C

BIBLIOTHEQUE

SCP

CERN DRDC
93-55

EUROPEAN ORGANIZATION FOR NUCLEAR RESEARCH

CERN LIBRARIES, GENEVA

SC00000108

CERN / DRDC / 93-55
RD-31 Status Report
22 December 1993

RD-31
Status report

NEBULAS: A high performance data-driven event building architecture based on an asynchronous self-routing packet-switching network

J. Christiansen, J-P. Dufey[1], M. Letheren[1], I. Mandjavidze[2], A. Marchioro, C. Paillard
CERN, Geneva

K. Agehed, A. Eide[3], S. Hultberg, T. Lazrak, Th. Lindblad, C. Lindsey, H. Tenhunen
The Royal Institute of Technology, Stockholm

L. Gustafsson
Institute of Radiation Sciences, University of Uppsala, Uppsala

---

M. De Prycker, B. Pauwels, G. Petit, H. Verhille
Alcatel Bell Telephone, Antwerp

M. Benard[4] D. Dack[5], S. Wright[5]
Hewlett Packard

[1] Joint spokespersons.
[2] At CERN on leave from the Institute of Physics of the Georgian Academy of Science, Tbilisi.
[3] Part time at Royal Institute of Technology on leave from Ostfold College, Halden, Norway.
[4] Research Grants Programme, HP European HQ, Geneva.
[5] HP Bristol Labs.

Table of Contents

EXECUTIVE SUMMARY

The RD-31 project aims at evaluating a new, parallel, data-driven approach to data acquisition at high rate experiments. It is based on the use of standard asynchronous transfer mode (ATM) packet-switching technology, which holds the promise of becoming a "universal" communications technology, unifying the telecommunications and local area network markets on the time-scale of the LHC.

Those aspects of the ATM technology that are relevant to its use for event building are now fully standardized by the International Telecommunications Union and the industry ATM Forum (summarized in section 2). The first ATM products are appearing on the market. However, a number of aspects relevant to the LAN and telecoms applications are not yet completely standardized, and it is likely that this will retard a significant penetration of the market by ATM. Nevertheless, all major telecom and computing companies are engaged in an unprecedented effort to produce useful standards, and it is expected that these standards will stabilize during 1994. Therefore it is likely that, from 1995 onwards, the prices of ATM switching hardware and workstation interfaces will begin to drop.

The work in RD-31 at this stage is limited to understanding the architectural issues, and price/performance/complexity trade-offs of various options for constructing event builders based on packet-switching fabrics. The main tool used for this is the simulation of the performance of switch and data acquisition architectures. We are currently developing suitable building blocks that could later be used to construct an ATM event builder. In order to promote interoperability, we base our developments on standards wherever possible.

Our modeling work has demonstrated that when ATM switching fabrics are used for event building, it is important to control the congestion that occurs within the fabric when data streams from multiple sources converge towards the destination. From the point of view of congestion control, there are two fundamentally different switching fabric architectures. A pseudo non-blocking, low-latency switch architecture is adopted for telecoms applications (see section 4.1). This uses no internal flow control, and when congestion occurs data are discarded. The traffic has to be "shaped" at the edges of the switch fabric in such a way that congestion is minimized and the probability of data loss becomes acceptably small.

We have simulated several traffic shaping schemes that could be used to implement a virtually lossless event builder based on a telecom ATM switch (sections 5.1 and 5.2). We have shown that when traffic shaping is applied, the switch can be scaled to the large dimensions required at the LHC experiments without loss of throughput. Furthermore, in the data acquisition system design, the achievable load on the switch (and hence its size and cost) can be traded against the amount of buffer memory in the system (and hence the event building latency) and the complexity of the real-time software environment in the destinations.

We have also studied the performance of lossless event builders based on switching fabrics incorporating internal flow control (sections 4.2 and 5.3). These perform well for small switch sizes, but their performance does not scale to large dimensions as well as that of the telecom switch used with traffic shaping. We propose to continue these studies to investigate whether the combination of a flow-controlled switch with traffic shaping will result in a lossless event builder with good scaling characteristics.

A small multi-path self-routing switching fabric is on the point of being delivered by our collaborator Alcatel Bell Telephone (Antwerp). Another industrial partner, Hewlett Packard, has donated workstations and broadband test equipment to the project, which will allow us to develop, test and diagnose ATM hardware. We are currently designing a VME ATM interface module that incorporates hardware to perform the special traffic shaping required for event building (section 6.1). This development is hardware and software compatible with the scalable data acquisition software developments of RD-13.

During the second year of the project we aim to complete these developments and to integrate the various parts into a small demonstrator system (section 6), which will then be used to experiment with various data acquisition protocols (section 5.4) and to make performance measurements.

1. INTRODUCTION

In the LHC experiments a variety of data acquisition and triggering architectures have been proposed [1,2] for use downstream of the first-level trigger. Depending on the architecture, the aggregate bandwidths required for event building are expected to be of the order of at least 10-100 Gbit/s.

These high bandwidths cannot be handled by the traditional bus-based data acquisition systems. A high-performance, cost-effective and expandable event builder can be realized with a parallel switching fabric providing N x N connectivity with of the order N.logN hardware complexity. RD-31 has the goal of evaluating the feasibility of using asynchronous transfer mode (ATM) switching fabrics for this purpose.

The International Telecommunication Union's standardization body, the ITU-TSS (formerly known as the CCITT), has recommended the use of ATM as the switching technology for the future broadband integrated services digital network (B-ISDN). The ITU's B-ISDN standards [3] were originally targeted at telecommunications and wide area networking (WAN) applications.

However, ATM technology is now also being adopted for high-performance local area networking (LAN) applications, and all major workstation companies are actively engaged in developing the technology (typical activities are the development of LAN hubs based on ATM switches, ATM interfaces to workstations, and the implementation of the internet TCP/IP protocol over ATM). Efforts in this area are coordinated by an industry association, the ATM Forum [4], which parallels the ITU's standardization efforts, while focussing on the needs of the (more cost-competitive) workstation/LAN industry.

It therefore appears likely that ATM technology will dominate both high-performance WAN and LAN networking throughout the time-span of experiments at the LHC. ATM products will most likely first appear in the telecommunications market, and only later, when all aspects of the standards are worked out and inter-working between products from different vendors has been demonstrated, will they start to sell in significant quantities. While the market is restricted to the medium-volume, traditionally high-priced telecommunications sector (where equipment is amortized over typical life cycles of 20 years or more), prices will probably remain too high for large scale application in a physics experiment. However, the growth of multi-media applications and the adoption of ATM by the more cost-competitive LAN industry suppliers, together with a policy of fibre-to-the-curb, are expected to render ATM affordable on the time scale of the LHC. It was with these predictions in mind that the RD-31 project [5] was proposed.

This document summarizes the work carried out by the collaboration since approval of the proposal by the Research Board on 26 November 1992. We recall here the milestones set by the DRDC for the first year of the project:

- design and simulate architectures and protocols for an ATM-based event builder system;
- design an interface from front-end buffers to an ATM switch, including destination routing and flow control;
- prepare a VME workstation-based test bench for evaluating commercial ATM switches.

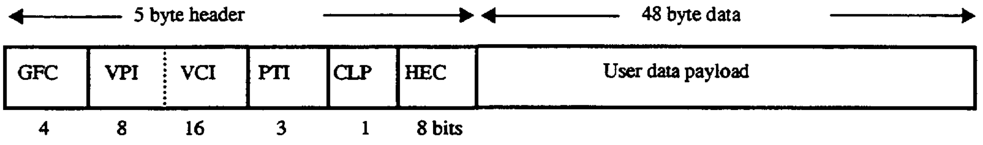
It should be noted that our proposal originally included work on asynchronous digital pipeline architectures with data time-stamping, embedded signal processing, and level-one trigger-filtering functions. This work was seen by us as being naturally linked to the design of the interface between the front-end electronics and the event-building switching fabric. However, the DRDC decided to focus the project onto the essential ATM switching and interfacing aspects by excluding the work on pipelines from the milestones. Therefore, although, this work has been continued outside of the framework of RD-31, our collaborators in this domain (Tampere University of Technology and LIP) are not signatories to this status report. On the other hand, the Institute of Radiation Sciences at the University of Uppsala has now joined into the effort. In addition, in June 1992, the Manne Siegbahn Institute of Physics merged with the Royal Institute of Technology. Some further changes, in individual collaborators, are reflected by the updated list of signatures on the cover page.

2. OVERVIEW OF ATM SWITCHING

For completeness we first briefly recall the basic principle of ATM technology and the standard ATM protocols.

2.1 The Principle of ATM

ATM is a connection-oriented technology in which data are segmented into short fixed-length cells, each of which carries 48 bytes of data and a 5-byte header as indicated in fig. 1.

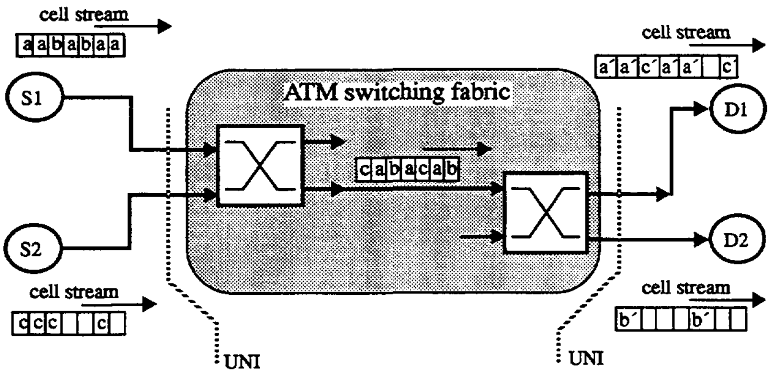


GFC = Generic flow control
VPI = Virtual path identifier
VCI = Virtual channel identifier

PTI = Payload type indicator
CLP = Cell loss Priority
HEC = Header error control

Fig. 1 The format of the ATM cell.

A 24-bit label in the header identifies the logical connection to which the cell belongs. The label information consists of a 16-bit virtual channel identifier (VCI) and an 8-bit virtual path identifier (VPI) which are used by a network of switching elements to route the cell to its destination.



Connection. Source	Destination	VCI label at source	VCI label at destination
S1	D1	a	a´
S1	D2	b	b´
S2	D1	c	c´

Fig. 2 The principle of the Asynchronous Transfer Mode.

Cells belonging to many different virtual connections can interleave in a cell stream transported over a physical link. This asynchronous multiplexing and switching technique, indicated in fig. 2, is fundamentally different from the circuit-switched technique. ATM has been chosen for the B-ISDN because it allows a single architecture to efficiently support virtual connections carrying traffic at widely different bandwidths, and to handle bursty traffic as expected in some multi-media applications. In addition it allows the telecommunications operators to multiplex their existing standard "tributaries" on the new high bandwidth B-ISDN infrastructure.

Virtual connections can be established dynamically by the communicating partners themselves via a signalling protocol. This process creates an entry in a table at the user network interface (UNI) that is used to map the cell header's VCI[1] label into the internal (manufacturer dependent) information necessary to route the cell to the appropriate physical output port of the switching fabric.

Alternatively, the ATM network's management system can directly assign and de-assign semi-permanent virtual connections by manipulating the VCI mapping tables (a so called "cross connect" configuration).

At the UNI, the VCI field of the input ATM cell is also mapped into a new value that is placed into the header of the ATM cell as it leaves the network. Thus, source and destination are not constrained to use the same VCI value to identify a given virtual connection, and a VCI label does not have to be unique network-wide.

The disadvantage of ATM compared to circuit-switching technologies is the requirement for connection admission control and bandwidth policing in order to manage network resources and control congestion. Network admission decisions can only be made on a statistical basis.

2.2 The B-ISDN protocols

ATM is a rapidly evolving technology and, although not all aspects required for its widespread acceptance for computer networking applications are fully worked out and demonstrated, the essential aspects needed to construct an ATM event builder, are now standardized. The ITU's B-ISDN standards [3] define a 3 layered protocol, which we will briefly describe below. Here we remark that the ATM Forum, while adopting the basic ITU standards, has added its own alternative standards designed to address the needs of the LAN / workstation industry and facilitate the early introduction of cost-effective products based on existing FDDI technology. This parallel standardization effort has complicated the task of interworking between products complying to alternative standards.

2.2.1 The Physical Layer

The physical layer handles the transport of valid ATM cells. It defines the transport medium (copper or optical fibre), data framing standards, access rates (155.520 Mbit/s, 622.080 Mbit/s, 2.48832 Gbit/s etc.), cell synchronization algorithms, and error detection and correction procedures for the cell header information. The ATM Forum has defined alternative standard bit rates (44.736 and 100 Mbit/s) and a low cost physical medium (multi-mode fibre with LED transmitters) for short distances. This does not directly interwork with the ITU's physical medium standard (mono-mode fibre and laser transmitters) for long-haul connections.

2.2.2 The ATM Layer

The ATM layer handles the switching of cells from source to destination according to the virtual path and virtual channel identifiers (VPI/VCI) carried in the cell headers. The differentiation of the routing label into separate VPI and VCI fields is intended to facilitate the grouping of a number of virtual channels that are to be switched together to follow a common virtual path through the network.

