8th IEEE Workshop on Local and Metropolitan Area Networks, Berlin/Potsdam, August 25-28, 1996

An Approach to Hardware-Supported Accounting Management in ATM-Networks

Georg Carle, Günter Schäfer, Jochen Schiller
Institute of Telematics,
University of Karlsruhe,
D-76128 Karlsruhe, Germany

FAX: +49 721 388097
e-mail: [carle, schaefer, schiller]@telematik.informatik.uni-karlsruhe.de

Introduction

As widely accepted, ATM is one of the most important networking technologies in private and public networks. Advances in protocol implementation [Stee94, KrKS93] and hardware support for protocol processing [CaSc95b, Schi95] are beginning to overcome the performance bottleneck in the end systems. However, there is still a lack of support for efficient management and billing of high performance communication services. While accounting management is not yet a major topic in private networks, it is essential for public service providers. Up to now, many service providers perform billing based on transmitted cells, allocated bandwidth, and connection time. This type of billing is usually used in combination with strict Usage Parameter Control (UPC), where cells that violate the traffic contract are discarded by the service provider. Typically, such services provide strong guarantees for the service quality and in particular very low cell loss rates. However, there are a number of cases in which non-negligible cell loss rates occur, as, e.g., for UBR (Unspecified Bit Rate) services. Conventional billing concepts similar to billing of ISDN services are not appropriate in these cases, and sophisticated management and billing support will become more and more important as low-cost low-quality ATM services are to be sold. In addition, group communication services impose even more complex requirements for the management and billing. For the establishment of alternative billing concepts, the ability to perform accounting and billing based on the actual amount of transferred information is of high importance.

Within the Internet, accounting of successfully transmitted user data (goodput) is also important in order to achieve fair billing. General concepts for accounting within the Internet were presented in [MiHR91]. In [EdMV95], a system (called the billing gateway BGW) was proposed which is capable of billing users for their TCP traffic. While usage accounting within the Internet is based on IP packets, usage accounting in ATM networks should be based on ATM cells and is therefore computationally more demanding. Our approach to hardware-supported management is intended to provide the processing capability needed in ATM networks.

Errors in ATM networks

There are a number of cases in which cell error and loss rates may be sufficiently high in order to justify the additional effort of accounting user goodput. An important case for ATM services which show a significant error and cell loss rate are networks with wireless links. Another potential source for significant errors and losses are low-cost ATM hardware which does not allow to perform accurate traffic control. In addition, significant cell loss rates can occur in cases where a high statistical multiplexing gain is desired for highly bursty sources over high-speed, wide-area connections. In such a scenario, the following problems are responsible for the deficiencies of conservative traffic control algorithms with respect to achieving a high resource utilisation. During transient periods, sources are not able to obtain an accurate view of the current load state. Their view of the network load is always outdated due to the propagation delay. It can not be expected that the actions performed by a source without accurate view of the current load allow to optimise the quality of service parameters (throughput, delay, and loss). Aggressive control algorithms, as for example proposed in [KiFa95], are designed to achieve a high multiplexing gain, at the cost of potentially high losses. While conservative control algorithms may lead to long transient periods, aggressive control algorithms allow to shorten the transient period under certain conditions. Then they reach an equilibrium state faster, where the parameters can be adjusted according to optimality criteria [HeRo95]. One example for a control algorithm that allows to adjust its aggressiveness is the concept of loss-load curves [WiCh91], which can be used to derive strategies for sources in order to maximise throughput or minimise end-to-end delay. In these cases, resource utilisation, throughput and delay can be increased by accepting higher losses.

For multicast ABR services, different service models are possible, which allow similar trade-offs. A conservative multicast service would limit the transmission rate according to the most congested link of a multicast tree, allowing to achieve very low cell loss rates. In contrast, an aggressive ABR multicast service may use higher transmission rates, while producing cell losses at some congested links.

Error control for reliable services

When applications require a higher reliability than offered by the network, protocols with error control mechanisms must be used. For reliable data communication, protocols with Automatic Repeat Request (ARQ) mechanisms are widely used. ARQ mechanisms have several drawbacks when used in high speed wide-area networks, such as high delay in case of errors, and bad scalability for multipoint connections. An alternative approach is Forward Error Control (FEC), which has a number of advantages such as reducing delay and improving scalability for multipoint connections when used in high-speed wide-area networks. Appropriately dimensioned FEC allows to achieve an overall gain, as shown, e.g., by Biersack [Bier93]. However, FEC is not yet widely used over high speed networks, as there still remain a number of important questions such as how to dimension the amount of redundancy and the size of protocol data units. There exist a number of protocols with ARQ or FEC mechanisms which are suitable for ATM networks, and which all have individual strengths and weaknesses. The adaptation layer protocol RMC-AAL (Reliable Multicast ATM Adaptation Layer) provides frame-based ARQ, cell-based ARQ and FEC mechanisms for reliable point-to-point and point-to-multipoint services [CaSc95, CaZi95]. It is an extension of AAL5 and is suitable for a hardware-based implementation. It can be integrated into end systems and into AAL-level servers. We selected RMC-AAL as platform for an implementation of the fair billing scheme, as AAL-level servers can perform goodput accounting within the network, and are suitable for billing of multipoint services (see Figure 1).

