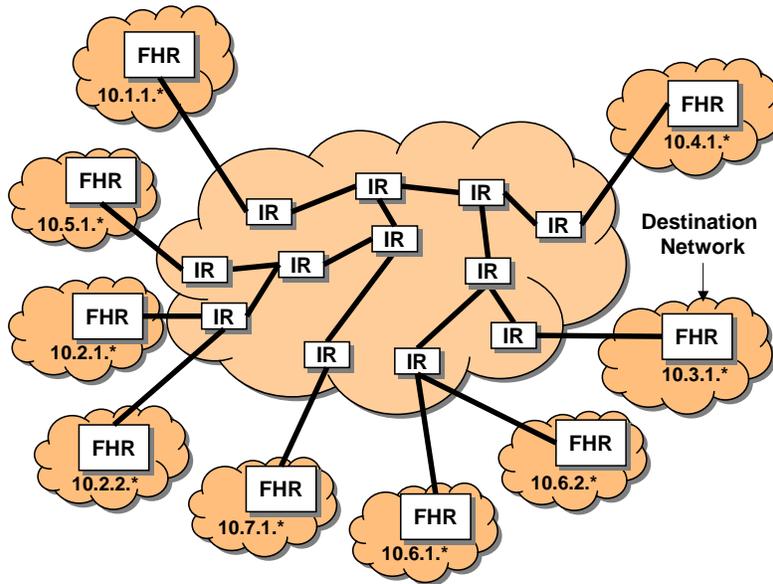


Fig. 4. A large DS domain, used for evaluation in scenario 2

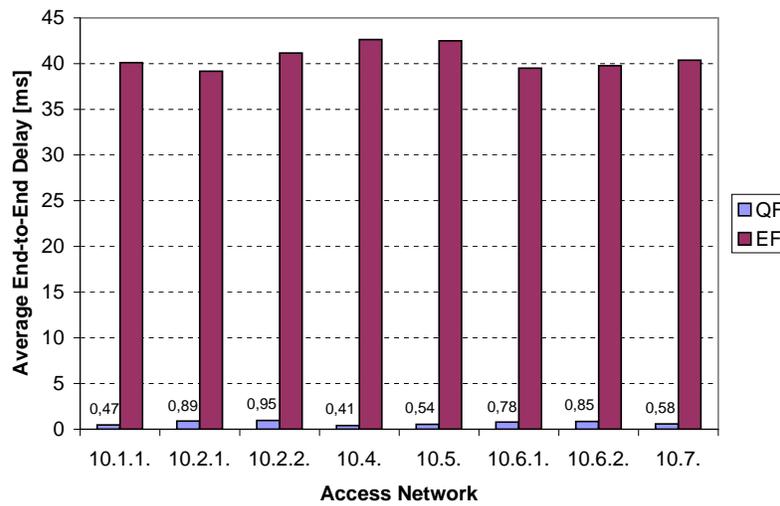


Fig. 5. Average end-to-end delay (in ms) of QF and EF packets for every access network to destination network 10.3.1.0/24

have links up to 300 Mbit/s. For every EF, QF and AF PHB, self-similar and bursty traffic is generated by 4 independent Pareto distributed On/Off-sources

(with cut-off). The burst size varied between 30 and 75 kbyte while the off-time varied between 0.25 s and 5 s. Thus, taking all 4 traffic generators together, the mean rate for every PHB was roughly about 10 Mbit/s. Best-effort traffic was sent at link rate so that it used up any residual bandwidth. All traffic was sent to the access network 10.3.1.0/24. Several simulation runs were executed with a duration of 500 s simulated time each.

Again, loss of QF or EF packets did not occur during the simulation. The maximum required buffer size for the QF queue in the last interior router before the destination network was 2.4 Mbyte to prevent packet loss. Figure 5 shows a comparison of average end-to-end delays between QF and EF for each subnet. In this scenario, QF packets are transmitted at least one order of magnitude faster than EF packets. This is mainly due to the traffic shaping of EF packets to an aggregate rate of 10 Mbit/s at all domain ingress points (FHRs), whereas QF packets can be forwarded with 100 Mbit/s.