The ATM layer also involves signalling protocols for the establishment of connections through the switch, traffic flow control functions (input traffic policing, discarding of cells in case of internal buffer overflows), and quality of service (bounds on the end-to-end delay and jitter). An important point to note is that, on each virtual connection, the ATM layer guarantees the delivery of cells in the correct sequence (even if the individual cells follow different paths through the network to the destination).

2.2.3 The ATM Adaptation Layer

The ATM adaptation layer (AAL) consists of two sub-layers which hide the complexity of the ATM layer from the user. The segmentation and reassembly(SAR) sub-layer segments packets (i.e. messages) received from the application into cells at the source, and reassembles them at the destination before passing them to

[1] Throughout the remainder of the report we use the short hand VCI to refer to the combined VPI/VCI 24-bit label.

the application. The convergence sub-layer (CS) performs multiplexing and de-multiplexing of concurrently received/transmitted messages from/to different sources/destinations. In addition it handles cell loss detection and cyclic redundancy check (CRC) generation and checking of the complete packet (including the user data).

A number of different AAL protocols have been proposed to adapt the underlying layers to the requirements of different classes of applications. The AAL protocol proposed by the ATM Forum for data transport applications is the so called AAL5 protocol [6]. This protocol is very popular because of its relative simplicity; it has now been adopted for standardization by the ITU.

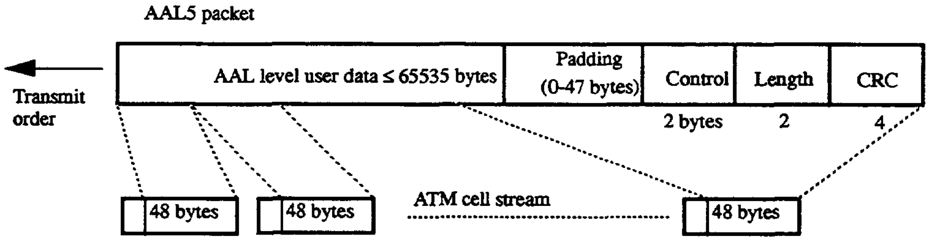


Fig. 3 AAL5 packet segmentation onto the ATM layer.

Figure 3 shows how an AAL5 packet, which carries up to 64 kByte of user data and is terminated by an eight byte trailer, is segmented on to a stream of cells at the ATM layer. The end cell of the AAL5 packet is identified by an ATM cell with the 3-bit PTI field set to the value 0X1. At the destination, the de-multiplexing of cells belonging to different virtual connections is performed on the basis of the VPI and VCI fields of the cell headers.

The RD-31 project will use the standard AAL5 protocol because it has now been adopted by all workstation companies, is implemented in commercial ATM protocol chip sets, and because it fulfils many of the basic requirements of a higher-level DAQ protocol.

3. THE PRINCIPLE OF EVENT BUILDING USING AN ATM CROSS CONNECT

Figure 4 shows the use of an ATM switching fabric configured as a cross-connect to provide the N x M semi-permanent virtual connections needed to perform event-building from N sources (S) into a farm of M processors (P). In the cross connect configuration the virtual connections are not established dynamically, and therefore the complexities of signalling and network admission control are avoided. It is important to note that, by using the AAL5 protocol, the complexities of the ATM layer are hidden from the user - he effectively sees only a web of virtual connections on which the lowest-level transaction is the transmission of a block of data (in AAL5 format).

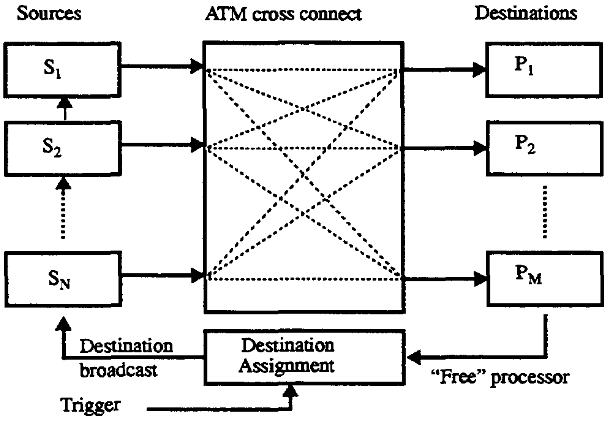


Fig. 4 Event Building with an ATM cross connect.

Each source uses M different VCI labels to identify its virtual connections to the M destinations. After each trigger, the destination assignment logic generates and broadcasts the information used by the sources to select the appropriate virtual connection for the event. Software or firmware in each of the sources then passes a descriptor of the event's local data block to the AAL5 protocol chip in the source interface. The protocol chip then collects the data block from data memory, segments it into cells, inserts the appropriate VCI label in the cell headers, and drives the cells into the switching fabric. As mentioned before, the switch maps the VCI labels into new values which are used by the AAL5 protocol chip in the destination to identify from which of the N sources the cell was sent, and thus to assign the cell to the appropriate data block reassembly buffer.

In the event building application, the many-to-one traffic flow and the burstiness of traffic associated with the trigger are very different from the random telecommunications traffic patterns for which ATM switches are engineered. We next consider a question of critical importance in the use of ATM switching fabrics; namely how to handle traffic congestion. Then we discuss more specifically the handling of traffic congestion in an event builder based on an ATM switching fabric.

3.1 Blocking and congestion control by traffic shaping in ATM Networks

ATM's statistical multiplexing of traffic over the hardware resources of the switching fabric can result in the aggregate cell throughput demanded from a physical link temporarily exceeding the available bandwidth; therefore an adequately sized cell queueing function must be incorporated in each switching element. These buffers are dimensioned such that the probability of overflow is acceptably small for the traffic patterns expected in telecommunications applications, normally characterized by a random (Bernoulli) distribution for the cell injection times and random cell destinations. In the typical ATM switch designed for telecommunications applications, cells are discarded when a buffer overflow occurs. The higher layers of the protocol stack can either ignore the lost cells (e.g. in a video or voice service) or take action to retransmit them or the entire packet (e.g. in a secure data transfer service).

When the switching fabric is used in an application in which the traffic patterns on different virtual connections are correlated, the probability of buffer overflow (congestion) is increased. When the application's traffic patterns would lead to congestion it becomes necessary to "shape" the traffic at the inputs of the switching fabric. Shaping can consist of reducing the admissible load, or of smoothing very bursty traffic. Each subscriber negotiates admission to the network and is authorized to use no more than an agreed peak and average bandwidth. A policing function enforces the agreed traffic rates for each VCI connection at the network input port. Nevertheless, because traffic shaping is enforced only at the edges of the network, and because cells incur variable delays within the network, there is still a finite probability that buffer overflow occurs.

The probabilistic approach to the security of data transfer over ATM that characterizes the telecommunications equipment vendors stems from their past history in handling mostly voice communications over long-distance, unreliable lines. Secure data transfer forms only a small part of the expected future B-ISDN traffic, and therefore has not dominated the criteria driving the design of the hardware. Of course, secure data transfer over an unreliable medium can be obtained by overlaying with a higher level protocol to detect and recover from cell losses. Additional reasons for the design strategy adopted for telecommunications ATM switches are the difficulty of implementing a window flow-control mechanism at high rates over the (possibly) long distances involved in WAN, and the desire for an essentially non-blocking architecture, in which the quality of service (e.g. delay and jitter) offered to each subscriber is unaffected by the traffic generated by other subscribers. Such considerations are important for constant bit-rate (uncompressed), real-time video services for example.

3.2 Flow-controlled switching fabrics

An alternative technique to control congestion and avoid cell losses is the use of a flow control protocol. Such a protocol may be implemented either in hardware at the level of the link between two communicating switching elements embedded inside the switch fabric itself, or, at a higher level, in the layered ATM protocol between source and destination. The standard ATM protocols define the so called forward explicit congestion notification mechanism, whereby cells passing a congested region of the switch fabric are tagged and carry notification of the congestion to the destination user. The higher layers of the protocol should then take action to reduce congestion, for example by reducing the data rate available to the virtual connection. Telecommunications switching fabrics are likely to implement the high level flow control, if any. However this high-level flow control technique suffers from a slow reaction time that renders it unsuitable for our event building application.

Although ATM is regarded as an attractive technology by the computer networking vendors, the telecommunications concept of traffic shaping and the use of an intrinsically unreliable communication medium overlaid with a higher-level protocol to provide reliable communication is foreign to them. In addition, the basic assumption made by telecom switch designers concerning the randomness of aggregated subscriber traffic probably does not hold in the case of LAN traffic [7]. For this reason, several research projects are investigating the implementation of flow control at the hardware level [8], [9]. In the most obvious approach [8], a switching element can assert a 'back-pressure' on an upstream element in order to block the transmission of further cells when its internal buffer memory is full.

3.3 Congestion control in ATM event builders

As we mentioned before, the many-to-one traffic patterns associated with event building clearly will lead to chronic congestion and cell loss in an ATM-based event builder unless steps are taken to avoid it. In one approach, the DAQ system design can incorporate traffic shaping in the form of bandwidth control at the sources and/or control of the relative cell injection times at the different sources.

On the other hand, link-level flow-controlled ATM switch architectures, should they appear on the market, would offer an alternative building block for constructing lossless ATM-based data acquisition systems. Such switches would, in principle, remove the need to perform traffic shaping at the sources. Back-pressure signals would be issued constantly as the concentration of the event's traffic in the switching fabric creates congestion near the outlets, and the blocking of the internal links could spread backwards as far as the data sources in principle. This blocking will have an effect on the latency of cell transport through the switch and on the effective aggregate throughput that is difficult to predict analytically, but can be evaluated by simulation.

In section 5 we present three possible traffic shaping strategies and compare traffic shaping with the flow-controlled approach.

4. SWITCHING FABRIC ARCHITECTURES

We have developed simulation models to investigate the design and performance of DAQ systems based on these two fundamentally different ATM switch architectures. For the traffic shaping approach we have modeled systems architectures using a telecommunications ATM switching architecture developed by Alcatel [10]. For the flow-controlled approach we used an experimental switching fabric architecture developed jointly by AT&T and the Ecole Polytechnique Fédérale de Lausanne [8].

4.1 The Alcatel MPSR Switching Fabric

This section describes a multi-path self-routing (MPSR) broadband switching fabric developed by Alcatel for public network applications. It has a number of properties that make it attractive for use as an event builder:

- it has an architecture that allows expansion up to and beyond the 100 Gbit/s aggregate bandwidth required for event-building in experiments projected at the LHC.
- it provides the essential operations and management facilities required in a large system, e.g. real-time self-test, automatic isolation and bypassing of faulty hardware modules, fault-tolerance, graceful performance degradation, diagnostic software etc.
- it is designed for at least a 10-15 year life-cycle which matches with the expected life-time of future large scale experimental facilities.
- an evolution path to higher performance is foreseen.

Because it is designed for telecommunications applications, the Alcatel switch has no internal flow-control on the links between switching elements. Its use for event building will require traffic shaping to be applied at the sources. Some appropriate traffic shaping strategies and their implications on the data acquisition system design are discussed in Section 5.2.

4.1.1 Principles of the MPSR switching fabric architecture

A detailed description of the architecture is given in [10], but the major characteristics are summarized as follows:

Self-routing: Each individual cell is independently routed through the network according to the routing information in its header. The routing algorithm is embedded in the hardware and distributed over the switching elements of the fabric. In the cross-connect configuration, all required virtual connections are predefined, and therefore there is no overhead penalty associated with the set up of a routing path before each data block is sent.

Asynchronous multi-slotted internal transfer mode: Each 53 byte ATM cell is mapped internally into a 68-byte multi-slot cell (MSC). The MSC is a train of eight 68-bit slots which are always routed via the same path and therefore remain in sequence. Because the switch operates internally on the smaller 68 bit slots, the required buffer memory size in the switching elements is reduced and higher performance is achieved by trading memory size against memory speed. In addition, lower latency is achieved because the leading slots of an MSC can already be sent out before the trailing slots are received ("worm hole" routing).

Internal link under-loading. The ratio of the number of 155 Mbit/s internal paths to input ports is 2, and the translation of the ATM cells into MSCs leads to a data expansion factor of 1.283. As a result the average traffic on paths inside the switch uses only 51% of their bandwidth when the external links are all loaded at 80%. The internal under-loading strongly reduces the probability of congestion and consequent cell loss.

Multi-path: The first stages of the switching fabric do not route cells, but instead they randomly distribute the traffic over alternative internal paths. This has the effect of distributing the data flow evenly over the switch resources and making the performance of the switch less sensitive to the mix of traffic. The multi-path architecture also provides fault-tolerance; modules are self-testing and faults are automatically isolated. In the presence of a fault a "back-pressure" signal is propagated upstream and is used to divert the traffic over the remaining alternative paths, resulting in a graceful degradation of performance. Finally the multi-path architecture allows bandwidth scaling because interfaces to external links running at higher bit rates can be accommodated by distributing the traffic over more internal paths.

Resequencing: The non-deterministic delay of cells routed over the different internal paths means that cell sequencing within a virtual connection might not be preserved, hence a re-sequencing function is provided at the outlet of the switch before the cells enter the output queue. This operates by time stamping cells at the inlet as they enter the switch and, before releasing them to the output queue, holding them in a resequencing buffer until they have spent a predetermined total time in the switch. This constant delay automatically resequences the cells and minimizes jitter. Cells that are not delivered to the resequencing buffers within this fixed delay budget are lost. Therefore, the delay budget is chosen such that the probability of a cell spending more time in the switch fabric is an order of magnitude lower than the probability of loosing a cell due to congestion in the switch fabric itself.

