

# QoS Management for All-IP Core Networks

## All-IP 核心網路之服務品質管理

Yao-Nan Lien, Chien-Tung Chen, Ming-Chih Chen, Tsung-Hsung Li, and Yi-Min Chen  
Computer Science Department  
National Chengchi University  
Taipei, Taiwan, R.O.C.  
lien@cs.nccu.edu.tw

### Abstract

This paper discusses the core network design under the Budget-Based management infrastructure, *BBQ*<sup>1</sup>, which is designed for All-IP networks to offer end-to-end QoS assurance to their services. Under BBQ scheme, the quality bound of each component network is controlled based on a calculated budget plan. End-to-end QoS is assured by a global QoS management agent. The management issues include software architectures in different layers, class based admission and resource reservation policies, as well as resource management infrastructure, policies, and mechanisms. The objective of this infrastructure is to facilitate network operators to tune their networks with a great flexibility and scalability to achieve their own operational objectives. One of the major components in an All-IP network is the backbone network which consists of some core network. In this paper, the design philosophy of core networks under BBQ will be discussed in details.

**Keywords:** All-IP Network, QoS, VoIP.

### 1. Introduction

An *All-IP Network* uses a single IP based packet-switched network to carry all types of network traffics [1,8,15]. This revolutionary All-IP network not only reduces network deployment and management costs, but also offers a great opportunity opening for various new services that are not possible on the conventional separated networks. However, running time-sensitive services such as VoIP on packet-switched networks may suffer from poor quality problem due to long delay time, large jitter, and high packet loss rate. To make All-IP networks possible, QoS is a critical problem yet to be overcome [3,4,5,14,15].

### 1.1. A Simplified All-IP Network

Without loss of generality, we assume the following simplified All-IP network architecture. A world-wide All-IP network consists of several core networks interconnected together through some interconnection links (e.g. cross Pacific undersea cables/fibers) and some number of stub networks (also named *access networks*) connected to core networks. A *core network* consists of some *Interior Routers (IR)* and some *edge routers*. An *edge router* is also an *Inter-Domain Gateway (IG)* if it is connected to another core network. A *stub network* is connected (attached) via an *Access Gateway (AG)* to an edge router, called the *Border Gateway (BG)*, of one (or more) core network. Typical stub networks are WLAN, GPRS, 3G, and conventional local loops. A service request, which may be a phone call, a video stream, or a file transfer, will be converted into IP packets first when it enters the network and be converted back to the original format when it leaves the network. Depending on the admission policy, when a packet is admitted into an All-IP network, it will enter the Entrance stub network, and will be forwarded to the first core network, the second one, etc., to the Exit stub network, and finally to the destination. Although, in reality, a stub network may attach to more than one core network for various reasons such as availability, we assume a stub network only attaches to one core network for simplicity. The network architecture is depicted in the Fig. 1.

### 1.2. Paper Organization

The related work will be shown in Section 2. Section 3 shows the overall BBQ infrastructure. Section 4 and 5 discuss the QoS management schemes for end-to-end QoS and core networks respectively.

---

<sup>1</sup> This research is supported by NSC grants NSC 92-2219-E-004-001,002 and NSC 91-2219-E-004-001,002.

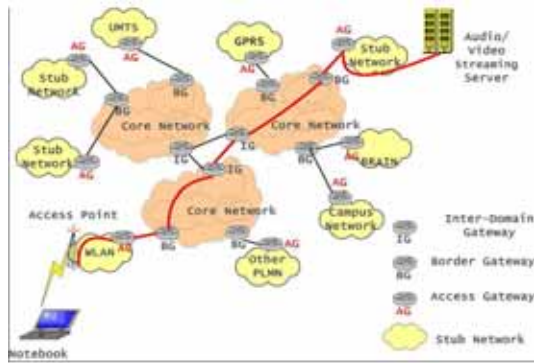


Fig. 1. A simplified All-IP Network.

## 2. Related Work

The two most popular QoS technologies are Differentiated Services (DiffServ) and Integrated Services (IntServ) [2,6,7]. The heart of IntServ is RSVP (Resource Reservation Protocol) [19,20]. Before admitting a service request, IntServ first reserves needed sources along the path selected for the request. It can provide end-to-end QoS with higher confidence, but it suffers from scalability due to tremendous management overhead. On the other hand, DiffServ can avoid scalability problem. DiffServ is a protocol for specifying and controlling network traffic by class so that certain types of traffic, such as voice, get precedence [2]. The major advantage of DiffServ is its simplicity and easy to implement. However, the end-to-end behavior is not controlled. Extra mechanisms are needed to enhance the QoS assurance to the end-to-end and per-flow level [16].