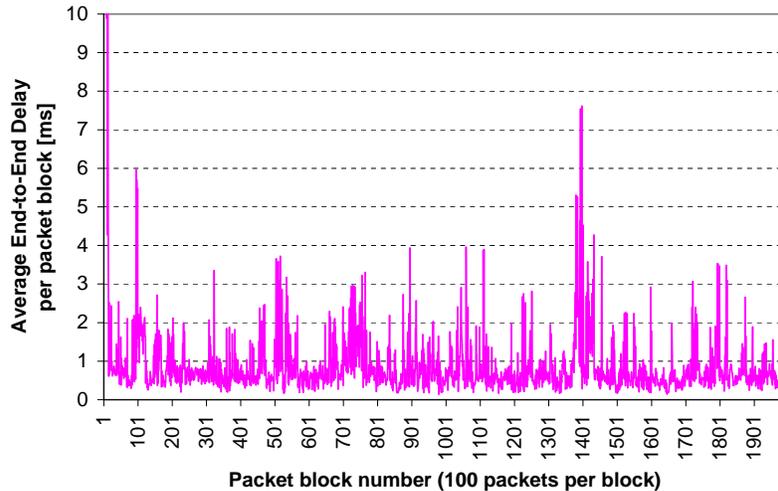


Fig. 6. End-to-end delay (in ms) averaged over a packet block of 100 QF packets from network 10.2.2.0/24

A more detailed view of the achieved end-to-end delay for QF packets is presented in Figure 6 where end-to-end delay of packets from network 10.2.2.0/24 is averaged over an interval of 100 packets (packet block, numbered on x-axis). Although there are several peaks present due to aggregation effects, the overall average end-to-end delay value lies around 1 ms.

The difference in order of several magnitudes between end-to-end delay values of QF, EF and also AF packets can easily be seen in the Figure 7 that uses a logarithmic scale on the y-axis. Delay of AF packets is even worse, since they share one queue with best-effort packets in this particular implementation.

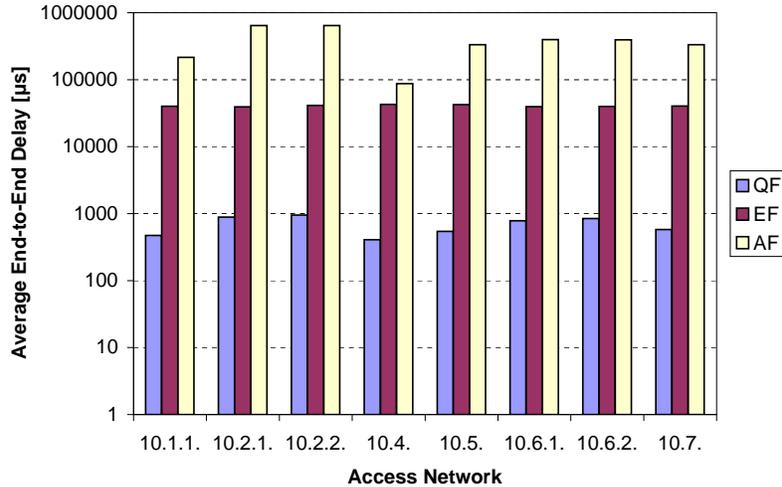
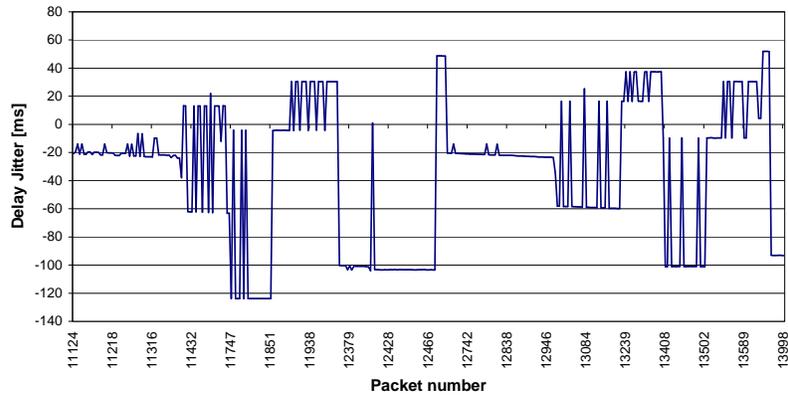