4.1.2 Implementation of the MPSR switching fabric architecture

The switching fabric is a folded, multi-plane, multi-stage structure built from switching modules (SM) that each switch 64 x 64 links running at 155 Mbit/s. In its largest configuration, shown in figure 5, the fabric will consist of an eight-plane, three-stage structure (AS, PS1 and PS2 stages of switching modules). It supports up to 2048 external 155 Mbit/s links or up to 512 external 622 Mbit/s links (mixed link rate configurations are possible). Future versions will be expandable to 16 384 external 155 Mbit/s links.

Traffic switching units consisting of link termination (TLK) boards and access switches are used to interface full duplex external links, running at 155 or 622 Mbit/s, to the switching planes. Modules are interconnected by 622 Mbit/s "quad" links that each multiplex the traffic of a group of four 155 Mbit/s links. The fabric can be expanded without rewiring by inserting additional modules as required.

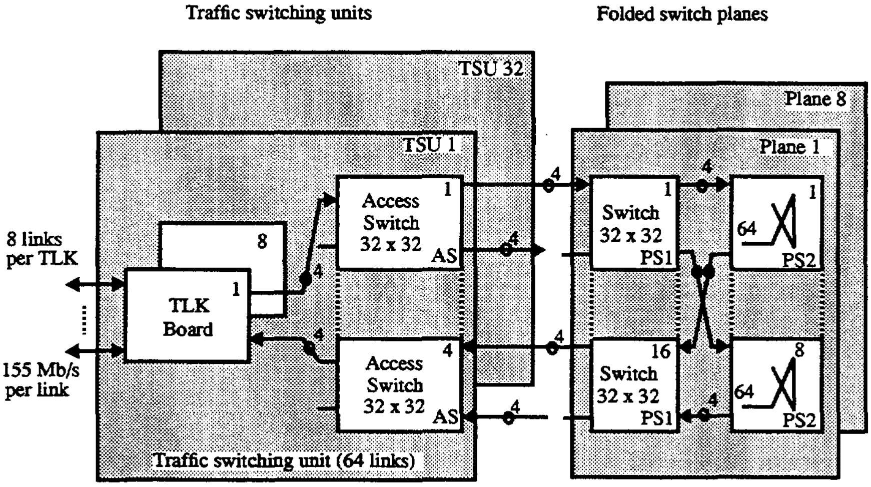
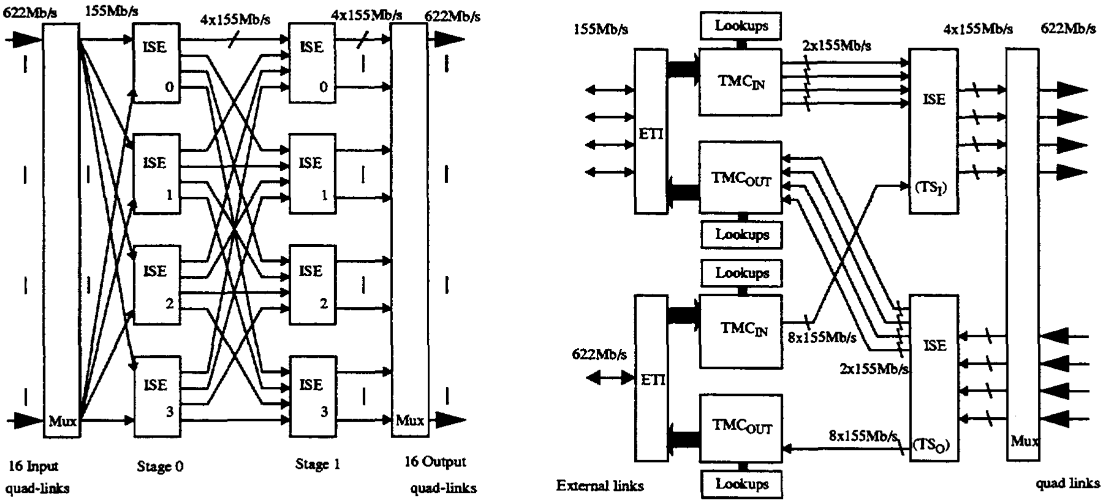


Fig. 5 The architecture of the MPSR switching fabric.

4.1.2.1 The switching module

The switching module (SM) board, shown in figure 6(a), is equivalent to a single-stage 64 x 64 switching element. It is built from two stages of eight identical 16 x 16 integrated switching element (ISE) circuits. Groups of four 155 Mbit/s links are multiplexed to form 622 Mbit/s quad-links for interconnection of SM boards. An on-board test processor (not shown) has input and output access to each ISE chip so that it can continuously check for correct operation of the switch module board.



(a) the 64 x 64 switching module

(b) the link termination board

Fig. 6 The basic building blocks of the MPSR switching fabric

4.1.2.2 The link termination board

On the TLK board, shown in figure 6(b), the external transmission interface (ETI) terminates the physical transmission medium and performs the framing of ATM cells in to and out of the SDH / SONET [11] payload enveloppe.The ETIs can be chosen to drive either a set of eight external links at 155Mbit/s, or two external links at 622Mbit/s. The transmission mode conversion chips (TMC) convert ATM cells to/ from the internal multi-slot cell format and expand to twice the number of internal links (to ensure internal link underloading). In the input direction the TMC_{in} chips also perform time stamping and traffic policing (using the leaky bucket algorithm [12]). In the output direction the TMC_{out} chips use the time stamps to resequence cells as described before. The uniform traffic distribution to / routing from the four access switches is performed by two ISEs, and bit synchronization and multiplexing is performed by two Mux chips. An onboard maintenance controller (not shown) performs on-line testing.

4.1.2.3 The integrated switching element

The switching and path randomization functions are performed by an integrated switching element (ISE) that interconnects 16 serial inlets to 16 serial outlets, each operating at 155 Mbit/s. The ISE uses a shared buffer memory architecture shown in figure 7. It can switch MSCs from any inlet to any one outlet, or it can broadcast or selectively multicast MSCs to two or more outlets. As mentioned above, the ISE operates at the slot level, permitting "worm hole" routing of MSCs. The input and output ports perform serial to parallel conversion and vice versa.

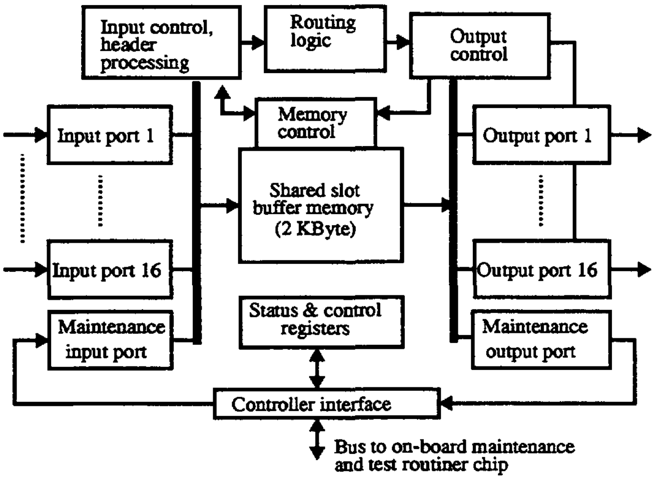


Fig. 7 The architecture of the ISE.

The routing logic interprets the routing data carried by the MSC in three different ways, depending on routing mode parameters loaded in ISE at initialization time. The internal 155 Mbit/s links are normally associated into groups of 4, which are carried between modules by the same 622 Mbit/s quad-link. The basic routing mode directs an MSC to a specific group of outlets and ensures both random distribution and even load balancing of traffic over all links of the group.

As a special case, when the group includes all 16 outlets, the MSCs are then randomly assigned to different quad-links. This mode is used by the ISEs in the traffic distribution stages of the network.

The two other routing modes are used for multi-casting MSCs to several groups of outlets and for directing an MSC to a specific outlet (for maintenance and test purposes). More details can be found in [13].

4.1.3 Evolution towards higher performance

The current configuration of the switching fabric is based on the 16 x 16 ISE, which is implemented in 0.8µ CMOS technology. The next version of the ISE will still be an 16 x 16 ISE, but with a bigger shared memory buffer, more multicast trees, and some other improvements. This ISE will be implemented in 0.5µ CMOS technology and will be available at the end of 1995.

The third version of ISE, forecast for the end of 1998, would be designed in 0.3µ CMOS and would have 32 inlets/outlets. In the target configuration the switching fabric will be built from switching modules interconnecting 128 input links with 128 output links. An upward compatible technology-tracking strategy has been implemented, whereby the first implementation is defined as a straight sub-equipment of the target configuration. If an extendable switch configuration is chosen, the switch can be upgraded while in operation by SM board substitution, without re-cabling or re-arrangement of the switch. In the target configuration a maximum capacity of 16K links at 155 Mbit/s or 4K links at 622 Mbit/s will be supported.

Improvements are also foreseen for TMC_{OUT} which will further reduce the probability of cell loss at a given load.

4.1.4 The model of the Alcatel MPSR switch

An extension of the object-oriented language C++ supporting concurrency [14] has been used to develop models of the ISE switching element, the 64 x 64 switching module (SM) and the link termination board (TLK). These building blocks have been used to assemble models of switching fabrics of up to 256 x 256 ports.

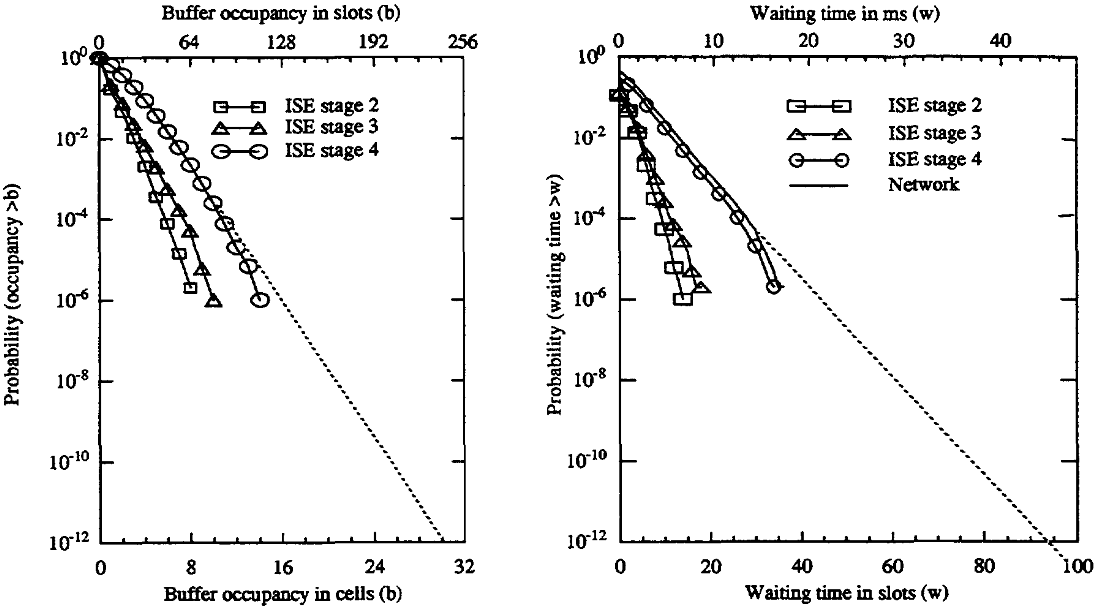


Fig. 8 ISE buffer occupancy and waiting time in a 128 x 128 switching fabric operated under a "telecommunications" traffic pattern at a load of 80%.

Figure 8 shows the tail distributions for shared buffer memory occupancy and cell "waiting times"[2] for ISE's positioned in various switching element stages of a 128 x 128 network driven by a "telecommunications" traffic pattern for the case where each external link is loaded to 80% of its nominal bandwidth. Very similar distributions derived by Alcatel engineers [15] have been used to dimension the shared buffer memory of the ISE and to engineer the maximum internal transfer delay used for resequencing the cell streams.

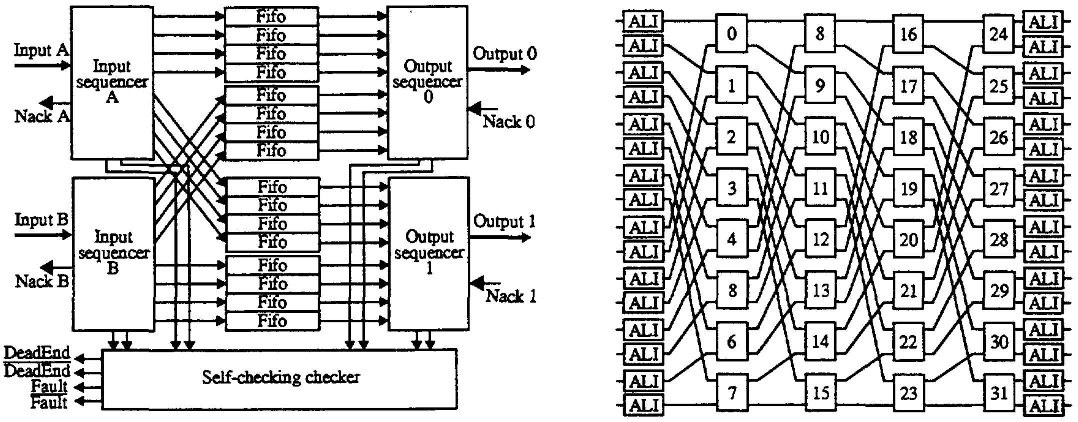
The probability of cell loss (due either to shared buffer memory overflow or cell delivery latency exceeding a maximum delay-equalization limit) is engineered to be less than 10^{-10} for the largest switching

[2] The difference between the observed delay and the delay expected when no congestion occurs within the switching fabric.

fabric operating under a "telecommunications" traffic pattern and a load of 80%. The ISE shared buffer memory is sized at 256 slots (2 kBytes) in order to achieve this level of performance.