The two most famous large scale efforts trying to provide end-to-end QoS for All-IP Networks are TEQUILA and AQUILA [1,4,15]. TEQUILA (Traffic Engineering for Quality of Service in the Internet, at Large Scale) is a project partially funding by the European Commission [15]. The objective of the project is to study, specify, implement and validate a set of service definition and traffic engineering tools to obtain quantitative end-to-end QoS guarantees through careful planning, dimensioning and dynamic control of scalable and simple qualitative traffic management techniques within the Internet (i.e., DiffServ). AQUILA (Adaptive Resource Control for QoS Using an IP-based Layered Architecture) is another European project aiming to provide end-to-end QoS to IP Networks [1,4]. The goal of this project is conception, design and development of an architecture to enable different service classes in the Internet.

Perhaps the most popular practical approach to solve the QoS problem is *over provisioning*, which

allocates extra capacity to the services to ensure their quality. This approach can save much effort on the network management and works very well nowadays because the bandwidth utilization on backbones is fairly low. However, it is not a sustainable solution for various reasons. First, there are many segments in an end-to-end path and some segments such as wireless access channels may have limited bandwidth, the cost of over provisioning on every segment is prohibitively high. Secondly, as a common sense, the demand of bandwidth grows much faster than that of network bandwidth. Many potential bandwidth-hungry services are actually confined by the last miles. Once the upward bandwidth of local access channel (e.g. ADSL) expanded, a tsunami scale surge of traffic will soon congest the backbone. Therefore, over provisioning is only applicable on very limited network and business situations.

## 3. Budget-Based QoS Management Architecture

### 3.1. Design Philosophy

#### Budget Based QoS Management

Based on the simplicity principle, BBQ requires each network component be able to guarantee a committed quality. The quality bound of each component network is controlled based on a calculated budget plan. End-to-end QoS will then be assured by a global QoS management agent, which will be discussed later. We assume each network router has DiffServ like capability. BBQ is actually a software layer above DiffServ domain. DiffServ routers take instructions from the QoS managers of upper layers and set the appropriate DiffServ parameters and QoS policies.

#### Pre-Planning vs. On-Demand Managements

In order to maximize network performance and to minimize service response time, many of management mechanisms in BBQ, such as resource allocation and reservation, take pre-planning approach, instead of real-time on-demand approach.

Pre-planning approach requires an accurate traffic forecast. Previous study shows that aggregated traffic on core networks usually demonstrates some repeated statistics pattern. For instance, the traffic statistics of most Monday noon are very similar. Based on this assumption, a network planner could use historical traffic statistics to forecast incoming traffics in the future. The granularity of the time

interval between two forecasts can be determined by the operators based on real network traffic statistics. On the other hand, the forecast error caused by the inevitable traffic fluctuation may hurt the management objectives. In this project, we propose several methods to compensate the performance degradation caused by forecast errors.

### Class Based Service Policies

For time-sensitive and connection-oriented (TSCO) service requests such as Conversational and Streaming classes, the admission control agent in BBQ will proceed with a light weight *call setup procedure* to designate a path and to reserve the required resources to assure the demanded end-to-end quality. For other types of services, BBQ does not reserve any resource. Instead, it takes best-effort policy to serve time-insensitive services. Operators determine their class-based pricing structure to maximize their operation objectives, while users choose appropriate service classes based on the demanded quality and the costs they are willing to pay.

### Path Centric Per-Flow End-to-End QoS Assurance

To ensure end-to-end QoS for a TSCO service request, the admission control agent designates to the request a pre-planned path with sufficient resources reserved. All packets of the same TSCO request are delivered along the designed path. Since the quality of each link is guaranteed, a controlled path will be able to guarantee the end-to-end quality level. In this way, per-flow end-to-end QoS is guaranteed. This path centric QoS mechanism is similar to the virtual circuit in some network components such as ATM. However, the end-to-end paths in BBQ are only pre-calculated, but not reserved until individual service requests arrive.