Influence of Error Control on Billing

For billing, current networks do not distinguish between initially transmitted and retransmitted data. Therefore, usage-based billing may lead to unfairness in the case of a public ATM service with non-negligible cell loss rate, as users have to pay also for retransmitted data. Up to now, neither the network provider nor the user is able to distinguish between initially transmitted data and retransmitted data at the user network interface. Therefore, users currently have no alternative than paying for retransmissions caused by network errors.

The conventional approach for billing is the reservation on peak rate allocation or on sustainable rate in combination with peak rate and burst tolerance. In combination with Call Admission Control (CAC), this enables the service provider to guarantee a certain quality of service.

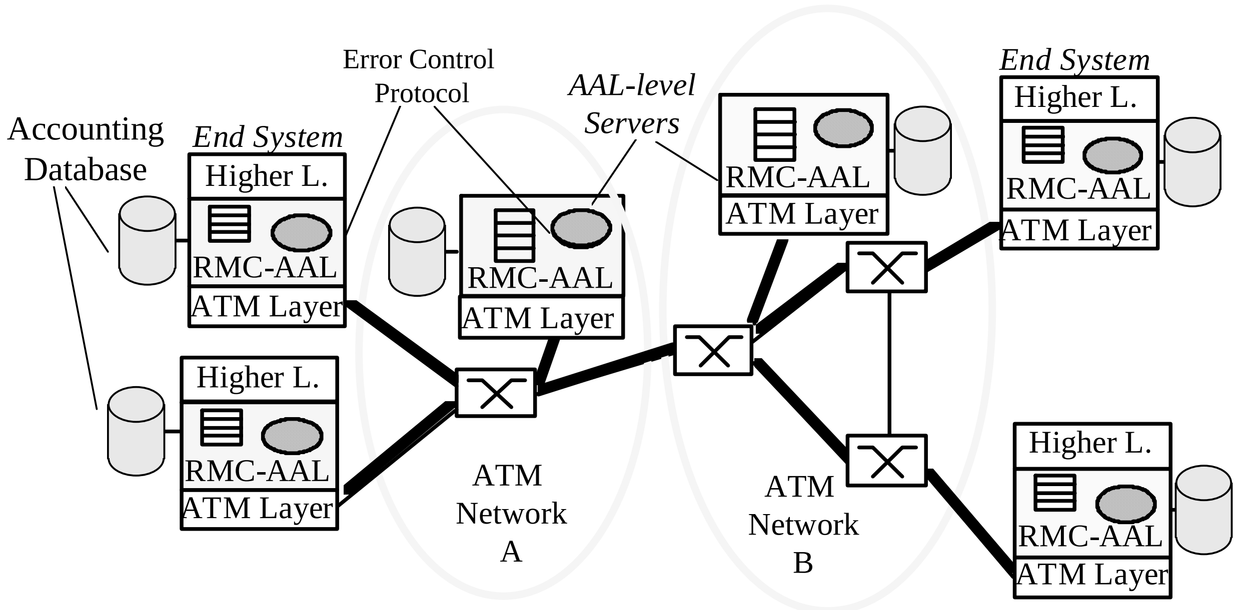

Figure 1: Accounting of user goodput

However, in case of highly bursty traffic, this approach allows only a low multiplexing gain and, therefore, leads to relatively high costs.

Our new approach is intended for best-effort services that allow to achieve a high multiplexing gain even for highly bursty traffic, providing a high-performance service at relatively low costs. Our approach aims at billing based on user goodput. This can be realised by counting of cells per connection that are transmitted the first time and distinguishing them from retransmitted cells. Alternatively, user goodput can be evaluated more accurately based on the number of user bytes by exploitation of control information of the error control protocol.

Implementation of high-performance accounting management

An implementation of this approach has to consider two main requirements. One is the easy access to collected data from management software, the other is the capability for high-performance processing needed for counting on cell or frame level. The appropriate component can be located at the customers network interface (customer or provider site) or, alternatively, within the public network for performance optimisation.