4.2 Architecture of the Phoenix ATM switching fabric

For the investigation of the flow-controlled approach, the switch fabric that has been modelled is based on the Phoenix single chip binary cross-point switching element developed by a joint project between AT&T and the Ecole Polytechnique Fédérale de Lausanne (Switzerland) [8]. Figure 9(a) shows the block diagram. It features 4 priority levels, independent buffers (512 bytes each) on each priority level and each input and output, self-checking and error signalling, etc. Each input and output operates at 400 Mbit/s which provides a safety margin for implementing the 155 Mbit/s rate of B-ISDN. Flow control is realized by means of a back-pressure signal (Nack) that a switching element can assert on its immediate upstream partner in order to halt further transmission of cells on the link when the receive buffer is full. When this happens, the transmitting switching element can choose either to discard the data (as in the Alcatel switch) or keep it until the receive buffer can accept it.



(a)

(b)

Fig. 9 Architecture of the Phoenix switching element and a 16 x 16 ATM shuffle network.

The architecture of the full switching fabric, shown in figure 9(b) for the 16 x 16 configuration, is of the so called perfect shuffle network type. Every input and output line is interfaced to the outside world by means of an ALI (ATM layer interface) ASIC [16]. Apart from mapping the VCI labels of incoming cells into the internal routing data used by the switching elements, ALI matches the internal 400 Mbit/s bandwidth with the external 155 Mbit/s B-ISDN link rate. For that purpose it provides for output buffering. On input, a memory buffers the incoming traffic when back-pressure blocks the switching fabric's input port. This input buffering is provided externally to the ALI chip. Cell losses can only occur in the switching fabric if this input buffering is insufficient to absorb those statistical fluctuations where the back pressure temporarily propagates back as far as the switch fabric's input ports. The segmentation of the data into ATM cells and their reassembly are performed outside of this structure in the user network interface hardware.

The designers of the Phoenix switch have obtained very good results with bursty traffic when back-pressure flow control was in action. It must be pointed out however that this traffic was generated at the boundary of the switch fabric and that the higher internal bandwidth helped to quickly reabsorb the congestion when a blocking situation had disappeared. In the event building application, the concentration of traffic inside the switch results in a congested region not being easily reabsorbed because cells can continue to arrive at the congested region at the same (or even higher) rate as they leave it.

We have simulated this switch, using only one priority level and the flow control option (i.e. keeping the cells when back-pressure is asserted), and allowing the back pressure to propagate as far back as the input ports of the switching fabric. As for the case of the event builder based on the Alcatel switch, the model is coded in µC++. Our model has been successfully cross-checked with simulation results obtained by the switch designers.

5. MODELING OF ATM-BASED EVENT BUILDERS

5.1 Low latency event building with the Alcatel MPSR switching fabric

In certain data acquisition architectures low latency event building is required. An example of this is the local/global second-level trigger architecture (proposed in [1]) where only data from selected "regions-of interest" (RoI) are used for the triggering decision, and where local feature extraction is performed by dedicated feature extraction devices. Because of the RoI and feature extraction, a reduced volume of data is sent to the global decision processors. However, the second-level trigger latency is limited to at most a few milliseconds by the available second-level buffer memory, which is used to hold the raw data during the second-level trigger decision time.

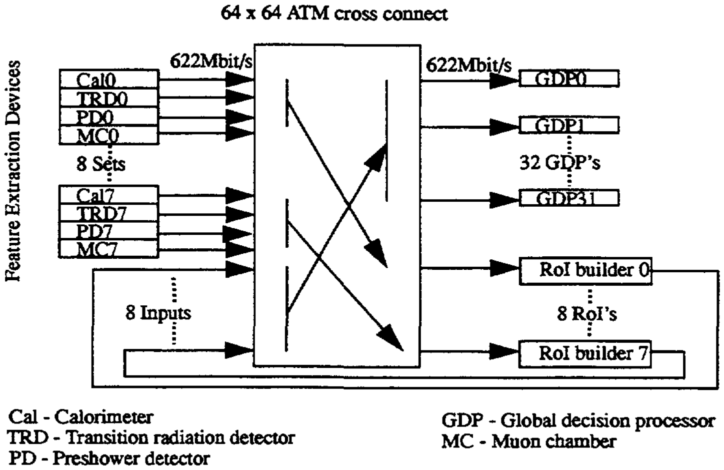


Fig. 10 Switching for RoI building and event building for the global 2nd level trigger.

In collaboration with RD-11 [17], we have used our model of the Alcatel MPSR switching fabric to build a model of the data acquisition architecture shown in figure 10. This performs RoI building and event building for the global second-level trigger. This operates at a first-level trigger rate of 100 kHz and, because of the small volume of data to be moved, the average load on the used external links is approximately 10%. At this load no cell loss was observed in a simulation run of 16 000 events (approximately 800 000 cells). The total latency, measured from the first-level trigger until the second-level trigger decision, is expected to be 600µs on average, of which data switching accounts for approximately 100µs.

5.2 Congestion control in an event builder based on a telecom ATM switch

We next consider event building in a data acquisition architecture where the aggregate bandwidth is so high that a very large switch is required (see for example [2]). Here it becomes important, for economic reasons, to efficiently utilize the available bandwidth. In order to be able to run a telecom ATM switching fabric (e.g. the Alcatel MPSR switch) with a high load, the traffic patterns must be shaped such that congestion is minimized. Traffic shaping involves input bandwidth control and/or control of the relative cell injection times at the different sources. The most efficient utilization of the switch resources occurs when many events are simultaneously built, and their individual traffic patterns are spread evenly over the switch.

5.2.1 Modelling an event builder based on the Alcatel switch

The performance of event builders based on the Alcatel MPSR switch has been simulated under a number of different generic traffic shaping schemes. In order to study the limits to the event builder performance, simulations were made in which it was assumed that event processing times in the destinations do not limit the rate at which events can be accepted. The inter-trigger delay used for generating the event building traffic followed an exponential distribution and, in order to emulate some degree of clustering of data in the detector, the local event fragment sizes generated in the sources also followed an exponential distribution.

The traffic patterns generated in these initial studies are certainly oversimplified. We expect "real" data to exhibit correlations between sources and a higher degree of "burstiness". For or this reason we have recently attempted to use typical "background" events generated by Monte Carlo and then filtered by the level-one trigger (see section 5.5).

5.2.2 Source traffic shaping scheme 1 - the event-based barrel-shifter

The first scheme studied, shown in figure 11, emulates a circuit-switched barrel-shifter, and is similar (but not identical) to the event-building schemes proposed under [18]. The switched circuits are virtual and they are allocated the full bandwidth of a 155 Mbit/s path. No rate control is applied at the sources, and each source module must transmit all of its data for a given event within a fixed time slot. For each event trigger a train of time slots is used to scan all sources starting from source 0. The train for an event is initiated by source 0 as soon as the event data is available and the previous time slot is finished. Parallel data transfer occurs between many source-destination pairs and many events are built simultaneously, but only one source at a time transmits data belonging to a given event.

This scheme requires time slot synchronization between the sources, and because of an uneven fill of the time-slots it suffers from a relatively low utilization of the available switching bandwidth.

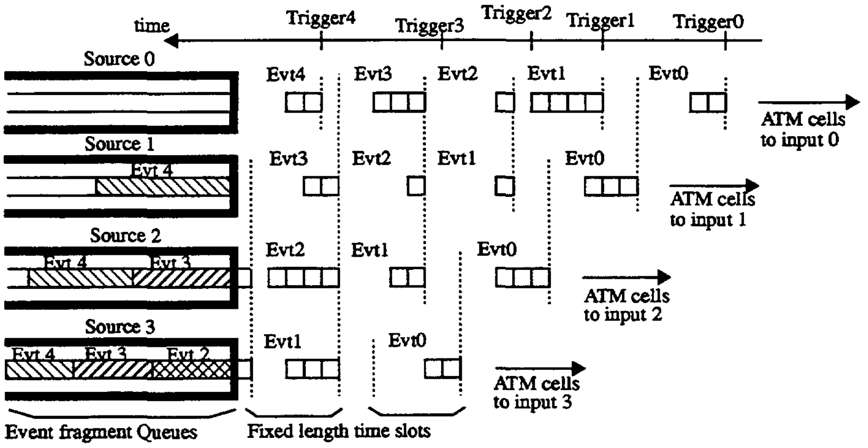


Fig. 11 Emulation of a circuit-switched barrel shifter scheme.

5.2.3 Source traffic shaping scheme 2 - the cell-based barrel-shifter

Figure 12 shows the second scheme studied. Rate control is used at the sources in order to limit the traffic on each virtual connection so that the aggregate bandwidth of all traffic to a given destination does not exceed the available bandwidth at the output port (155 Mbit/s or 622 Mbit/s). Thus traffic per virtual connection should be limited to the output bandwidth divided by the number of sources.

Nevertheless, congestion can still occur because the trigger correlates the injection times of cells from different sources, leading to bursty traffic which may temporarily overflow the shared buffer memory of the ISEs. Therefore it is also necessary that the injection of cells destined to the same destination by different sources is skewed in time. Thus synchronization is required between sources at the level of one cell transfer time (2.7μs).

Each source must queue data for many events which are simultaneously being built in the destinations. In order to fill the available bandwidth on each physical link, all virtual connections should be kept busy; therefore each source starts sending the data for the next event assigned to the virtual connection as soon as the current local event fragment is transferred (asynchronous event building). Because event fragments have varying sizes, the building of different events will overlap in the destination. The number of event fragments queued in the sources and the number of concurrently built events in a destination depends on the spread of event fragment sizes and the load on the switch. The total event building latency (including source queuing time) therefore also depends on these same factors.

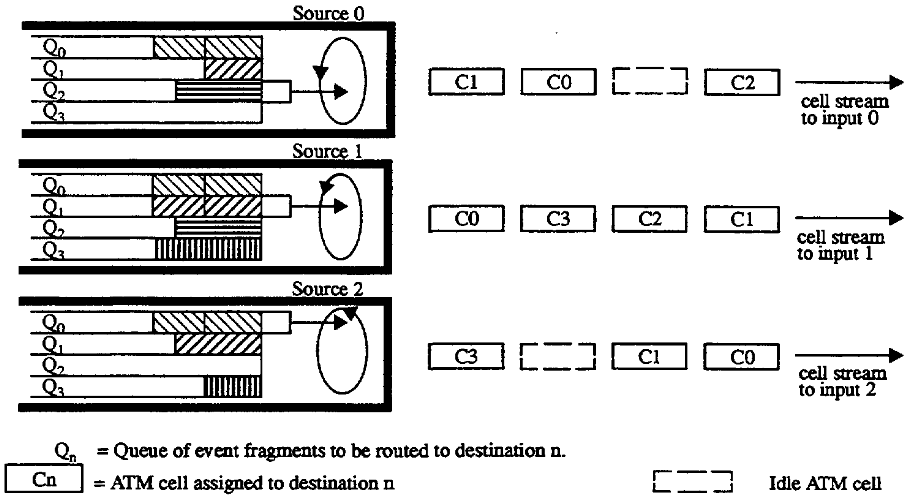


Fig. 12 A cell-based barrel shifter scheme.

In the destination, data belonging to different events are written to the appropriate buffers selected on the basis of the VCI field, which depends on the event sequence number. This scheme behaves like a cell-based barrel shifter. It uses the bandwidth resources of the switch efficiently. However, it is not obvious how to maintain the synchronization between sources in a practical implementation.

5.2.4 Source traffic shaping scheme 3 - the randomization scheme

The previously mentioned schemes require synchronization between the different sources, and for this reason they may be difficult to implement in practice. The third scheme dispenses with synchronization and centralized control logic by using rate control at the sources and introducing a random jitter on the cell injection times in order to emulate the "telecommunications" traffic patterns for which the switch design has been optimized. The internal buffering of the ISEs smooths any residual statistical burstiness of the traffic. This scheme should be easy to implement and robust because it avoids centralized control or token passing mechanisms.

Figure 13 shows the functionality of a source module. Like the source for the cell-based barrel-shifter scheme, it contains M (logical) queues of cells, each queue containing the cells to be routed to one of the M destinations. The rate control logic ensures that periodically one cell is read from the head of each logical queue. As before, the servicing of the queues is performed at a rate determined by the bandwidth of the destination output port and the total number of sources (N).

Before injection into the switch, cells are written into a buffer memory at a pseudo-random address. The buffer memory is read by scanning in round robin order, ensuring that no cell suffers a delay greater than one complete scan. The purpose is to randomize cell injection times and to break any long-term time correlation between traffic generated in different sources.

