Quality of Service Assurance for the Next Generation Internet

Dimitrios P. Pezaros and David Hutchison

Computing Department
Faculty of Applied Sciences
Lancaster University
Lancaster, UK
LA1 4YR
E-mail: {dp, dh}@comp.lancs.ac.uk

Abstract. The provisioning for multimedia applications has been of increasing interest among researchers and Internet Service Providers. Through the migration from resource-based to service-driven networks, it has become evident that the Internet model should be enhanced to provide support for a variety of differentiated services that match applications and customer requirements, and not stay limited under the flat best-effort service that is currently provided. In this paper, we describe and critically appraise the major achievements of the efforts to introduce Quality of Service (QoS) assurance and provisioning within the Internet model. We then propose a research path for the creation of a network services management architecture, through which we can move towards a QoS-enabled network environment, offering support for a variety of different services, based on traffic characteristics and user expectations.

1. Introduction

Quality of Service (QoS) is a generic term, and people often use it to express a variety of different things, depending on their discipline, as well as on their particular subject area. Depending on the context in which the term is being used, it can have slightly different meanings. Within telecommunications research, and more specifically in the broader Internet arena, QoS can be interpreted as a general effort to apply the necessary mechanisms and techniques, in order to enable different behavior from the network infrastructure among the different types of traffic that are being transmitted over it. More specifically, a common thread running through most definitions of QoS is the ability to differentiate between traffic or service types, so that the network can treat one or more classes of traffic differently than other types [Huston00].

Traditionally, the Internet was designed having as its main objective to perform fast and efficient datagram forwarding. The major design requirements were for the network to be scalable (i.e. support increasing numbers of users) and decentralized enough, so that it does not have a single point of failure. The establishment of these goals was of course influenced by the types of traffic that the Internet was originally designed to carry. These included electronic mail and file transfer, and later on Web traffic. Algorithms and techniques were then applied on the higher layers to support the requirements of these types of traffic, which basically was the reliable delivery of data. The outcome was that the Internet was built on the concept of a dumb network with smart edges at the sender and the receiver ends, which has, and is still, seamlessly operating, satisfying its original requirements.

During the early 1990s multimedia peripherals started being introduced in computer systems, and this was the beginning of a new class of applications with its own requirements and characteristics. It is often referred to as “the new environment”. By definition, multimedia is information represented through digital audio and/or video, in addition to text, image and graphics. The continuously increasing capabilities of workstations and personal computers, as well as the advances in network linkage technologies (able to support transfer rates on the magnitude of Mega-bits per second), led to a situation where multimedia content started being transmitted through the networks (networked multimedia – mid 1990s). In addition, the tremendous success of the web due to its ability to delegate tasks that it was not originally designed for, resulted in an integrated environment having different sorts of traffic being carried over computer networks. The major concern raised by this new environment was how to make the Internet able of satisfying the different requirements of all the types of traffic, without having to change its original design or infrastructure.

In the remainder of this paper, we will critically present the recent deployments towards QoS, as well as some new ideas and aspects of future work, which we believe they will exploit the newest technologies that will lead the next generation Internet. Section 2 will briefly discuss the requirements of multimedia traffic and the need for QoS assurance in the Internet. In section 3 we present the major architectures and protocols that could potentially lead to QoS-enabled networks, highlighting their strengths and weaknesses. In section 4 we will discuss the linkage between Quality of Service and Network Management, and we will raise our research directions in this area. Finally, section 5 provides a number of concluding remarks and future goals.