For management purposes we started to define an appropriate SNMP management information base (MIB). The following small example from this MIB describes a counter for cells with user data which are transmitted the first time:

```
fairATMBillingCellCounter OBJECT-TYPE
SYNTAX Integer64
MAX-ACCESS read-only
STATUS current
DESCRIPTION "Cells, which are transmitted for the first time"
::= { fairATMBillingCounters 1}
```

High-performance Processing of Accounting Management

The implementation of the hardware-component that collects the appropriate values for the MIB is done using the modular hardware-architecture CHIMPSY, which was especially designed for high-performance communication processing [ScZi95, Schi95]. Our concept for hardware-supported accounting management uses a dedicated processing component, the Management Processing Unit (MPU), see Figure 2. This component will be an additional element of our hardware library, which provides RISC-kernels, timers, and other dedicated components for communication. Using these components, we developed a hardware architecture called the Generic ATM Protocol Processing Unit (GAPPU) [CaSc95], see Figure 3. Attached to the central crossbar of the GAPPU, the management processing unit listens to the cell stream and updates data structures containing per connection cell counters. A new counter is initialised at connection setup. The component can now perform operations like increment, read, and delete the counter. It provides primitive services to the management proxy agent which is executed by a host CPU. Communication with the management agent is performed using the following MPU functions:

mpu_set (connection_id, parameter_id, value)
mpu_get (connection_id, parameter_id)
mpu_trap (connection_id, parameter_id, value)

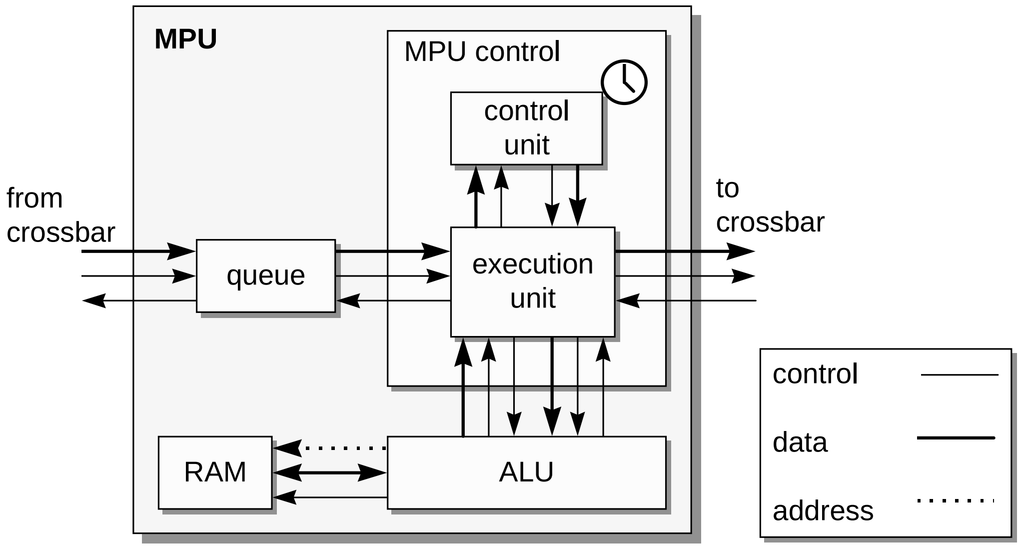

Figure 2: Management Processing Unit (MPU)