To reduce real-time overhead in the resource reservation procedure, we partition the reservation into two phases. In the first phase, pre-planning phase, each edge router of each core network is allocated with certain amount of *short-paths*. A short-path is a path from one edge router to another in the same core network. Each short-path has bandwidth reserved and quality level guaranteed. At the second phase, the time of admission, the admission control agent first selects a pre-planned end-to-end path that meets the bandwidth and quality requirements, and proceeds with short-path reservations. Since an end-to-end path may travel only a handful core networks, the real time overhead for the reservation procedure can be greatly reduced as compared to the conventional link based

reservation such as IntServ/RSVP protocol. The detailed design will be shown later.

### 3.2. Bearer Service Hierarchy

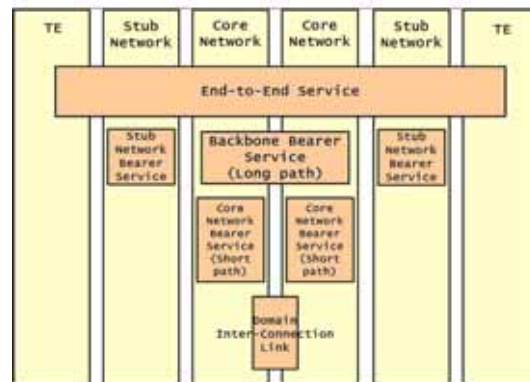


Fig. 2. Bearer Service Hierarchy

As depicted in Fig. 2, an end-to-end service is carried by many smaller bearer services. Each core network provides a short-path bearer service, and a long-path is the combination of all short-paths that a packet travels. Adding together the bearer services provided by the Entrance and Exit stub networks, an end-to-end bearer service is formed. In this way, with piecewise QoS assurances provided by smaller bearer services, the end-to-end QoS is guaranteed.

### 3.3. QoS Management Hierarchy

Table 1 shows the QoS management hierarchy in BBQ.

Based on the simplicity principle, BBQ organizes the software agents in different network components into layered management hierarchy. The end-to-end QoS assurance responsibility is then decomposed into smaller pieces and distributed to many agents in different layers. With autonomous authority within the designated responsibility, each agent may make some decisions by themselves without any negotiation with other entities. The response time to a service request can be greatly reduced while the resource efficiency can be maintained.

### 4. End-to-End QoS Management

The most important task in the End-to-End QoS Management functionality is to plan a set of long-paths to meet the quality requirements for anticipated service requests. Core network operators then provision their own core networks based on the forecasted demand. Since an All-IP network is a federation of many sub-networks, we assume there is

no any global network planer existing to plan the entire network from the global viewpoint. Network planning has to be performed by all core network operators in a cooperation (distributed) manner, instead.

Table 1. QoS Management Hierarchy

Management Layer	Managing Agent	Responsibility
End-To-End Resource Coordination	Long-path planning agent of each core network	plan long-paths with various quality level and provide information for long term network capacity provisioning
End-To-End QoS Control	admission controller of Entrance stub network	select appropriate end-to-end paths, short-path reservation, and perform admission control
Sub-Network Resource Management	Bandwidth Broker of each core network	allocate resources, e.g. bandwidth of links, to the resource mediators, e.g. edge routers in a core network, RRM in a 3G access network.
Sub-Network QoS Control	admission controllers of each stub/core network	execute the QoS policy of the sub-network, e.g. admission control, load control, routing and path selection, packet scheduler, etc.

### 4.1. Long-Path Planning

Each core network has a *Core Network Controller* (CNC) to perform all centralized management operations. One important component in CNC is the *Long-Path Planning Agent* (LPPA). The LPPAs of all core networks work together to compute all possible long-paths that meet the QoS requirements of potential requests. Based on the obtained results and the traffic statistics, the agent forecasts the capacity demand for the short-paths it may use in the future time slots. The long-path planning procedure is as follows:

1. Each core network publishes the specification of its short-paths;
2. The LPPA in a core network computes the best long-paths for all potential service requests that will be originated from this core network;
3. Based on the computed results and the traffic statistics, the agent forecasts the

capacity demands for the short-paths it may use in the future;

4. Each core network collects forecasted demand for short-paths and provisions the network with sufficient capacity.

Note that it is impractical to make long-path reservation at this stage since a long-path may cross several core networks in several different countries. Long-path planning in BBQ is only a procedure to compute end-to-end paths and to forecast bandwidth demand for network operators to provision their networks.