2. The Need for Quality of Service

The Internet operates on a best-effort basis. That is, there are no guarantees provided by the network upon the delivery of content, from the source to the destination. The basic requirement for traditional data transferred over computer networks was reliable delivery. End-to-end services were therefore built on top of the network layer, mostly by transport protocols (e.g. the Transmission Control Protocol (TCP)), using mechanisms such as timestamps and checksums, in order to ensure that all bits of information were successfully delivered to the final destination. However, the presence of multimedia traffic introduced a new basic requirement, i.e. the timeliness of the arrival of information. Datagrams arriving out of time cause as much disruption as datagrams not arriving at all. The new classes of application that emerged, such as multimedia streaming, Video on Demand (VoD), interactive voice, remote conferencing, etc., raised the requirements of minimizing delay upon content delivery and the variations in this delay. Moreover, some of these applications would not be worth running if consistent throughput capacity could not be guaranteed over their entire lifetime. In addition, the trend of moving towards a single telecommunications network and the candidacy of the Internet to provide such a service, makes the support of these applications essential, rather than a luxury. For example, if the Public Switched Telephone Network (PSTN) is to be substituted by some form of Voice over IP (VoIP), then being able to establish a call and not being disrupted/disconnected before hanging up, will be critical.

It became evident that there was a need for breaking Internet traffic down to several different classes according to content's characteristics, and henceforth to provide different services, trying to satisfy the discrete requirements of each class. In order to maintain the positive characteristics of the Internet while not changing the fundamental principles of its operation, researchers have made efforts to build mechanisms that would enable the co-existence of several classes of service on top of the same physical infrastructure, including the best-effort class, in a way that the network's behavior would become customizable, offering different service guarantees to each class of service. As an outcome of these efforts some architectures were defined, as well as some protocol deployments, which we will outline in the next section.

From a different viewpoint, migrating from today's best-effort networks to QoS-enabled ones would introduce overhead in the Internet's operation. The vast majority of Internet traffic still is TCP-based, having the traditional requirements of reliable delivery. Such a migration would require more complex mechanisms to operate in the network devices, violating the principle of fast and efficient packet forwarding. In addition, it is claimed that due to the increases in bandwidth capabilities, networks will soon be capable of accommodating vast amounts of any sort of traffic, simply by being over-engineered, without needing any QoS mechanisms. Moreover, there is a belief that users don't need these sorts of services, since they experience an acceptable performance from the network at reasonable flat prices, whereas a QoS-enabled network operator would charge more for services experiencing preferential treatment from the network.

The arguments against these statements are that the user requirements increase as fast as the technologies improve and, as already stated, in a future environment the different classes of service will be essential. Network over-engineering could be a solution for the core, but we need to have in mind that congestion appears at the access points, as well as at some links-bottlenecks. Delays cannot be minimized, simply by adding more bandwidth. In addition, mobile computing is a major issue, since mobile devices have neither the capabilities nor the power to take advantage of the high transfer rates. Of course advanced mechanisms for accounting and charging, as well as for security and authentication need to be taken into consideration since premium services can (and must) be offered to people that need them, and are therefore willing to pay for them. It is our belief that Quality of Service assurance has a major role to play in telecommunications generally and the next generation Internet in particular.

3. QoS Efforts

In this section we briefly describe the major architectures and protocol deployments in the QoS arena, highlighting the positive aspects as well as the limitations of each of them.