In this scheme the tail distributions for ISE buffer occupancy and network latency are very similar to the case for the "telecommunications" traffic patterns shown above in fig. 8, and we conclude that cell loss probabilities will be very similar to those predicted by Alcatel engineers ($<10^{-10}$ at 80% load). Utilization of available bandwidth can be as high as 80% and event building latencies are the same as for the cell-based barrel shifter scheme.

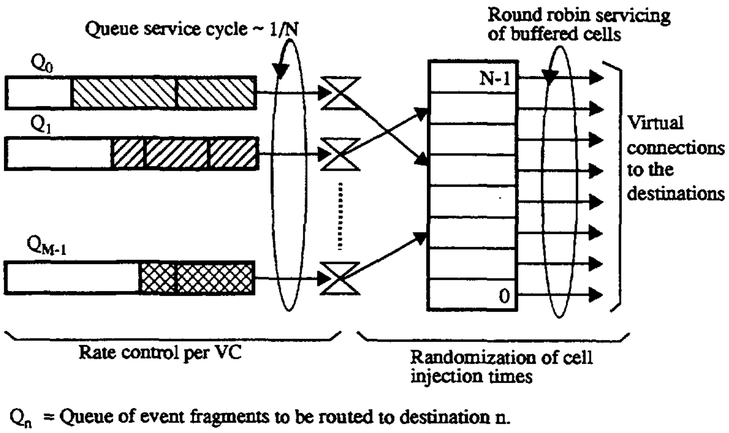

Queue service cycle ~ 1/N
Round robin servicing of buffered cells
Q_0
Q_1
Q_{M-1}
N-1
0
Virtual connections to the destinations
Rate control per VC
Randomization of cell injection times
Q_n = Queue of event fragments to be routed to destination n.

Fig. 13 Source traffic randomization scheme.

The rate-division on each VC stretches in time the transfer of each individual event fragment. As a result the overheads associated with the software protocols required to initiate and terminate the individual event fragment transfers are small compared to the transfer times. Therefore the impact of software protocol overheads on the achievable load factor is much smaller than it is in the case of the circuit-switched barrel shifter

When traffic shaping is used to minimize congestion, the bandwidth that can be delivered to a given destination is independent of the size of the switching fabric. Therefore the switch can be scaled to very large dimensions without losing efficiency.

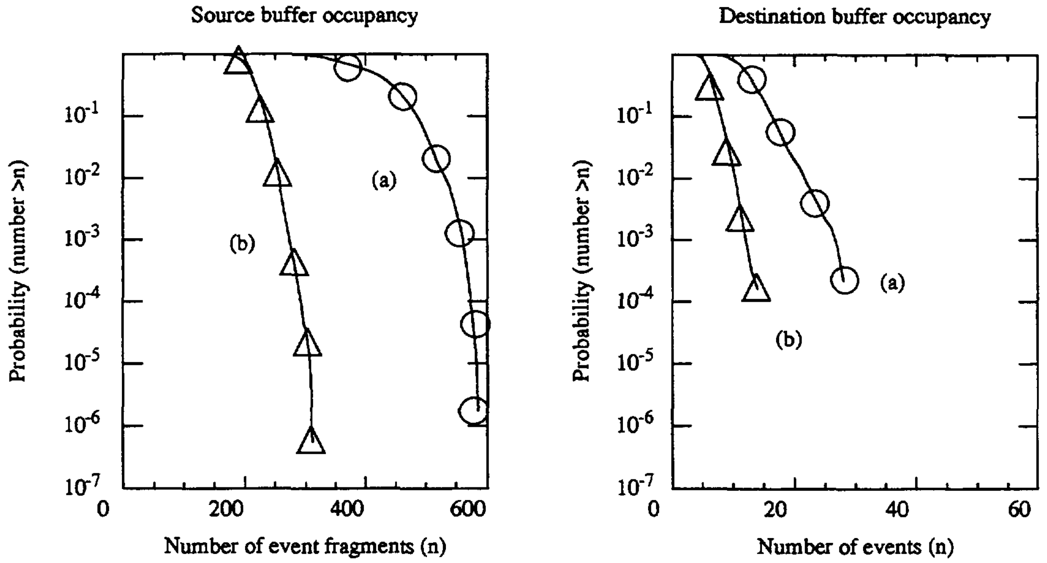


Fig. 14 Source and destination buffer occupancy for two switch configurations, both building events of an average size of 74 kByte at a trigger rate of 24 kHz; (a) a 128 x 128 switch operated at 80% load, (b) a 149 x 149 switch operated at 70% load.

Figure 14 shows, for an average total event size of 74 kByte and a trigger rate of 24 kHz, the tail distributions for the number of event fragments buffered in a source and the number of events being concurrently built in a destination. It shows the required source and destination buffering for two event builders of different sizes, which are operated at a load factor of 80% in one case and 70% in the other case. The average event building latency for these two cases is 67 ms and 40 ms respectively. A trade-off can be made between event building latency, the efficiency of utilization of available switch bandwidth (i.e. cost of the switch), and the buffer memory requirements at the sources and destinations. Figure 15 shows the dependency of event building latency, and the required buffer space in the sources and destinations on the average load.

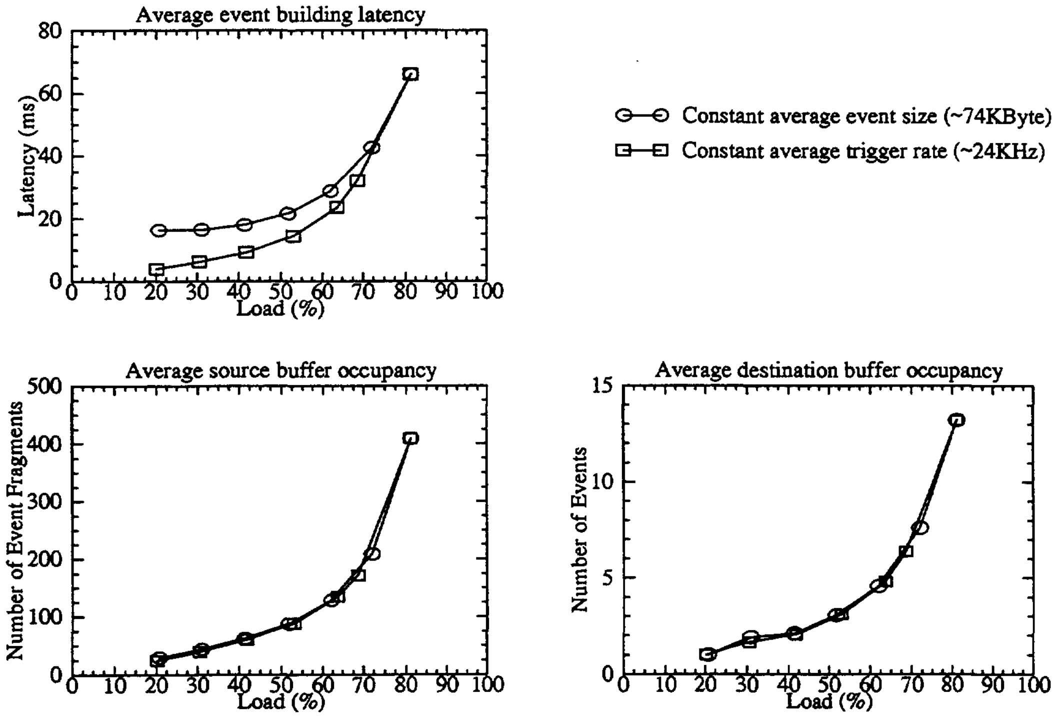


Fig. 15 Load dependency of event building latency and required system buffering

Figure 16 shows how the average event building latency (dominated by queueing in the sources) scales with the size of the event builder for a constant load of 80%.

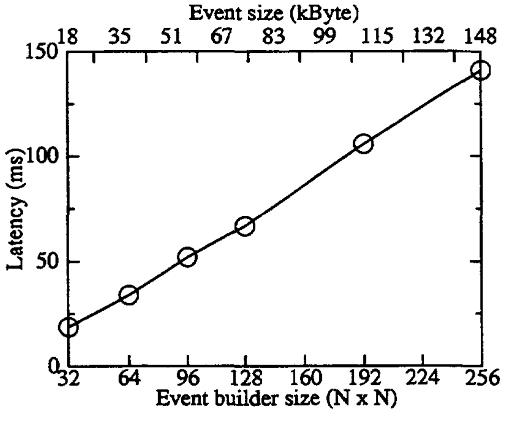


Fig. 16 Scaling of the total event building latency at a constant load of 80% (trigger rate = 24 KHz; local event fragment size = 0.58 kByte) using the traffic randomization scheme.

5.2.5 Destination Assignment Strategies

Destination assignment is an important aspect of traffic shaping such that a minimum of congestion occurs. The optimal destination assignment strategy is very dependent on the specific switching architecture used. A strategy assigning events sequentially (event i assigned to processor i) may work efficiently for some switching architectures. For architectures where some resource in the switch is shared between adjacent destinations a serious hot spot in the switch may occur.

An optimal destination assignment strategy can be found which takes into account the internal architecture of the Alcatel MPSR switch and assigns events to destinations so that cells belonging to successive events are always served by different queues in all switching element stages of the switch, thereby minimizing congestion. When the switch is loaded to 80% and the traffic randomization scheme is employed, the probability of a cell being discarded by the switch is 10^{-9} with sequential destination assignment, and 10^{-10} with the optimal destination assignment scheme.

5.3 An event builder based on a flow-controlled ATM switch

In this section we discuss an event building architecture that adopts the approach of using a flow-control protocol on the internal links between switching elements. This model is based on the AT&T / EPFL Phoenix switching fabric previously described in section 4.2.

The flow-controlled event builder architecture dispenses with the previously described traffic shaping function in the sources. In contrast to the case where traffic shaping was used, each source contains just one logical FIFO queue in which the event fragments are stored in the order in which they are generated. On receiving the broadcast trigger and destination assignment information, each source outputs the entire event fragment at the full bandwidth of the interface (nominally 155 Mbit/s).

5.3.1 Modeling of the flow-controlled Phoenix-based ATM event builder

As for the case of the event builder based on the Alcatel switch, the model has been coded in μC++. The main program plays the role of an event generator, deciding when an event occurs and distributing its data among the sources. The sources, destinations, ALI interfaces and the switch fabric are C++ objects that execute concurrently. The events are generated at random times at an average rate of 50 kHz, and the event fragment sizes are chosen so that the desired average load is achieved. In the results presented here, we used an average event fragment size of 270 bytes, corresponding to an average external load of 80%. There is no correlation of event fragment sizes between neighboring sources.

In our model, the assignment of a destination to an event is done by sending consecutive events to destinations that are chosen in such a way that congestion within the switch is minimized. Several switch sizes have been modelled, all being of type N x N with N = 16, 32, 64, and 128. For the various switch sizes we generated between 4000 and 8000 events.

5.3.2 Simulation results and comparison with the traffic shaping approach

Figure 17 (a) shows mean event building latencies, i.e. the mean elapsed time from the event trigger until the entire event has been collected in the destination as a function of the number of sources (N) and destinations (N) for a “square” N x N event builder, where the traffic load factor is kept constant at 80%.

The figure shows the event building latencies obtained by using the ALI and Phoenix chips, in the shuffle network configuration described above, and applying the back-pressure flow control mechanism. For comparison, figure 17 also shows the event building obtained under the same load conditions with the Alcatel MPSR switch and the randomizing traffic shaping scheme.

In the case of the switch using flow control, the event building latency is the sum of the time spent queueing at several different points; namely the source queues, the input ALI buffers, the internal queues in the Phoenix switching elements, and the output ALI buffers.

For small switch sizes (N), the event building latencies are considerably shorter for the switch using the back-pressure technique. This is because the number of events being simultaneously built in each destination is different in the two cases, as shown in figure 17(b).

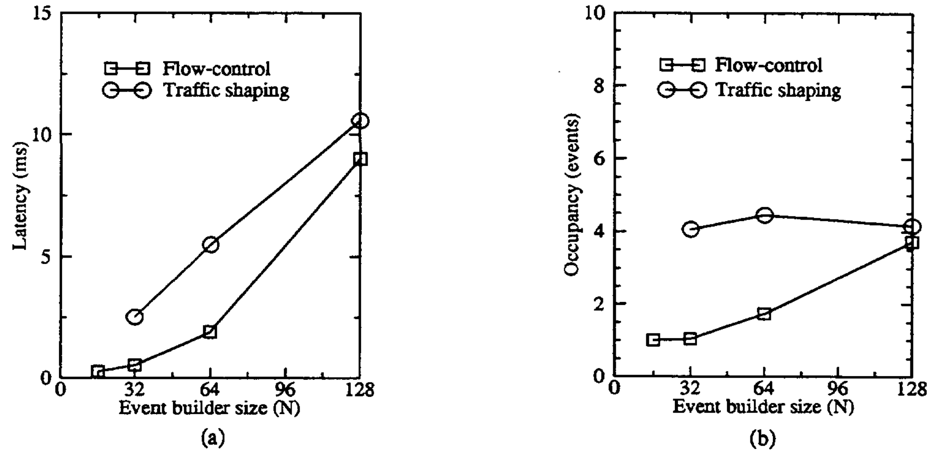


Fig. 17 (a) mean event building latencies and (b) the average number of concurrently built events per destination. The load factor is 80%.