### 4.2. Global Admission Control

To reduce call setup time, The global ACA at an Entrance stub network uses a light weight on-demand procedure to reserve a long-path for each request as depicted in Figure 3:

1. From the long-path table, select an appropriate long-path.
2. Reserve a short-path from each of the core, Entrance, and Exit stub networks.
3. If fail, try another alternative long-path.
4. If no path is available, reject the request.

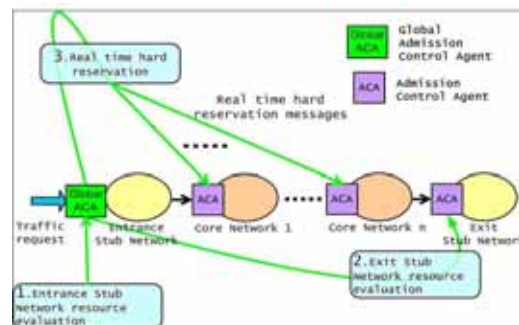


Fig. 3. Global Admission Control Procedure.

To reduce call setup time, BBQ uses a light weight on-demand procedure to reserve a long-path for each request as depicted in Fig. 6.

Since a long-path may travel only a handful of core networks, the overhead for the above procedure will be very low so that it is applicable for large scale networks. The selection of appropriate long-paths for service requests is a typical optimization problem. We model the problem as an integer programming problem and relax it into a linear programming problem. The objective is to minimize the penalty caused by unsatisfying aggregates of traffic with the constraints of (1) fixed resource (Limited Short Path bandwidth); (2) traffic aggregate QDF budget must be satisfied; (3) allocate resource to traffic

aggregates; (4) trading off QDF with bandwidth in each short-path.

## 5. QoS Management for Core Networks

A core network is owned and operated by an independent operator. Under BBQ infrastructure, each core network is responsible to provide many QoS assured short-paths. The traffic that is sent into a short-path will travel along that short-path so that its quality will be guaranteed. The Global ACA of each stub network will perform admission control so that it will not send too much traffic to the network. Under this circumstance, each core network will be able to guarantee the QoS level for all admitted traffic flows.

A key philosophy in designing QoS management scheme for core networks is to reduce the real time per-flow reservation overhead by batch order. Under this scheme, each edge router is pre-allocated with some short-paths and equipped with an admission control agent (ACA) to perform the admission control autonomously. BBQ proposes several resource allocation mechanisms to best utilize network resources in order to achieve the maximum performance.

### 5.1. Software Architecture for Core Networks

We assume each core network has DiffServ-like capability. With respect to a traffic flow, the edge router that receives the flow is called the *Ingress* and that sends flow out of the network is called the *Egress*. The Ingress of a traffic flow may also be the Egress of another traffic flow. Other routers are Interior Routers. There is a central management module handling centralized policies such as resource allocation, etc. Under BBQ infrastructure, some software modules are designed to implement various functions that will be explained in the remaining of this section.

The software architecture for a core network is depicted in Figure 4.

There are three components in the central management module *Core Network Coordinator* (CNC):

- *Bandwidth Broker* (BB) controls and allocates link resources to Ingress routers.
- *Long Path Planning Agent* (LPPA) coordinates with the LPPAs of other core networks to plan long-paths.

- *Short Path Planning Agent* (SPPA) coordinates with Local SPPAs to compute short-paths based on the acquired link resources as well as allocates link resources to short-paths.

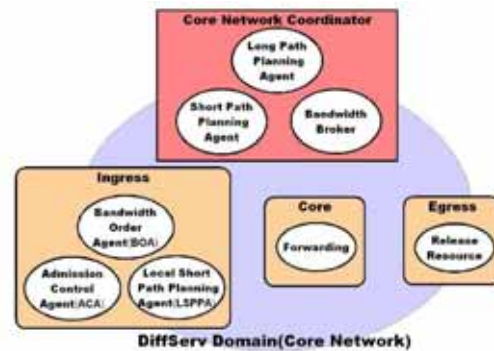


Fig. 4. Software Architecture within a Core Network.

There are also three components in each Ingress:

- *Bandwidth Order Agent* (BOA) acquires link resources from BB.
- *Local Short Path Planning Agent* (Local SPPA) coordinates with the SPPA in CNC to compute short-paths as well as to reserves link resources for the obtained short-paths.
- *Admission Control Agent* (ACA) manages the allocated short-paths and performs admission control.