3.1 The Integrated Services (IntServ) Architecture and the Resource ReSerVation Protocol

In 1994 IntServ (RFC1633) was designed to provide a set of extensions to the Internet best-effort traffic delivery model. The framework was set up to provide a mechanism to allow applications to choose between multiple levels of delivery services for their traffic [Huston00]. The belief of the IntServ designers was that before real-time applications could be broadly used, the Internet infrastructure should be modified to support real-time QoS, which would provide some control over end-to-end packet delays [Braden94]. In IntServ three classes of service were proposed, based on applications' delay requirements. The guaranteed service class providing for delay-bounded service agreements, the controlled-load service class providing for a form of statistical delay service agreements which would not be violated more often than in an unloaded network, and the existing best-effort service class [Mathy00]. Both the guaranteed and the controlled-load service classes are based on quantitative service requirements, and require signaling and admission control in network nodes. The signaling protocol adopted by the architecture was the Resource ReSerVation Protocol (RFC2205). RSVP is based on the concept of a session and the reservation style is receiver-oriented. The receiver sends messages towards the source, requesting resources to be reserved in each node along the path. If these resources are available, then a session can be established. Reserved resources have a timer associated, which is refreshed by messages sent towards the sources throughout the session. The general model can be viewed as being host-centric, since the end-host is responsible for directing the network to allocate adequate resources to the end-to-end traffic flow. This model can be compared to a telephone call in the sense that if the network and the remote party both accept the call, the network is expected to support the call's resource requirements indefinitely. The major advantage of IntServ is that the service classes it provides match the different applications' delay requirements. Furthermore, it leaves the best-effort service class mostly unchanged and therefore does not require any change to existing applications, unless one wants them QoS-enabled. The architecture attempts to offer QoS guarantees per flow or per flow aggregate, so it retains the possibility of operating on an end-to-end basis. It also allows for incremental deployments, since it maintains the forwarding mechanism of the network. So, IntServ capable devices can co-exist with traditional best-effort devices, which will simply make packet forwarding ignoring the concepts of reservations and service classes. This way, the end-to-end best-effort service is improved. On the other hand end-to-end guarantees can only be provided if all the nodes along the path support the IntServ architecture. Scalability concerns are also raised, which make it doubtful whether IntServ can be applied in the Internet, due to the overhead of the per-flow guarantees and the overhead of signaling. In addition, RSVP's receiver-based approach may lead to static reservations that do not suit all the types of applications. Finally, the end-systems involvement in the reservation establishment raises some security concerns and makes the existence of policies that ensure traffic profiles are honored, necessary. The problems of scalability in addition to the need of a more coarse grained QoS support for the core of the network, led the research efforts towards the definition of the Differentiated Services (DiffServ) architecture.

3.2 The Differentiated Services (DiffServ) Architecture

By the time network service providers realized that large-scale deployment of RSVP was impractical and that this sort of solution would always suffer from scalability problems, the networking community began to focus on an alternative model that would provide simple differentiation of traffic while not requiring per flow resource reservation state (which consumes critical network processing resources). The basic idea was to classify the applications' data flows into a few general categories, and break the Internet down to a number administrative domains (clouds). Then DiffServ (RFC2475) could provide a simple and scalable service differentiation by discriminating data flows according to their traffic class, and by providing a model for intra-domain and inter-domain network resource management. Within a domain the service provider is responsible for the provisioning of resources, whereas at domain boundaries traffic entering a network is treated according to agreed local Service Level Agreements (SLAs). End-to-end services can be built by concatenating such local agreements at each domain boundary along the route to the final destination. Each packet can carry in its header the identification of its traffic class, and therefore the network can easily distinguish between traffic classes and treat packets accordingly. The major benefit of DiffServ is that traffic classes are accessible without signaling so that scalability is achieved, and no set-up delay is imposed on the applications. The overhead in the routers is relatively small, since it does not follow a linear (or even higher) increase with the number of application flows. Classification of traffic is performed at domain boundaries where concentration is relatively small, and the core of the networks is only required to perform class-based packet forwarding. In addition, the end-systems do not get involved in packet classification. This is being done at the ingress point of the provider network, so that security functions can be more easily applied, and at the same time the overhead imposed can be kept at a minimum. The basic problem with DiffServ is that the key to provisioning is the knowledge of traffic patterns traversing each node of the network, as well as the knowledge of network topology and routing [Mathy00]. Since network and traffic dynamics occur on a faster time-scale than Internet provisioning, it is likely that unless resources are over-provisioned, violation of service agreements will be inevitable, especially when attempting to provide quantitative services. Another drawback of the architecture is that bandwidth is shared between all flows in a single class and further discrimination and preferential treatment of traffic within a class is not possible. In addition, SLAs have a static form that can result in network underutilization [Blake98]. On the other hand, dynamic SLAs would require complex signaling and would lead very close to a scenario that is as complex as the IntServ/RSVP approach discussed above.