Fig. 7. Average end-to-end delay (in μs) of QF, EF and AF packets for every access network (logarithmic scale on y-axis)

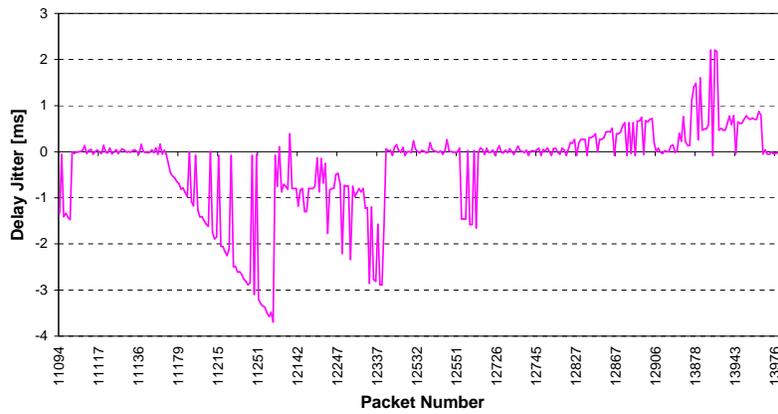
The current delay jitter of packets (calculated by subtracting end-to-end delay values of consecutive packets) was also examined. Additionally, the delay jitter behavior of QF is not worse than EF, it is even better as Figures 8(a) and 8(b) show. Please note that the scale in Fig. 8(a) ranges from -140 ms to 80 ms whereas only from -4 ms to 3 ms in Fig. 8(b). This is due to the fact that bursts of EF packets are stretched by traffic shapers at the domain ingress, thus enlarging the individual delay jitter of packets.

4 Conclusion and Outlook

The presented simulations have shown that the defined Quick Forwarding behavior is able to support transaction oriented communication better than other currently existing behaviors. Particularly, the EF PHB was chosen for comparison, because up to now it would have been an alternative to set up “virtual leased lines” [7] for transmission of transaction oriented traffic to use the low end-to-end delay and low loss properties of EF. Now, by using Quick Forwarding a better possibility exists. Many client/server applications or every kind of control and management traffic will profit from a service based on Quick Forwarding. Even when TCP is used as a reliable transport protocol, bursts of several segments (up to current maximum window size) will typically be generated if the window is completely opened once. Because QF offers also a rate guarantee, TCP connections that use QF will not experience congestion. Consequently, there is always an open window comprising several segments. However, for transaction



(a) EF packets



(b) QF packets

Fig. 8. Delay Jitter in *ms* of some packets from network 10.2.2.0/24

oriented applications a better suited protocol such as SCTP [10] (it preserves message boundaries) should be used if available.

On the one hand, there are large buffers needed in order to prevent packet loss. On the other hand, providing large buffers would be better than having to retransmit data resulting in a much larger end-to-end delay. Large buffers that potentially hold many packets are not a problem for QF whereas it is for EF due to the requirement of almost empty queues. Since a QF queue is always served at link speed, high bandwidth links will empty those buffers faster. The actual gain of QF is caused by avoiding traffic shaping (and using WFQ for QF which

has a similar limiting effect) at domain ingress. Thus, QF packets are always forwarded with the highest possible rate. However, damage must be limited that QF traffic could inflict on other traffic, because forwarding of other lower priority classes' packets is suspended until the QF queue is emptied. Therefore, the ratio of QF aggregate bandwidth to overall bandwidth should be not too large. The latter condition must be enforced by using admission and usage control, thus effectively controlling the overall amount of QF traffic (indirectly limiting QF flows and users, too). In our simulations the QF share was set to 20% of the overall bandwidth. Note that unused QF bandwidth can be used by other PHBs of lower priority.