### 5.2. Short-Path Planning Procedure

Time is divided into time slots. The time slot a planning process is concerning is called *Concerned Time Period* (CTP). Those time slots whose demand statistics are similar to the CTP are called *Reference Time Periods* (RTPs). Traffic statistics in RTPs with respect to a CTP is used to forecast potential incoming requests in the CTP.

Path planning can be done centralized for simplicity. However, a centralized planning approach may induce fairness problem such that some Ingress may be overly allocated with too much resource while some may be under allocated. On the other hand, a distributed planning approach may have less fairness problem and higher capability to deal with local traffic fluctuation, but is more complicated. BBQ proposes to use a hybrid approach. The SPPA in CNC initiates the short-path planning and allocation process first. If the result is acceptable according to some fairness criteria, the process terminates.

Otherwise, a distributed process is initiated. The entire procedure is as follows:

### (a) Centralized Phase

The SPPA in CNC computes a set of short-paths using a routing algorithm based on the path-based demand statistics in a RTP (the latest RTP, for instance.) If the result satisfies predefined criteria, terminates. Otherwise, proceeds with the following distributed process.

### (b) Distributed Phase

1. The Local PPA of each Ingress computes link-based resource demand by feeding path-based demand collected in RTPs into the short-paths that are obtained in the centralized phase.
2. For each link, BOA calculates the needed bandwidth according to an optimization objective and places an order to the BB to acquire it.
3. The BB proceeds with an allocation procedure to fulfill the orders placed by all the Ingresses within the same core network. If link resources are not sufficient to fulfill all orders, compromise must be made.
4. The Local LPPA of every Ingress constructs short-paths based on the acquired link resources. These short-paths will be used at the time of CTP.

The above procedure is depicted in Fig. 5.

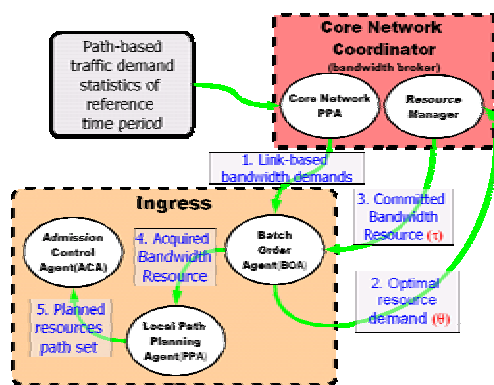


Fig. 5. Short-Path Planning Procedure within a Core Network.

## 5.3. Link Resource Allocation by Batch Order

In the Distributed Phase of short-path planning, each BOA must calculate and acquire the right amount of bandwidth to fulfill the demands of anticipated

TSCO traffic requests for each link. Since real-time on-demand resource allocation is usually more expensive than pre-allocation, an insufficient resource allocation may induce some extra cost for real-time on-demand resource acquisition. On the other hand, an over resource allocation will waste unused resources. Assuming traffic arrives in a repeated pattern and is statistically stable, we can use the traffic history to estimate the right amount of bandwidth to acquire. We have designed an optimization model aiming to maximize the expected profit as well as their solutions for BOA. The details can be found in [11].

When the BB receives the orders submitted from BOAs, it needs to determine the right amount of bandwidth to fulfill these bandwidth orders. The basic allocation policy is as follows. If the remaining bandwidth of a link is more than the sum of orders, it just allocates the demanded bandwidths to these orders. Otherwise, it divides the remaining bandwidth proportional to the demanded bandwidth and allocates them accordingly.

## 5.4. Compensation for Forecasting Errors

Traffic forecast may not be perfectly accurate due to traffic fluctuation and imperfect prediction method. These errors may lead to poor resource allocation such that a network may reject some service requests while it actually has some resources held unused. We have designed some mechanisms that can compensate forecast errors and thus may reduce performance degradation accordingly. The objective is to maximize resource utilization in terms of operational objectives [11,12,13].