3.3 Multi-Protocol Label Switching (MPLS)

Together with the QoS architectures described above, there are also recent protocol deployments that offer some QoS functionality, based around the idea of label switching. MPLS (RFC3031) was mainly the result of efforts to effectively match IP over ATM networks. It tries to integrate layer 2 switching and layer 3 datagram forwarding. Within an MPLS network, a label and a Forwarding Equivalence Class (FEC) are assigned to each packet when entering the network, and then all forwarding decisions are based on these values. Packet forwarding is performed on a hop-by-hop basis and Label Switched Routers (LSRs) simply perform label swapping and take local decisions about the next hop that the packet should be addressed to. The basic motivation for MPLS is that it provides for simplified forwarding based on the match of a short label, and that it also provides for efficient explicit routing carried only at the time a label switched path is set up, rather than within each packet. MPLS provides for traffic engineering by selecting paths chosen by data traffic in order to balance the traffic load within the network, a process particularly difficult to accomplish with pure datagram forwarding. It also provides for QoS routing, where a route for a particular stream is chosen in response to the QoS required for that stream [Callon99]. MPLS can support QoS on a per-user basis by assigning per-user labels to packets, or on a per-flow basis by detecting and assigning appropriate labels to individual flows. Labels can make use of a Class of Service (CoS) field, which offers the flexibility of choosing between coarse or fine–grained QoS support. On the other hand, MPLS raises some scalability concerns, when it is to support label assignment for short flows, and its normal operation can be assured only for well-managed environments, because of its complex mechanisms. MPLS is favored by telecommunication operators who were traditionally basing their services on top of ATM, but it is doubtful whether it can provide end-to-end QoS solutions across large networks, and consequently in the Internet.

3.4 Internet Protocol version 6 (IPv6)

IPv6 (RFC1883) deployment became a necessity when the IPv4 world realized that it was running out of address space. Some people claim that with the 128-bit addresses that IPv6 provides, the world will have enough IP addresses to provide for every electronic device within each and every household. In addition to the longer address fields, IPv6 designers modified the protocol header, in order to minimize the processing time as well as to include the potential to support differentiated services within the network. The Traffic Class field can be used to identify and discriminate traffic types, and the Flow Label field can be used to enable labeling of packets that belong to particular traffic flows.
Another interesting option of IPv6 is its extension headers. Options can now be added on demand at the end of the IP header, and therefore reduce the common type processing of the packets. In addition, new types of headers can be introduced in due course, in order to provide for additional functionality. IPv6 also offers built in support for security and authentication, by providing relevant extension headers. On the other hand, there are people who believe that apart from the extended address space, there are no mechanisms introduced to IPv6 that cannot be retrofitted into IPv4. They claim that the only useful mechanism is that of the Flow Label, which is not yet fully utilized. Furthermore the existing transport protocols need to be modified in order to co-exist with IPv6, since they include IP addresses in their checksum computation.

4. Discussion

As seen in the previous section, all of the proposals and implementations have their strengths and weaknesses and no one seems to be able to be applied "as is" and significantly provide for QoS assurance and/or delivery guarantees to Internet traffic. DiffServ seems to be the most promising of the two QoS architectures, mainly because of its scalability properties. There is currently some on-going work in combining and integrating some of these technologies, trying to achieve the best out of two worlds. The combination of DiffServ and MPLS could play an important role in provisioning some form of quantitative service guarantees. DiffServ Behavior Aggregates (BAs) can be mapped onto Label Switched Paths (LSPs) to provide for traffic engineering and for protection within a network. Because MPLS is path-oriented it can potentially provide faster and more predictable protection and restoration capabilities in the face of topology changes, than conventional hop by hop routed IP systems [Fausheur01]. Another interesting combination is a possible IntServ/DiffServ interoperation (RFC2998). A potential framework could provide end-to-end quantitative QoS by applying the IntServ model end-to-end across a network containing one or more DiffServ regions. DiffServ regions may or may not participate in end-to-end signalling. IntServ enables hosts to request per-flow resources along end-to-end data paths and to obtain feedback regarding the admissibility of these requests. DiffServ enables scalability across large networks [Bernet00].