The traffic shaping scheme results in several events being simultaneously built in each destination, even for a small switch. The sharing of the output port's available bandwidth between the data flows of the several concurrently built events results in a stretching of the latency. In the case of the flow-controlled switch, the uncontrolled concentration of traffic starts to fill the internal buffers of the Phoenix switching elements, causing them to apply back pressure. The resulting blocking of the network has the effect of spreading, in time, the data flow of each event and causing the traffic of consecutive events sent to the same destination (every Nth event) to start overlapping. In figure 17(b) we see that, for a 128 x 128 event builder, event fragments from an average of 4 different events arrive at an output before the first event has been completely built. The comparison of the two approaches for larger event builders is currently limited by the run time of the simulation model of the flow-controlled switch.

More detailed discussion of the results is given in [19].

5.4 Event building protocols

The modeling work described above has mostly evaluated the event building performance under the traffic patterns associated with a simple generic "level-3" event builder in which the entire event data are sent to the destinations. It is also proposed to implement more complex strategies, e.g. a "virtual level-2" trigger in which a fixed sub-set of the detector, or only the data from a region of interest (RoI), defined by the first-level trigger, are sent to members of the RISC processor farm, which will subsequently ask for the entire event data to be sent for those events that pass the second-level trigger algorithms. These more complex traffic patterns will be controlled by a data acquisition protocol, which can be implemented in a flexible manner by incorporating a protocol processor in each source. Participants in the protocol would be the first-level trigger, destination assignment logic and the sources and destinations.

The efficient implementation of these protocols will be facilitated by using a hardware medium which supports bi-directional source-to-destination communication, multiple selective multicast trees (to multicast "send" requests to the sources within a region of interest) and broadcast functions. Rather than installing a dedicated communication medium for carrying the protocol traffic, one should try to use a medium which anyway has to be installed for other purposes. The fibre optic timing and control distribution system [20] cannot be used for this task because it is unidirectional. Therefore the best candidate appears to be the ATM event builder itself. The Alcatel switch fabric does support full-duplex communication, selective multi-cast trees and broadcast capability. However, if we use this option the switch configuration will have to be twice the size of a switch in which we use only one direction.

Just as we pre-assign virtual connections to carry the event data, dedicated virtual connections and multicast-trees would be pre-assigned to carry the protocol traffic. Of course the protocol and data cell streams will be mixed inside the switching fabric, and there will be a small probability of losing a cell carrying protocol information. Another point to be investigated concerns the contribution of the protocol traffic on the overall load on the switch and how it will impact the functioning of the traffic shaping strategies we propose to use for the event data flows.

The ATM layer of the ITU standards allows individual cells to be marked with a high or low priority for being discarded when congestion is encountered. However this mechanism is not (and will not be) implemented in the Alcatel switch (and maybe not in other telecoms switches?). Clearly an important issue, that has to be studied, is whether we can design a robust data acquisition protocol that can function even in the presence of the occasional loss of a cell carrying protocol information.

A critical part of the event builder design will therefore be the protocol software (firmware) running in the sources. Because event fragment sizes are expected to be relatively short, the protocol software overhead will be significant when using the typical (~20 MIPS) embedded processors available today, but will scale down with the introduction of more powerful processors. In the level-two event building application, if we are to support level-one trigger rates up to 10^5 Hz, it will be important to use the RoI information to selectively activate sources and thereby reduce the average protocol handling load on individual sources.

The detection of errors in the transfer of packets to the destinations is handled by the AAL5 protocol, and the segmentation chip sets [21, 22] support this through CRC checking and packet reassembly time-outs. Another layer of protocol, that sits between the AAL5 layer and the previously described DAQ protocol layer, will involve signaling that will allow the destination to decide when event building is complete (i.e. the destination has received all the data from all the sources).

5.5 Event builder performance evaluation using Monte Carlo data

All results presented up to this point are based on a very simple "event generator" that uses an exponential distribution, with an upper cut-off for the event fragment size, and exhibits no correlation of the event fragment sizes between sources. The degree of clustering of the data in this simplified model, measured by the ratio of maximum to average event fragment size is 3. In reality, we expect event data will be more strongly clustered in space, and will also exhibit correlations between sources. Our model is therefore certainly oversimplified.

Therefore, in collaboration with colleagues from the Atlas experiment [23] we are using physics Monte Carlo data from the Atlas calorimeter barrel to generate a more realistic model of the expected traffic. However, Monte Carlo generation of the dominant background events and the simulation of their interaction with the detectors and first-level trigger is a very computing intensive task, and only limited statistics have been produced.

5.5.1 The Monte Carlo Data Set

A sample of 117 735 two jet events were generated using PYTHIA with mod(η) $\leq$ 0.5 and with the initial parton transverse energies above 35 Gev. These events were filtered by requiring that there be at least 35 GeV of energy in the event contained in a cone of dimensions $\Delta\eta \times \Delta\phi = 0.2 \times 0.2$. The level-one calorimetry trigger algorithm operates by scanning a search window of dimensions $\Delta\eta \times \Delta\phi = 0.4 \times 0.4$ over the entire surface of the calorimeter. This scan proceeds in steps of 0.1 in both η and ϕ and involves making energy sums around a reference cell (see figure 18 for cell numbering).

In the first stage the energy sums

$$\sigma_1 = E(em)_1 + E(em)_2$$

and

$$\sigma_2 = E(em)_1 + E(em)_4$$

are made, where *em* refers to the electromagnetic section of the calorimeter.

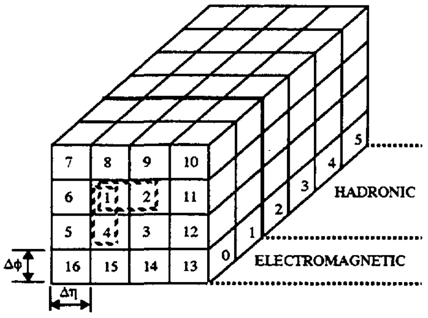


Fig. 18 A section, 0.4 by 0.4 in Δη×Δφ, of the calorimeter

In the second stage the energy sum

$$\sigma_3 = \sum_{i=5}^{16} E(em)_i + \sum_{i=1}^{16} E(had)_i$$

is made, where *had* refers to the hadronic section of the calorimeter. For the local level-one single isolated cluster calorimetry trigger, which is expected to produce the dominant rate, the core energy, defined as, $E_{\text{core}}= \max(\sigma_1,\sigma_2)$, is required to be more then 35 GeV and the isolation energy, defined as, $E_{\text{iso}} = \sigma_3$, is required to be less than 5 GeV.

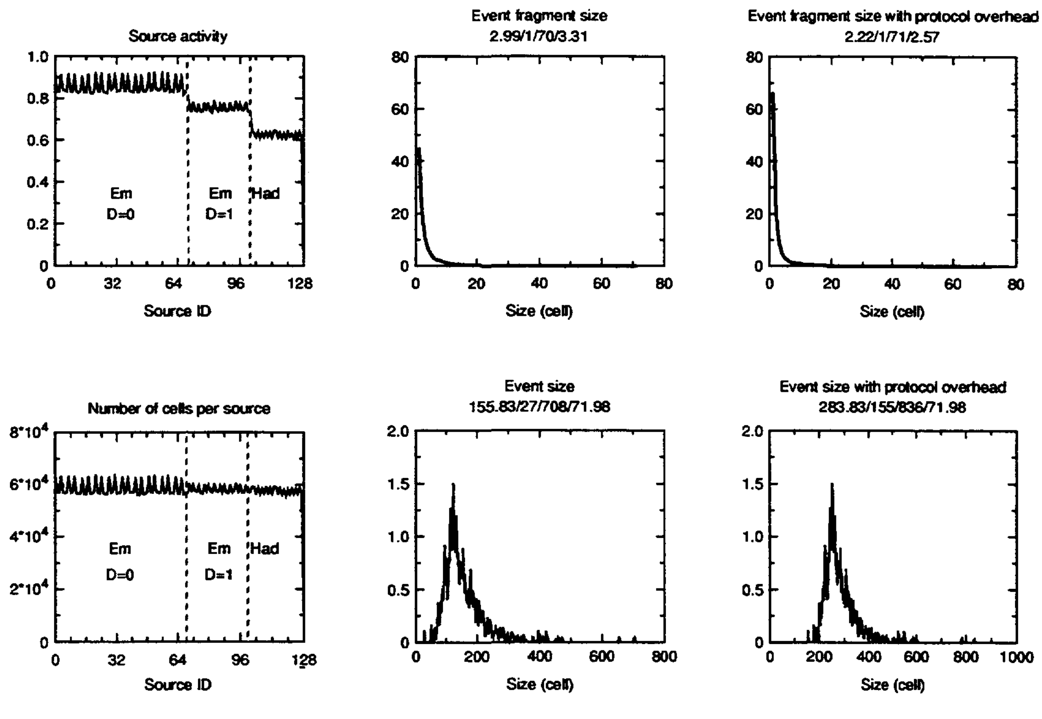


Fig. 19 Event data statistics obtained from the Monte Carlo data.

Only 529 events passed this level-one trigger selection. These events are now being used for the event building simulation studies. They contain data from the entire calorimeter map after suppression of data below a threshold of 100MeV. In order to effectively increase the statistics for the event builder simulations, events were picked at random from this master set, rotated randomly in the ϕ co-ordinate and then the event data were mapped on to 128 event builder source modules. A sample of some 25 000 events was generated in this way. Figure 19 shows the resulting statistics for sources, along with the event fragment size and event size distributions.

Note that, although the sources in different regions participate in the event building process with different frequency ("Source activity" histogram), the total number of cells to be injected in the event builder cross-connect is approximately equal for all sources ("Number of cells per source" histogram). Due to the clustering of data, the traffic emitted from the sources is characterized by a high degree of "burstiness". Taking into account the overhead cells required to implement a simple protocol allowing detection of the completion of event building, the average event fragment size is 2.22 cells and the maximum is 71. In this case, the additional traffic generated by the overhead of a simple protocol to signal completion of event building is ~80%. By mapping data from more channels onto each source we can increase the average event fragment size, reduce the burstiness of the data, and reduce the relative protocol overhead. More details can be found in [24].

5.5.2 Performance Evaluation of the Event-Builder with the Monte Carlo Data

We simulated a 128 x 32 event builder using the source fragment distributions derived from the Monte Carlo events as described in the previous section. The inter trigger delay follows a negative exponential distribution with an average value corresponding to a trigger rate of 33 kHz, which resulted in an average load of 20% on the input links, and 80% on the output links. The event fragment sizes for the active sources are optionally adjusted by adding protocol overhead cells.

For the 128x32 event-builder operating with the simple protocol traffic overhead, the event-fragment size and event size distributions are shown in figure 19 (the "Event fragment size with protocol overhead" and the "Event size with protocol overhead" histograms). Figure 20 shows the simulation results for some 24 000 assembled events.

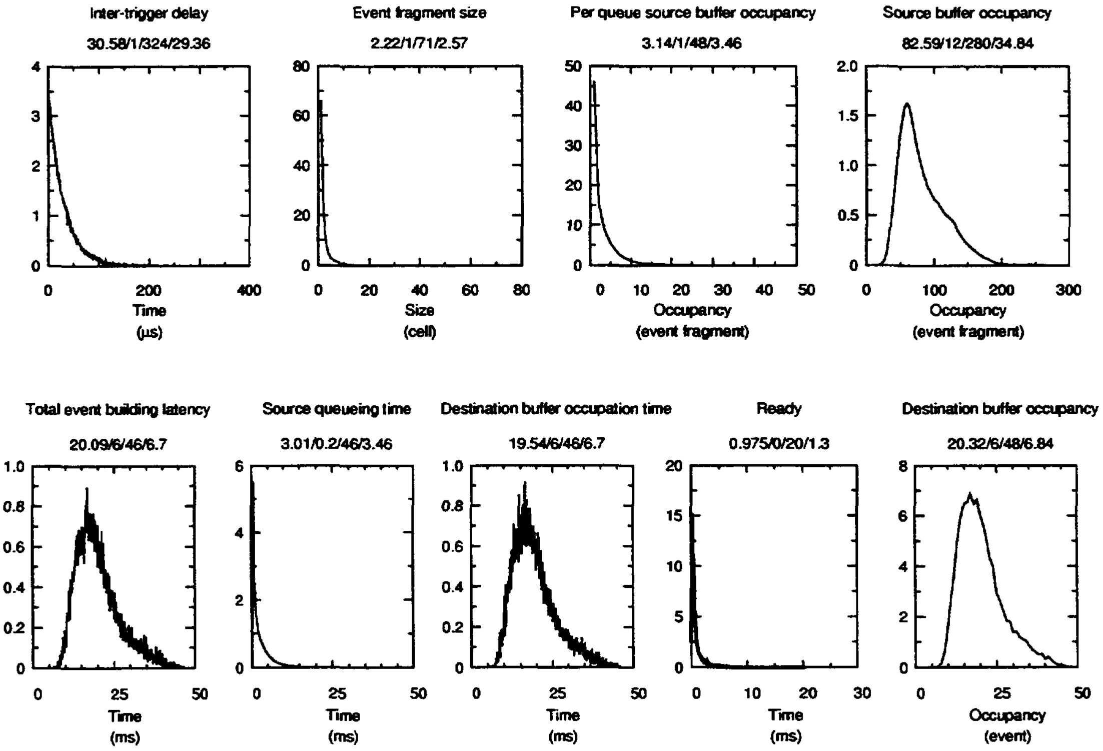


Fig. 20 The 128x32 event builder operating with Monte Carlo data and protocol overhead traffic

In order to sustain the average 80% load at the outputs with the wide spread of event fragment sizes presented at the input of the event-building cross-connect, it is necessary to increase the size of the buffers in the destinations (because more events are concurrently assembled in each destination). The studies of this event-builder's performance under the "bursty" traffic patterns generated by the Monte Carlo data show that the randomizing traffic flow control scheme smooths the highly bursty traffic presented by the detectors, so that the event builder performs with a cell-loss probability of 10^{-10}. More details are given in [24].