However, it may be difficult to exactly characterize and predict application behavior in advance. Specifying a mean rate may not be flexible enough for all applications. Nevertheless, it allows some kind of admission control which is essential for providing the desired guarantees.

Our simulations prove that providing a QF within a DS domain (i. e., a reasonable PDB) is feasible, but more experience and analysis is needed for providing an end-to-end service. This is, however, also true for other currently defined PDBs. Large buffer sizes for QF queues in order to prevent packet losses may be required, but can be effectively controlled by the admission control procedure. To reduce the required sizes it may be possible to use some degree of statistical multiplexing thereby reducing the loss guarantee to quantifiable percentiles. In the worst case, traffic shaping for QF aggregates will be occasionally required for long end-to-end paths. Therefore, we will investigate possibilities for a new kind of (to some degree) "burst-friendly" re-shaping mechanisms.

Future work will include extended simulations of several DS domains as well as an evaluation by a real implementation using an updated version of our Linux-based DS implementation architecture KIDS [12,4]. Determining a sensible and suitable QF share is of particular interest during our further research.

References

1. F. Baker, D. Black, S. Blake, and K. Nichols. Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers. RFC 2474, Dec. 1998.
2. F. Baker, J. Heinanen, W. Weiss, and J. Wroclawski. Assured Forwarding PHB Group. RFC2597, June 1999.
3. S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss. An Architecture for Differentiated Services. RFC 2475, Dec. 1998.
4. R. Bless and K. Wehrle. Evaluation of Differentiated Services using an Implementation under Linux. In *Proceedings of the 7th IFIP Workshop on Quality of Service, London, June 1999*. IEEE, 1999.
5. B. Davie, A. Charny, F. Baker, J. Bennet, J.-Y. Le Boudec, A. Chiu, W. Courtney, S. Davari, V. Firoiu, C. Kalmanek, K. Ramakrishnam, and D. Stiliadis. An Expedited Forwarding PHB. Internet draft – draft-ietf-diffserv-rfc2598bis-01.txt, Apr. 2001.
6. V. Jacobson, K. Nichols, and K. Poduri. An Expedited Forwarding PHB. RFC 2598, June 1999.

7. V. Jacobson, K. Nichols, and K. Poduri. The ‘Virtual Wire’ Per-Domain Behavior. Internet draft – draft-ietf-diffserv-pdb-vw-00, July 2000. Work in progress.
8. K. Nichols and B. Carpenter. Definition of Differentiated Services Per Domain Behaviors and Rules for their Specification. RFC 3086, Apr. 2001.
9. N. Seddigh, B. Nandy, and J. Heinanen. An Assured Rate Per-Domain Behaviour for Differentiated Services. Internet draft – draft-ietf-diffserv-pdb-ar-00, Feb. 2001. Work in progress.
10. R. R. Stewart, Q. Xie, K. Morneault, C. Sharp, H. J. Schwarzbauer, T. Taylor, I. Rytina, M. Kalla, L. Zhang, and V. Paxson. Stream control transmission protocol. RFC 2960, Oct. 2000.
11. A. Varga. Omnet++ discrete event simulation system 2.0. <http://www.hit.bme.hu/phd/vargaa/omnetpp.htm>, Feb. 2000.
12. K. Wehrle. An Open Architecture for Evaluating Arbitrary Quality of Service Mechanisms in Software Routers. In P. Lorenz, editor, *Networking - ICN 2001*, Lecture Notes of Computer Science 2094, pages 117–126. Springer, 2001. First International Conference on Networking (Proceedings Part II), Colmar, France, July 9-13, 2001.
13. K. Wehrle, V. Kahmann, and J. Reber. A simulation suite for the Internet Protocol with the ability to integrate arbitrary QoS behavior. Proceedings of the Computer Networks and Distributed Systems Modelling and Simulation Conference (CNDS’01), Phoenix, Jan. 2001.