4.1 The Link to Network Management

It is our belief that QoS provisioning is closely related to Network Management. Today's networks are mostly unmanaged, whereas when network management is considered, it is restricted to the level of managing the devices of the network (e.g. routers, servers) as independent entities. From our point of view, there is a management hierarchy, which begins from the resources (CPU, storage, communications), continues to the devices (usually managed by SNMP or Web management) and ends up at the services offered by a network, which involve application flows traversing the network and sessions established on top of the network. Of course, managing the infrastructure is a pre-requisite that must be satisfied, before we can try to manage the QoS for individual or groups of flows.

4.2 Proposed Research

Our research focuses on the threefold of Measurement, Monitoring and Control (MMC) of the services provided by a next generation network, and we will try to exploit the advantages of recent technologies in order to enable QoS provisioning. We believe that the mechanisms provided by IPv6 could be combined and used for this purpose, because of the end-to-end operation of the Internet Protocol. Mechanisms deployed at the IP level are more likely to be ubiquitously applied in an Internet environment. By exploiting the concept of Flow Labels we can assign unique identifiers to flows that might require special treatment, or mark packets that belong to different levels of hierarchical encoding. This technique could provide both for traffic management, as well as for accounting and billing purposes. In addition, IPv6 Extension Headers could prove to be another powerful tool for similar purposes. The way Extension Headers are implemented offers the flexibility of defining new ones, serving specific purposes. These headers, inserted in a packet of a monitored service can trigger processing and actions at identified nodes.

Our goal would be to design a Measurement, Monitoring and Control Framework (M^2CF), which would enable for both performance measurement and control, as well as for accounting measurements and charging. We will therefore establish a taxonomy of services based on both technical and organizational trends. That is, the selection criteria will be influenced by the users' connection capabilities, but at the same time by users' needs and desires, their willingness to pay for better than best-effort services, and by commercial trends. Therefore, SLAs won't be flat, but instead will be influenced by physical characteristics, QoS requirements, value-add, and content provision. Figure 1 sets the broad scene of our proposed environment and highlights the inter-relations between the different participating entities. Taking into consideration that the user would wish to deal with a single provider, and therefore receive one unified bill for the offered services, our architecture will involve a single SLA between the ISP and the customer. Potential third-party providers (providing either content or value-added services) will establish individual agreements with the ISP, and the value of these services will be integrated into this SLA. The agreements between the various providers, as well as specific charging and billing schemes are out of the scope of our initial research investigation. However, we intend to define both the form of the ISP-customer SLA and the mechanisms through which measuring and monitoring will be performed. We believe that based on these mechanisms, charging can also be applied and monitored.

Our work will initially be towards defining what we mean by the terms "content" and "value-added" services in this context. These terms are complementary, rather than distinct. A value-added service can be the necessary infrastructure and overall software environment to support the provision of a specific type of content. As an example, consider a service offering on-line weather forecasting. The requirements would be to adapt the types of data transmitted to each user, according to his/her connection capabilities. So, calling the service from a mobile phone would result in a short descriptive message being displayed, whereas calling it from a desktop personal computer connected to a high-speed corporate network would result in a high resolution satellite weather map image, accompanied with the relevant information. The overall infrastructure to provide this service and make it customisable for individual users can be considered as being the value-add, and can be provided either by the ISP or by a third-party provider. Weather information is considered to be the content transmitted through this service, and can be provided by a totally independent organizational entity.

