A Functional Architecture for Quality-of-Service in Multi-domain Networks*

Elena Pagani and Gian Paolo Rossi

Dipartimento di Scienze dell'Informazione, Università degli Studi di Milano
via Comelico 39, I-20135 Milano, Italy
{pagae, rossi}@dsi.unimi.it

Extended Abstract

The tremendous diffusion of the Internet is making IP an attractive platform to support multimedia traffic too. Unfortunately, despite the continuous growth of the link capacity and of the switching power of the routers, the best effort nature of IP is not sufficient, alone, to ensure the required quality to that traffic. Several initiatives have been undertaken in recent years to bridge this *Quality Of Service* (QoS) gap. Protocols like RSVP [1] and RTP [2], and the definition of *integrated services* (int-serv or IS, for short) [3] and *differentiated services* (diff-serv or DS, for short) [4] are just a few examples of the efforts in this direction. So far, however, the standardization process is far to be completed, the available proposals do not refer to a common architectural framework and the literature is fragmented. Moreover, the available protocols apply to single, homogeneous, QoS-domains, while the understanding of the inter-domain communications is at a preliminary stage. Although many multimedia applications are *multicast* in nature, the problems involved in supporting multicast communications with QoS are still unclear.

We have characterized a functional and architectural framework for end-to-end QoS in large multi-domain networks [5], where both IS and DS may be deployed. The interconnected system we consider is composed of a set of networks connected by *QoS-capable* routers, i.e., routers that implement the QoS services. We call *domain*, a set of contiguous networks which operate with a common set of service differentiation models, provisioning policy and service definition. We denote as *border* routers those routers connecting contiguous network domains. Border routers perform the mapping between the services and forwarding policies provided in adjacent domains, to guarantee the requested end-to-end QoS to the flows traversing the domain boundaries. We assume a three-layer routing architecture: routing is performed within a single network, at *intra-domain* level (i.e., among a set of homogeneous networks) or at *inter-domain* level (i.e., among domains).

The framework we propose provides a common model to which the QoS initiatives can refer and which is helpful to identify duplicate functions and the relationship among functional modules. It guarantees backward compatibility with existing protocols and services. It characterizes the building blocks to design

* This work was supported by the MURST under Contract no.9809321920, "Transport of multicast packets with QoS guarantees".

the architecture of the routers operating at the border between domains adopting different differentiation policies. We have expanded this architecture to consider the case of QoS multicast traffic and to provide a preliminary understanding of the architecture of a QoS and multicast capable border router.

The QoS architecture is composed by three layers. Each layer consists of *control* modules, for the service set-up, maintenance and monitoring, and of *data management* modules, for the transmission and delivery of the QoS data flows.

The functional modules in the *end-to-end* layer operate on top of the UDP protocol. The data management modules perform the fragmentation (`fragmentation` module) of the applications messages and the marking (`translator`) of the data packets for the appropriate processing at the lower layer. At the destination, the `delivery` module performs the delivery of the data to the upper layer according to the specified QoS requirements. The control modules perform policy and accounting functions (`policy control`), monitor the supplied end-to-end QoS (`meter`) and provide the applications with an abstraction of the underlying QoS implementation (`flow manager`). The `flow manager` also performs the configuration of both the local `translator` and the `delivery` module at the destination.

At the *network* layer, above the IP protocol, the data management modules perform the classification (`classifier`) and forwarding (`traffic conditioner`) of the data packets according to the QoS they expect. The `traffic conditioner` also takes in charge the management of the traffic peaks and the metering of the QoS provided by the output link. The functionalities in charge to the control modules concern the set-up of the service for a new data flow, namely, the check for the availability of local resources (`admission control`) and their reservation (`reservation` module). The `reservation` module also configures the `packet scheduler` module at the *data link* layer, so that it implements the appropriate packet forwarding policy.

When the need of supporting multicast communication arises, the proposed framework becomes more complex. The issues involved by multicast affect the functionalities offered by both the end-to-end and the network layer. The end-to-end modules are required to be group-aware and must compensate the dishomogeneity among different destinations. This is achieved by having the `flow manager` that properly configures the group of `delivery` modules, according to the statistics gathered by the `meter` from all the destinations. Moreover, the `translator` may adopt different data encodings depending on different resource availability along the paths to the destinations. At the network layer, both a membership service and a multicast routing service are needed by the `reservation` module. In the case of a hierarchical routing architecture, such as that we consider, those services are replicated at each level of the hierarchy. The `traffic conditioner` is modified so that it separately forwards and manages the data flow on the multiple output links.

The above defined common framework is mainly intended to ease the deployment of new QoS-sensitive infrastructures and to expedite the interoperability among etherogeneous domains. In [5] we apply the proposed architecture to border routers to show how interoperability can be obtained between int-serv and diff-serv. The framework also highlights a few critical aspects of the current QoS process; the most important of them is that QoS is introducing extra complexity in the routers in terms of both time and resources consumption. The complexity is highly concentrated during the set-up phase. This is a hint to privilege the diff-serv approach, which allows to statically define the service levels and resource reservations and to change them periodically (e.g.,monthly). Dynamic creation of service classes or of int-serv flows seems unfeasible with the current technology when scaling to large amount of users in a multi-domain network.

We are currently evaluating the performance of some protocols proposed in the literature that implement the `reservation` and `multicast routing` modules at the network layer. In the future, we plan to investigate the problem of realizing the mechanisms and the end-to-end functional modules needed to support end-to-end QoS multicast communications in a diff-serv environment.

References

1. Zhang, L., Braden, R., Berson, S., Herzog, S., Jamin, S.: Resource reSerVation Protocol (RSVP) - Version 1 Functional Specification. RFC 2205 (1997). Work in progress.
2. Schulzrinne, H., Casner, S., Frederick, R., Jacobson, V.: RTP: A Transport Protocol for Real-Time Applications. RFC 1889 (1996). Work in progress.
3. Braden, R., Clark, D., Shenker, S.: Integrated Services in the Internet Architecture: an Overview. RFC 1633 (1994). Work in progress.
4. Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z., Weiss, W.: An Architecture for Differentiated Services. Internet Draft, draft-ietf-diffserv-arch-02.txt (1998). Work in progress.
5. Pagani, E., Rossi, G.P.: A Functional Architecture for End-to-end Quality-of-Service in a Multi-domain Network. Technical Report 231-99, Dipartimento di Scienze dell'Informazione, Università degli Studi di Milano (1999).