5.5.3 Simulation of a simple protocol with the Time-Out Technique

In the previously presented case, we have simulated a simple generic event builder in which all sources participate for every event, and in which we add the traffic overhead associated with a simple protocol that handles sources with no data for particular events by having them send an "empty" AAL5 packet (one cell) in order to allow the destination to decide when it has received the data from all sources.

Another possible technique is based on the use of a time-out watch dog. Only those sources which contain data for an event participate in the event-building process. As soon as the first cell of an event arrives at the destination, the watch dog counter is started. When the counter overflows some predefined time value the event building for the event is regarded as finished and the event is passed on to processing.

Figure 21 compares simulation results for the two different protocol techniques (event-building "by event size" and "by time-out") for the case of the 128 x 32 event builder. The traffic overhead associated with the "by time-out" protocol is 33% for this case.

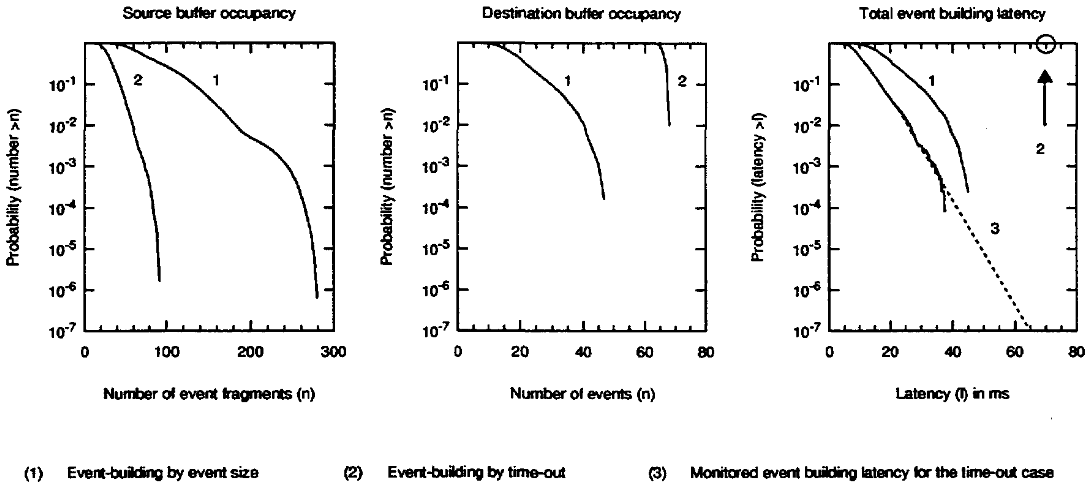


Fig. 21 Tail distributions for event building parameters for the two different protocol types (128 x 32 event-builder)

As before, the average trigger rate was 33 kHz. The time-out protocol creates ~50% less overhead traffic compared to the "by event size" case. This leads to a significant reduction of the required amount of memory in the sources. On the other hand, approximately 65 events are built concurrently in each destination. Indeed, by linear extrapolation of the monitored event-building latency tail (curve 3 on the figure) the time-out delay budget was found to be equal to 65ms. This value guarantees event loss probabilities of the order of 10^{-7}.

6. A TEST BENCH FOR EVALUATION OF ATM EVENT BUILDERS

As shown in figure 22, we are assembling a test bench for the evaluation of ATM-based event builders. Hardware components of the test bench are the ATM event builder itself, VME modules that act as sources and destinations for the ATM traffic, HP broadband test equipment for SDH/SONET and ATM protocol testing, and optionally an HP 9000/747i UNIX workstation with an ATM interface (for evaluation of event building in the UNIX workstation environment).

The UNI (user network interface) will be based on the SDH STM1 (equivalent to SONET OC-3) standard in order to facilitate interworking between equipment from different manufacturers. In principle any ATM switch that is SONET/SDH compliant could be plugged in to this test environment

We propose to commence the evaluation of ATM event builders using an 8-port Alcatel MPSR switching fabric configured as a VP/VC cross-connect. Each port supports a full duplex 155 Mbit/s SDH (equivalent to SONET) long-haul link, consisting of two single-mode fibres driven by laser-diode transmitters. The switch is delivered with management software for configuration, testing and monitoring. The management software runs on a SUN workstation in an X-windows environment. An Inmos B300 ethernet to transputer-link interface acts as the hardware interface between the switch and the management workstation.

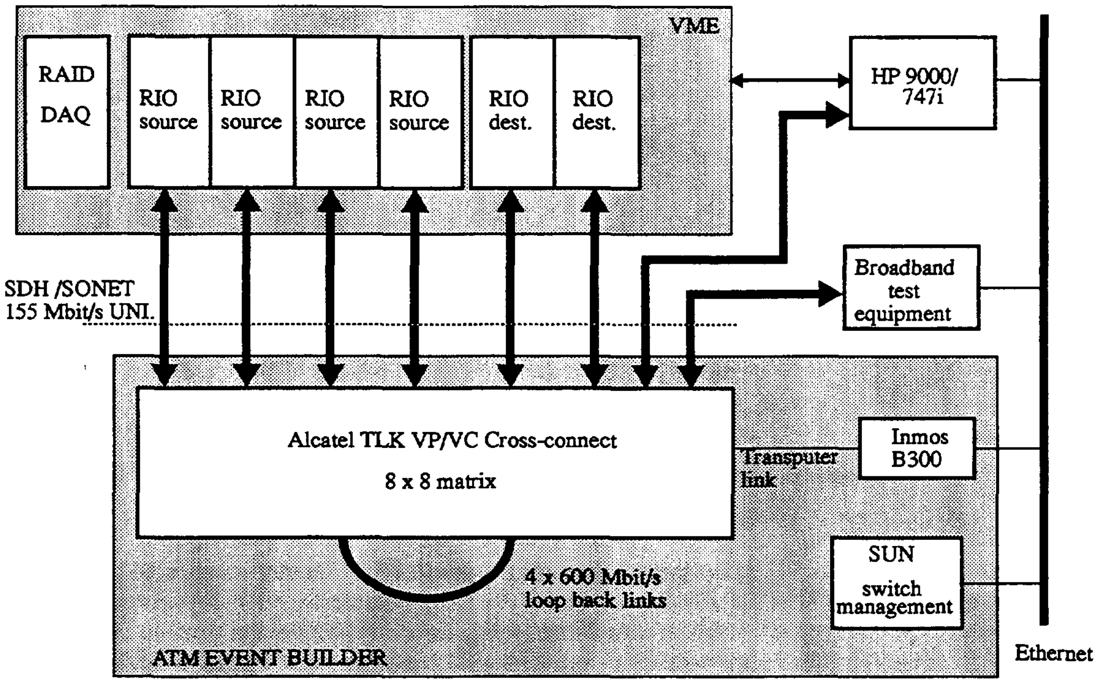


Fig. 22 Test bench for evaluation of ATM event builders

The HP workstation is already delivered and serves as a good host for our simulation work. The Alcatel switch (with Inmos interface and management software), and the HP broadband test equipment are expected to be delivered shortly. An ATM adapter has been developed by the university of Pennsylvania [25] for the "Afterburner" experimental high performance network buffer [26] for the HP workstation. However, the physical layer of the university of Pennsylvania's adapter is not compliant with the standard SDH STM1 physical layer UNI, and we are therefore developing the required physical layer interface daughter board (see below). Further details on the workstation interface can also be found in [27].

6.1 VMEbus ATM interface development

Although commercial VME-ATM interface modules exist, they do not provide the special traffic shaping function required in the source modules by the event building application. Therefore we are implementing our own VME ATM-interface modules. These modules use commercial chip sets supporting the standard ATM protocols and will incorporate our special event building traffic shaping logic. Note that the major part of the hardware and software design effort for the VME interface could be reused at a later stage when developing an interface to a front end buffer memory.

The ATM source and destination VME-modules will be implemented using the RISC I/O (RIO) board [28]. The RIO is designed to ease the hardware and software development effort needed to implement VME I/O protocols. Part of the board is empty, and can be used by the designer to implement specific hardware to efficiently support the lower layers of the target I/O protocol. Higher layers of the target protocol can be implemented in firmware. For this purpose the board is equipped with a MIPS R3000 processor, memory, a VMEbus interface, timer circuitry etc.

Figure 23 shows the architecture of the RIO-based hardware we are developing in support of the physical, ATM and AAL5 layers of the standard ATM protocols. Higher layers of the DAQ protocol will be implemented in firmware (or software) in the RIO host processor.

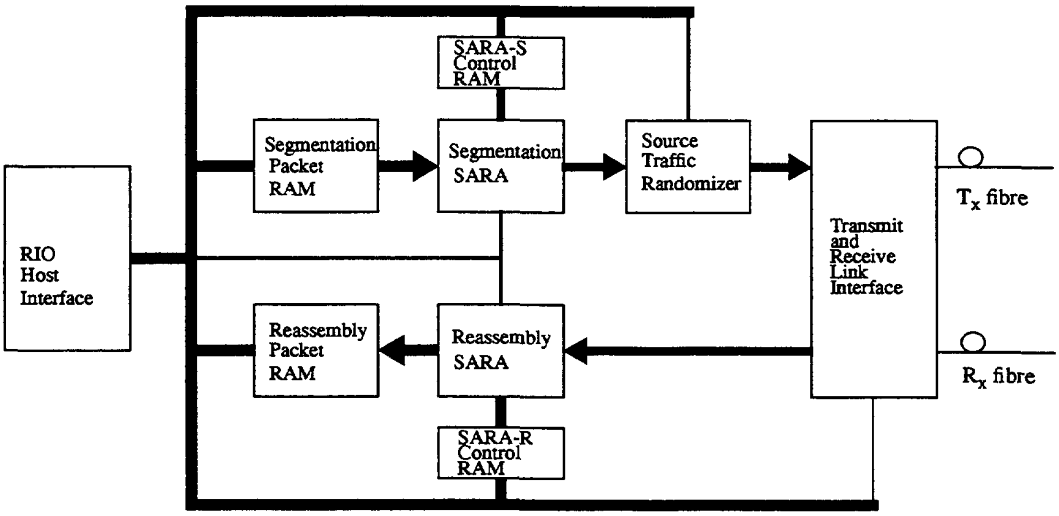


Fig. 23 Architecture of the RIO-based VMEbus ATM source module

The AAL5 and ATM layers of the protocol are implemented in hardware by commercial segmentation and reassembly (SARA) chips [21]. These support up to 8k simultaneously active virtual connections per interface and include sophisticated features for data queuing and virtual connection bandwidth control. For flexibility, the special traffic shaping logic will be implemented in programmable gate arrays. The physical layer will be implemented on a separate daughter board, that can be interchanged with other daughter boards to support different physical layer standards.

In order to transmit an AAL5 data packet, the RIO host requests the SARA to allocate a descriptor number that identifies a free buffer in the segmentation packet RAM. The SARA retrieves the descriptor number from a "transmit complete queue" that it maintains in the SARA-S control RAM. The host then places the event data block in the allocated buffer in AAL5 format. The length field and the padding field of the packet trailer are created by the host firmware. The host then builds a packet descriptor that defines the VCI and the cell transmit rate to be used on the virtual connection, and places it in the descriptor table (also maintained in the control memory). In the cross-connect configuration used for event building this descriptor table can be pre-calculated and pre-loaded. Packet segmentation and transmission are then initiated by writing the descriptor number in a "packet ready queue" in the control RAM. The SARA fetches the packet descriptor and segments and transmits the packet according to the parameters defined therein. The packet CRC is calculated on the fly by the segmentation SARA and inserted at the end of the trailer. After packet segmentation is completed the packet descriptor number is returned to the "transmit complete" queue. A similar process occurs for packet reassembly at the destination.

In addition a simple VME module is being developed to act as an ATM traffic generator. This will be used to provide an early means of generating traffic for testing the Alcatel switch and to gain experience with ATM chip sets from a different silicon manufacturer [29]. It will use an adaptation layer controller (ALC) chip that can perform AAL5 segmentation and reassembly, with "leaky bucket" rate control and support for up to 36 different peak rates. A network termination controller (NTC) chip will provide SDH/SONET framing and an address translation controller (ATC) chip will be used to provide VCI mapping at the UNI.

Figure 24 shows the daughter board that we are developing to implement the 155 Mbit/s SDH/SONET based option for the physical layer of the UNI. All of the required functionality of the SONET UNI (SUNI) is provided by a commercial VLSI chip [30]. This board could be reused to provide the required physical layer interface for the ATM adapter to the Afterburner interface.