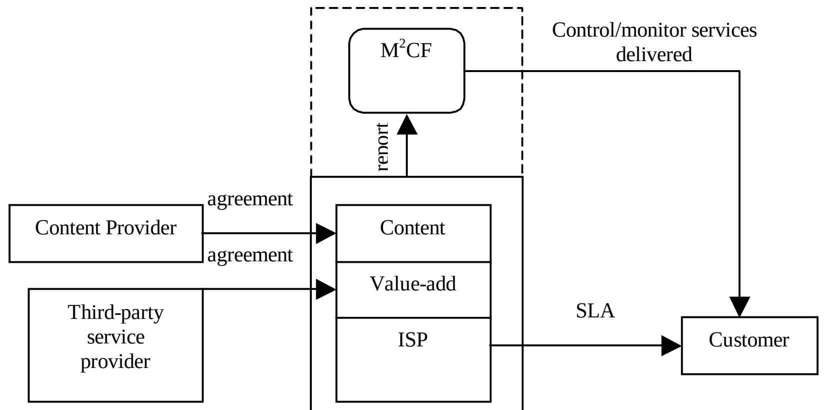

Figure 1: *Inter-relationships between multiple entities in a service-based management environment*

The definition of both content and value-added services can vary from digital broadcasting and QoS-enabled multimedia services, to any form of services offering intellectual property material. So, examples of value-added services can be live broadcasting of video and/or audio, Multimedia on Demand (MoD), on-line digital libraries (e.g. on-line book-store), remote conferencing with service delivery guarantees. The QoS parameters will vary from one service to the other, and will be individually defined. We will also investigate how, for example, a hypothetical premium service could be provided; a pre-requisite could be for the user to have either a premium SLA or a relatively high-speed connection. Alternatively, the same categorization of services could be adapted to different types of connections and/or SLAs. As an extension, we may consider whether specific QoS requirements (e.g. low jitter levels, high bandwidth links) will be by default associated with specific services, or users will request QoS values, either on demand or through their SLA. The SLAs will initially be static, but since we aim to produce an extensible networked environment, we will explore the potential of adding dynamism to the service contracts, so that users can be offered services or preferential treatment for their traffic on demand, and their SLAs adjusted (amended) appropriately.

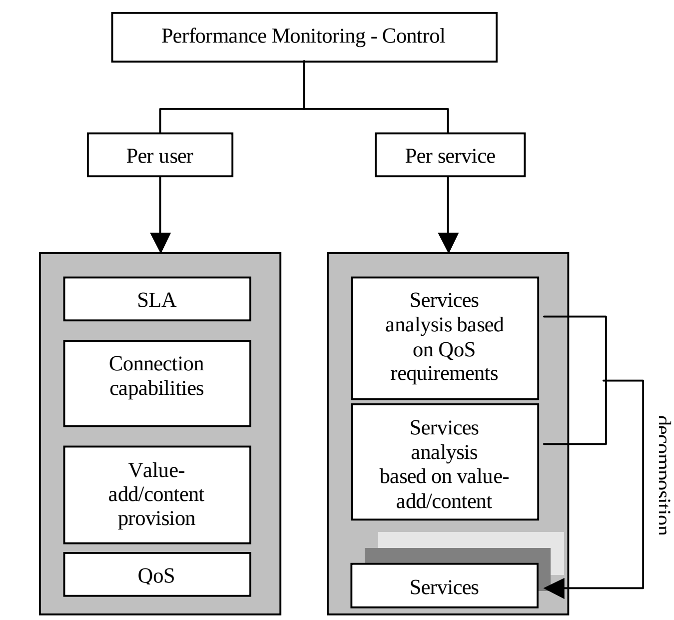

Figure 2*: Measurement, Monitoring and Control Framework (M^{2}CF)*

The approach towards the establishment of the Measurement, Monitoring and Control Framework (M^2CF) can be summarized in figure 2. Performance monitoring and control will be performed on a per user basis as well as on a per service basis. This way we will try to create an environment offering differentiated classes of service, based both on the types of traffic that are present and on a customer's demands and willingness to pay. In the SLA, the customer will specify the sorts of services he wishes to be provided, together with his/her QoS requirements and connection capabilities. In parallel, the services will be decomposed down to distinct classes based on their content/value-add and the QoS properties they need, in order to be offered seamlessly. We believe that this two-dimensional customizable behavior from the network can be achieved by exploiting the IPv6 mechanisms mentioned above. We plan to define a special purpose Extension Header to be used for triggering management actions at identified nodes, for identified types of traffic. Its format, as well as whether it will be a totally new header or a variation of an existing one, are subjects of further research. Furthermore, the possibility of taking advantage of the currently loosely defined IPv6 options, and maybe defining new ones serving for our management purposes, is still open.