Under BBQ, link bandwidths are allocated to edge routers and then to short-paths. An imperfect resource allocation may lead to undesired results that some edge routers are allocated more bandwidth than what are actually needed and some are allocated insufficient bandwidth. Two straightforward approaches are *central pool* and *distributed reallocation*. The central pool approach is to keep a small portion of link bandwidth at the BB for real-time on-demand allocation and to have edge routers making on-demand reservation when the preallocated bandwidth of any link is exhausted (or below a certain level). This approach has some drawbacks. First, on-demand resource allocation is expensive. Secondly, it is not easy to determine the right amount of bandwidth reservation for real-time on-demand allocation in order to minimize the unused bandwidth that is kept in BB and all edge routers.

On the other hand, in the distributed reallocation approach, edge routers perform bandwidth reallocation by themselves. Whenever the bandwidth of a link allocated to an edge router is exhausted, that router will negotiate with other edge routers for bandwidth reallocation. This approach is greedy and has less computation overhead. However, its reallocation efficiency is not predictable at all while it can't avoid the expensive real-time on-demand management overhead.

Under BBQ infrastructure, we have designed an innovative mechanism, *overbooking*, in which a BB can commit more resources to the requesters than what it possesses. Assuming that some edge routers may have some unused bandwidth due to forecasting error, we may be able to control the level of over-commitment such that the total actual bandwidth consumption for a link will not likely exceed the link capacity. The challenge is to determine the right level of over-commitment such that bandwidth demands can be all fulfilled and the probability of traffic overflow is minimized. We have designed an optimization model with good solution provided. The details can be found in [13].

## 6. Summary

This paper discusses the design of core network under BBQ management infrastructure, which is designed for All-IP networks to offer end-to-end QoS assurance to their services. Under BBQ scheme, the quality bound of each component network is controlled based on a calculated budget plan. End-to-end QoS is assured by a global QoS management agent. We designed a software architecture in layer manner each playing various roles, some class-based admission and resource reservation policies, a resource management infrastructure, some management mechanisms such as QoS based routing model and algorithms, as well as resource allocation models and tools. This infrastructure and the associated tools will facilitate network operators to tune their networks with a great flexibility and scalability to achieve their own operational objectives.

## References

1. AQUILA, <http://www.salzburgresearch.at>.
2. D. Black, M. Carlson, E. Davies, Z. Wang, "An Architecture for Differentiated Services", RFC 2475, Dec. 1998.
3. Nicolas Christin and Jorg Liebeherr, " A QoS Architecture for Quantitative Service Differentiation ", *IEEE Communications*, June 2003.
4. Thomas Engel et al., "AQUILA: Adaptive Resource Control for QoS Using an IP-Based Layered Architecture", *IEEE Communications*, Jan. 2003.
5. Janusz Gozdecki, Andrzej Jajszczyk, and Rafal Stankiewicz, "Quality of Service Terminology in IP Networks", *IEEE Communications*, Mar. 2003.
6. IETF RFC 1633, Integrated Service Framework (IntServ).
7. IETF RFC 2205, Resource reSerVation Protocol (RSVP).
8. Andrzej Jajszczyk, "Telecommunications Networking at the Start of the 21st", *IEEE Communications*, Jan. 2001.
9. Yao-Nan Lien, Hung-Chin Jang, Tse-Chieh Tsai and Hsing Luh, "BBQ: A QoS Management Infrastructure for All-IP Networks", *Communications of IICM*, vol. 7, no. 1, Mar. 2004, pp. 89-115.
10. Yao-Nan Lien, Chien-Tung Chen, "Budget-Based End-to-End QoS Management for All-IP Networks", NCCU-CS Tech. Report, Sep. 2003.
11. Yao-Nan Lien, Ming-Chih Chen, "Distributed Resource Management and Admission Control in Budget-Based QoS Management for All-IP Core Networks ", NCCU-CS Tech. Report, Sep. 2003.
12. Yao-Nan Lien, Tsung-Hsung Li, "Path Planning in Budget-Based QoS Management for All-IP Core Networks", NCCU-CS Tech. Report, Sep. 2003.
13. Yao-Nan Lien, Yi-Ming Chen, "Forecasting Error Tolerable Resource Allocation in Budget-Based QoS Management for All-IP Core Networks", NCCU-CS Tech. Report, Sep. 2003.
14. Neal Seitz, "ITU-T QoS Standards for IP-Based Networks", *IEEE Communications*, June 2003.
15. TEQUILA, <http://www.ee.ucl.ac.uk/~pants/projects/tequila/>.
16. Michael Welzl, Max Muhlhauser, "Scalability and Quality of Service: A Trade-off?", *IEEE Communications*, June 2003.