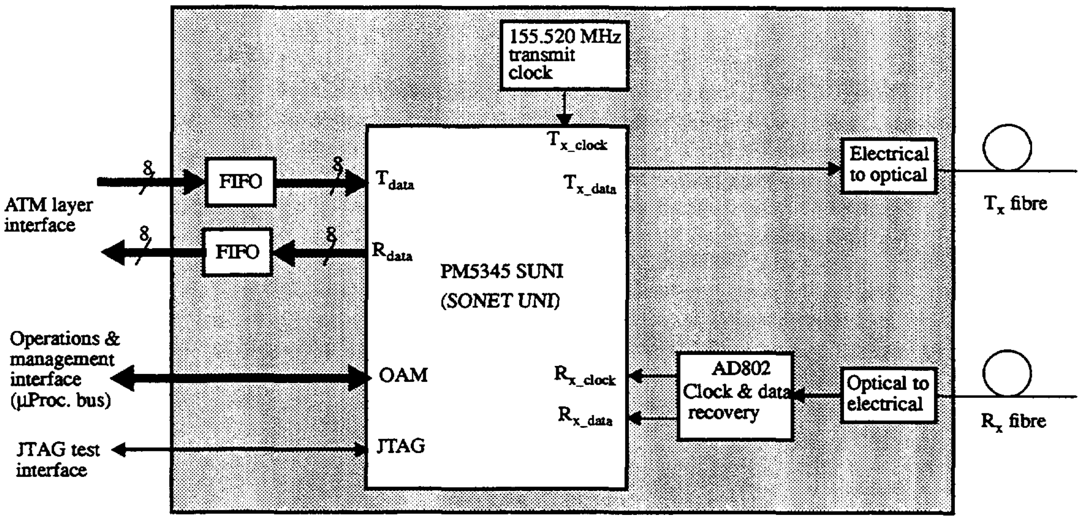


Fig. 24 Block diagram of the SDH/SONET physical layer daughter board

6.2 Data acquisition software for the test bench

The VME environment and the RIO modules were chosen in order to capitalize on the DAQ software developments and experience of RD-13 [31], where a similar development has been made for testing HiPPI-based event builders [32]. RD-13 have developed a scalable data acquisition software system running under the TC/IX real time UNIX operating system (a proprietary version of LynxOS) on a RAID [33] VME master module. The RIO module can be supplied with a stand-alone monitor and a VME library that allow communication with a RIO driver in the RAID. The RIO acts as an I/O server for one or more RAID clients. At a later stage the RD-13 DAQ software running in the RAID could be used to readout data from an on-line detector and push it into the event builder via the RIO clients.

7. PLAN OF WORK FOR THE SECOND YEAR

We will continue our simulation studies in order to conclude the comparative performance evaluation of traffic shaping versus flow control techniques and the scaling of the flow control technique to large switches. We will also investigate the performance of an event builder based on the combination of a switch fabric using internal flow control and a data acquisition system architecture that applies the traffic shaping technique. Some effort will be spent on the evaluation of alternative switch fabric architectures, for example LAN switches or a dedicated architectures optimized for event building.

An important task will be DAQ protocol design and the investigation of the feasibility of using the ATM switching fabric itself to carry the protocol traffic. This would include a study of the impact of the protocol traffic overhead on congestion control by the traffic shaping scheme.

We will continue the design and construction of a VME-based ATM source module with traffic shaping hardware. As part of that effort the design of a modular SONET physical layer link adapter board that can be plugged into the VME mother board has started, and will be continued. Interworking tests between the VME modules and the Alcatel switch will be carried out.

Hardware and software integration of the VME-based demonstrator system will be carried out, with the implementation of some simple DAQ protocols. Performance measurements will be made.

8. BUDGET REQUEST

In order to carry out the programme outlined in the above section for the year 1994, we estimate the required global materials budget (including travel and subsistence for the Swedish institutes, but not for CERN) as 490 kSFr. We request from CERN a contribution of 50%, corresponding to a 1994 budget allocation of 245 kSFr.

Global budget breakdown:

Maintenance of broadband test equipment and workstations (12% per annum)	25 kSFr
Broadband test equipment AAL protocol test and validation software	15
Alcatel 8x8 switch fabric, ethernet interface and management software	55
Switching hardware maintenance and upgrade options	30
Small workstation for operations and management of the switch	10
RAID-based development system, software licenses, etc.	30
(Opto)-electronic and electronic components	60
Circuit board layout, production and assembly in workshops	35
Lab instruments and infrastructure (VME crates, power supplies, scopes etc.)	60
Workstations for modelling and software development	60
Engineering consultancy	20
Travel and subsistence (KTH and Uppsala only)	40
Miscellaneous	50
TOTAL	490 kSFr

9. List of RD-31 publications and internal notes

(i) L. Gustafsson, Evaluation of different ATM test tools to be used in an ATM switch demonstrator system, RD-31 internal note 93-02

(ii) M. Letheren et al., An asynchronous data-driven event building scheme based on ATM switching fabrics, Proc. of the eighth Conf. on Real-time Computer Applications in Nuclear, Particle and Plasma Physics, Vancouver, Canada (June 1993), pp. 1-10.
Also available as CERN / ECP 93-14.

(iii) T. Lazraq et al., Performance evaluation of an event builder based on an ATM switching fabric with an internal link-level hardware flow control protocol, Proc. of the Open Bus Systems Conference, Munich (Nov 1993), pp. 163-169.
Also available as CERN / ECP 93-24.

(iv) I. Mandjavidze, Modelling and performance evaluation for event builders based on ATM switches, RD-31 internal note 93-06.

(v) I. Mandjavidze, A data-driven event building scheme based on a self-routing packet-switching Banyan network, RD-31 internal note 93-07.

10. Acknowledgements

We wish to acknowledge the following colleagues for their help with various aspects of the work reported here, or for enlightening discussions: D. Banks, E. Barsotti, F. Bourgeois, B. Brunner, R.K. Bock, J. Bovier, C. Calamvokis, B. Carpenter, J. Carter, S. Cittolin, C. Dalton, M. Delfino, A. Edwards, J. Harvey, W. Hawe, J. Hughes, P-G. Innocenti, D. Johnson, J. Joosten, J. Kneuer, S. Lorentzi, J. Lumley, L. Mapelli, R. McLaren, G. Mornacchi, M. Nomachi, P. Oechslin, C. Petitpierre, M. Prudence, D. Samyn, K. Sarkies, J. Smith, J. Strong, B. Traw, W. von Rüden, A. Wiesel and F. Worm.

11. References

[1] Atlas collaboration, Letter of intent for a general-purpose pp experiment at the Large Hadron Collider at CERN, CERN / LHCC 92-4, October 1992.

[2] CMS collaboration, Letter of intent for a general purpose detector at the LHC, CERN / LHCC 92-3, October 1992.

[3] Relevant recommendations, drawn up by the ITU-TSS committee, and available from the International Telecommunication Union, Geneva, Switzerland are:

I.150, B-ISDN Asynchronous Transfer Mode Functional Characteristics,
I.211, B-ISDN Service Aspects,
I.311, B-ISDN General Network Aspects,
I.321, B-ISDN Protocol Reference Model and its Application,
I.327, B-ISDN Functional Architecture,
I.361, B-ISDN ATM Layer Specification,
I.362, B-ISDN ATM Adaptation Layer (AAL) Functional Description,
I.363, B-ISDN ATM Adaptation Layer (AAL) Specification,
I.413, B-ISDN User-Network Interface,
I.432, B-ISDN User-Network Interface - Physical Layer Specification,
I.610, OAM Principles of the B-ISDN Access.

[4] The ATM Forum, c/o Interop Inc., 480 San Antonio Road, Suite 100, Mountain View CA94040-1219.

[5] J. Christiansen et al., NEBULAS - A high performance data-driven event building architecture based on an asynchronous self-routing packet-switching network, CERN / DRDC 92-14 and CERN / DRDC 92-47.

[6] The International Telecommunication Union, Geneva, Switzerland, recommendation I.363, B-ISDN ATM Adaptation Layer (AAL) Specification.

[7] W.E. Leland et al., On the self-similar nature of ethernet traffic, presented at SIGCOMM'93.

[8] V.P. Kumar et al., Phoenix: A building block for fault tolerant broadband packet switches, Proceedings of the IEEE Global Telecommunications Conference, December 1991, Phoenix, USA.

[9] H.T. Kung et al., Use of link-by-link flow control in maximizing ATM network performance: simulation results, to appear in Proc. IEEE Hot Interconnects Symposium, '93, Palo Alto, California, 5-6 August 1993.

[10] M. Henrion. and D. Boettle, Alcatel ATM switch fabric and its properties, in Electrical Communications, Vol. 64, No. 2/3, (Alcatel Paris HQ, 33 rue Emeriau, 75725 Paris Cedex 15, France, 1990) pp.156-165.

[11] Relevant recommendations, drawn up by the International Telecommunications Union, Geneva, Switzerland are:

G.707, Synchronous Digital Hierarchy Bit Rates,
G.708, Network Node Interface for the Synchronous Digital Hierarchy,
G.709, Synchronous Multiplexing Structure.

See also G. Pellegrini and P.H.K. Wery, Synchronous digital hierarchy, Telecommunication Journal Vol 58 - Nov. 1991, pp815-824.

[12] J-Y. Le Boudec, The asynchronous transfer mode: a tutorial, Computer Networks and ISDN Systems 24 (1992) 279-309.

[13] M. Henrion et al., Technology, distributed control and performance of a multipath self-routing switch, in Proc. of the XIV International Switching Symposium, Yokohama, Japan, October 1992, Vol 2, pp. 2-6.

[14] P.A. Buhr et al., Concurrency in the object-oriented language C++, Software Practice and Experience, Vol. 22(2), pp. 137-172, (Feb. 1992).

[15] X. Yi-Jun, H. Michiel and G. Petit, Performance assessment of an ATM self-routing switching network using parallel programming techniques and a network of transputers, Proc. of the International Conference on Computers and Communications, Beijing, China, 1991, Session 5.23, pp.1-8.

[16] P. Oechslin et al., ALI: A versatile interface chip for ATM systems, Proc. of the IEEE Global Telecommunications Conference '92, Orlando, 6-9 December 1992, pp. 1282-1287.

[17] R.K. Bock et al, Embedded architectures for second-level triggering (EAST), CERN / DRDC 90-56, CERN / DRDC 92-11 and CERN / DRDC 93-08.

[18] D. Black et al, Results from a data acquisition system prototype project using a switch-based event builder, 1991 Nuclear Science Symposium, Santa Fe, New Mexico, November 2-9, 1991, Conference Record, Vol. 2, pp. 833-837.

[19] T. Lazraq et al., Performance evaluation of an event builder based on an ATM switching fabric with an internal link-level hardware flow control protocol, Proc. of the Open Bus Systems Conference, Munich (Nov 1993), pp. 163-169.
Also available as CERN / ECP 93-24.

[20] B. G. Taylor, Optical timing, trigger and control distribution for LHC detectors, CERN / ECP 93-10 (October 1993).

[21] Transwitch Corp., 8 Progress Drive, Shelton, CT 06484, SARA chipset, technical manual, November 1992.

[22] Adaptive Corp., 200 Penobscot Drive, Redwood City, California 94063, FRED chipset, technical manual, version 2.0, October 1992.

[23] D. Johnson and J. Strong, A study of a calorimetry based trigger system using single electron events and two jet events, ATLAS internal note, CAL-NO-30, October 1993.

[24] I. Mandjavidze, Modelling and performance evaluation for event builders based on ATM switches, RD-31 internal note 93-06 (December 1993).

[25] C.B.S. Traw and J.M. Smith, A high-performance host interface for ATM networks, in Proc. SIGCOMM '91, Zurich, Switzerland, September 1991, pp. 317-325.

[26] D. Banks and M. Prudence, A high performance network architecture for a PA-RISC workstation, IEEE Journal on Selected Areas in Communications, Vol. 11, No. 2 (February 1993).

[27] M. Letheren et al., An asynchronous data-driven event building scheme based on ATM switching fabrics, Proc. of the eighth Conf. on Real-time Computer Applications in Nuclear, Particle and Plasma Physics, Vancouver, Canada (June 1993), pp. 1-10.
Also available as CERN / ECP 93-14.

[28] Creative Electronic Systems SA, Geneva, RIO 8260 and MIO 8261 RISC I/O processors - user's manual, version 1.1 (March 1993).

[29] Fujitsu Mikroelektronik GmbH, Am Siebenstein 6-10, 63303 Dreieich-Buchschlag, Germany, the MB86687 adaptation layer controller (ALC), the MB86683 network termination controller (NTC) and the MB86689 address translation controller (ATC).

[30] PMC-Sierra Inc., 8501 Commerce Court, Burnaby, British Columbia, Canada V5A 4N3, the PMC5345 Saturn user network interface manual (May 1993).

[31] L. Mapelli et al., A scalable data taking system at a test beam for LHC, CERN / DRDC 90-64, CERN / DRDC 91-23, CERN / DRDC 92-13 and CERN / DRDC 93-25.

[32] W. Bozzoli et al., High performance event distribution using HiPPI, in Proc. of the International Conference on Computing in High Energy Physics, Annecy, France, September 1992, (CERN 92-07) pp. 192-195.

[33] Creative Electronic Systems SA, Geneva, RAID 8235 VME RISC processor board - user's manual, version 2.3, (February 1992)