5. Conclusion and Further Work

This paper examined some of the aspects of QoS provision within today's and the future Internet. After discussing some essential points of QoS and establishing our own views concerning the broad area, we have briefly examined the architectures and protocols designed in the recent years, in order to provide for an environment where differentiated classes of service are present and extend (or even replace) the flat best-effort service currently offered by the Internet. We then set the outline of a research path that we intend to follow, and which we believe that will contribute towards the realization and implementation of QoS-enabled IP networks within the Internet. After establishing our initial research goals, we plan to build an IPv6 test bed, in order to experiment with the identified mechanisms, and apply policies to traffic with different characteristics and requirements, in an effort to create an extensible QoS-enabled environment for next generation IP networks. We believe that by exercising the stateless IPv6 mechanisms, we will achieve dynamic and adaptable SLAs and therefore provisioning, without resulting in the overheads of signaling experienced with reservation protocols. This way, we can potentially be able to create a taxonomy of profiles for the customers, providing for adaptable services, depending on the conditions under which a user is connected at a particular time. So, under an integrated SLA, we will be able to offer services to the same user irrespective of whether he/she uses a desktop PC connected through a high-speed corporate network, a laptop using a dial-up connection, or a mobile phone. Ultimately, we will try to generalize our model to function within a broad DiffServ architecture, so that we facilitate interoperation between multiple QoS-enabled domains.

Acknowledgements

We are grateful to Agilent Technologies for the support of Dimitris Pezaros' work through an industrial fellowship. Many thanks are due to Dr. Joseph Sventek and Dr. Francisco J. Garcia for their influence and contribution to this research. The authors would like to specially thank Dr. Christopher Edwards for kindly reviewing and commenting on an earlier version of this paper.

References

[Bernet00] Bernet, Y., Ford, P., Yavatkar, R., Baker, F., Zhang, L., Speer, M., Braden, R., Davie, B., Wroklawski, J., Felstaine, E., *A Framework for Integrated Services Operation over DiffServ Networks*, IETF, ISSLL Working Group, RFC2998, November 2000

[Blake98] Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z., Weiss, W., *An Architecture for Differentiated Services*, IETF, DiffServ Working Group, RFC2475, December 1998

[Braden94] Braden, R., Clark, D., Shenker, S., *Integrated Services in the Internet Architecture: an Overview*, IETF, IntServ Working Group, RFC1633, June 1994

[Callon99] Callon, R., Doolan, P., Feldman, N., Fredette, G., Swallow, G., Viswanathan, A., *A Framework for Multiprotocol Label Switching*, IETF, MPLS Working Group, work in progress, draft-ietf-mpls-framework-05.txt, September 1999

[Deering95] Deering, S., Hinden, R., *Internet Protocol, Version 6 (IPv6) Specification*, IETF, IPNG Working Group, RFC1883, December 1995

[Fausheur01] Fausheur, F., Wu, L., Davie, B., Davari, S., Vaananen, P., Krishnan, R., Cheval, P., Heinanen, J., *MPLS Support of Differentiated Services*, IETF, MPLS Working Group, work in progress, draft-ietf-mpls-diff-ext-08.txt, February 2001

[Huston00] Huston, G., *Internet Performance Survival Guide*, Wiley Computer Publishing, 2000, ISBN 0-471-37808-9

[Mathy00] Mathy, L., Edwards, C., and Hutchison, D., *The Internet: a Global Telecommunications Solution?*, IEEE Network, July/August 2000