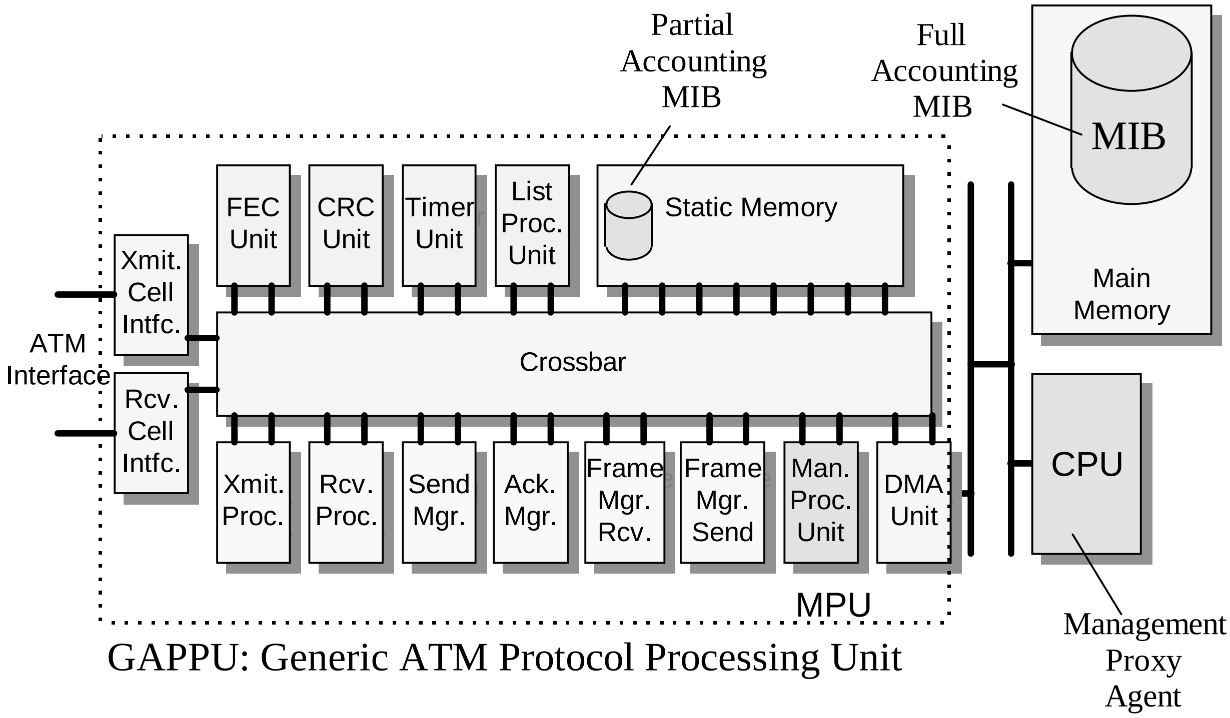

Figure 3: Hierarchical Accounting Management with the GAPPU

Conclusions

Main benefits of our approach are the possibilities for users to control traffic, and for the providers to perform accurate and fair billing. Furthermore, network providers can easily distinguish between retransmissions caused by errors inside the public network and errors caused by end-systems (buffer overflow etc.), by simply comparing the incoming cell counts of the network interface at the sender side with the outgoing cell count at the receiver side. This first step towards fair billing allows the provision of high-performance best-effort services which are more cost-effective than traditional, conservative services that provide quality of service guarantees.

References

[Bie93] Biersack, E.: *Performance Evaluation of Forward Error Correction in an ATM Environment,* IEEE Journal on Selected Areas in Communication, Volume 11, Number 4, pp. 631-640, May 1993

[CaSc95] Carle, G.; Schiller, J.: *Enabling High Bandwidth Applications by High-Performance Multicast Transfer Protocol Processing*, 6th IFIPConference on Performance of Computer Networks, PCN95, Istanbul, Turkey, Oct. 1995

[CaZi95] Carle, G.; Zitterbart, M.: *ATM Adaptation Layer and Group Communication Servers for High-Performance Multipoint Services*, 7th IEEE Workshop on Local and Metropolitan Area Networks, Marathon, Florida, Mar. 1995

[EdMV95] Edell, R.; McKeown, N.; Varaiya, P.: *Billing Users and Pricing for TCP,* IEEE Journal on Selected Areas in Communications, Volume 13, Number 7, pp. 1162-1162, Sept. 1995

[HeRo95] Heinrichs, B.; Rossmann, M.*: ABR: The ATM Service for Data Communication Applications?,* First Workshop on ATM Traffic Management (WATM‘95), Paris, France, Dec. 1995

[KiFa95] Kim, H.; Farber, D.: *New Congestion Control Framework for Large Bandwidth-delay Product Networks,* Proceedings of 6th International Conference on High Performance Networking (HPN'95), Palma de Mallorca, Spain, Sept. 1995

[KrKS93] Krishnakumar, A.S.; Kneuer, J.G.; Shaw, A.J.: *HIPOD: An Architecture for High-Speed Protocol Implementations,* in: Danthine, A.; Spaniol, O. (Eds.): High Performance Networking, IV, IFIP, North-Holland, pp. 383-396, 1993

[MiHR91] Mills, C.; Hirsch, D.; Ruth, G.: *Internet Accunting: Background,* Internet Engineering Task Force, RFC 1272, Nov. 1991

[Schi95] Schiller, J.: *A Flexible Co-Processor for High-Performance Communication Support,* IEEE Globecom'95, Singapore, Nov. 1995

[ScZi95] Schiller, J.; Zitterbart, M.: *Modular VLSI Implementation Architecture for High-Performance Communication Support,* 7th IEEE Workshop on Local and Metropolitan Area Networks, Marathon, Florida, Mar. 1995

[Stee94] Steenkiste, P.: *A Systematic Approach to Host Interface Design for High-Speed Networks*, IEEE Computer, pp. 47-57, Mar. 1994

[WiCh91] Williamson, C.; Cheriton, D.: *Loss-Load Curves: Support for Rate-Based Congestion Control in High-Speed Datagram Networks*, Proceedings of ACM SIGCOMM 1991, Zurich, Switzerland, pp. 17-28, Sept. 1991