DD

CERN - ECP 93-14

ju 8404

c 2

EUROPEAN ORGANIZATION FOR NUCLEAR RESEARCH

CERN/ECP 93-14
5 November 1993

CM-P00049302

**AN ASYNCHRONOUS DATA-DRIVEN EVENT BUILDING SCHEME
BASED ON ATM SWITCHING FABRICS**

M. Letheren, J. Christiansen and I. Mandjavidze^(*)

CERN, Geneva, Switzerland

H. Verhille, M. De Prycker, B. Pauwels and G. Petit

Alcatel Bell Telephone, Antwerp, Belgium

S. Wright and J. Lumley

Hewlett Packard Laboratories, Bristol, United Kingdom

ABSTRACT

Asynchronous transfer mode (ATM) packet-switching network technology is proposed as the interconnect for building high-performance, scalable data acquisition architectures. This paper introduces the relevant characteristics of ATM and describes components for the construction of an ATM-based event builder: (i) a multi-path, self-routing, scalable ATM switching fabric, (ii) an experimental high-performance workstation ATM interface, and (iii) a VMEbus ATM interface. The requirement for traffic shaping in ATM-based event builders is discussed and an analysis of the performance of several such schemes is presented.

Presented at the Eighth Conference on Real-Time Computer Applications in Nuclear, Particle and Plasma Physics, Vancouver, Canada, 8-11 June 1993.

(*) From the Georgian Academy of Science, Tbilisi, Georgia.

1. INTRODUCTION

The very high data rates expected in experiments at the next generation of high luminosity hadron colliders will be handled by pipelined front-end readout electronics and multiple levels (2 or 3) of triggering. A variety of data acquisition 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 10–100 Gbit/s.

These high bandwidths cannot be handled by the traditional time-division multiplexed (TDM) approach of bus-based data acquisition systems. The exploitation of multiple, optical or electronic point-to-point links (space-division multiplexing) will be necessary. A cost effective and expandable event-builder using both time and space division multiplexing can be realized with a parallel switching fabric providing $N \times N$ connectivity with of the order $N \log N$ hardware complexity. Event builders based on a number of commercial or special-purpose switches have been proposed [3–5] or used in experiments [6].

Recently the International Telecommunication Union's (ITU) CCITT committee has recommended the use of ATM as the switching technology for the future broadband (155 Mbit/s and upwards) integrated services digital network (B-ISDN). The CCITT's B-ISDN standards [7] are targeted at wide area networking (WAN) and telecommunications applications. More recently, ATM technology is also being adopted for high-performance Local Area Networking (LAN) applications.

Many believe that ATM technology will dominate high performance WAN and LAN networking throughout the time span of experiments at the next generation of hadron colliders. Pricing of early ATM products is probably too high for large scale applications in a physics experiment, but the growth of multi-media applications and a policy of fibre-to-the-home are expected to render ATM affordable on the time scale of the next generation of colliders. With these predictions in mind the RD31 project [8] has been started in order to investigate the use of scalable ATM switching architectures for event building.

In this paper we review the relevant characteristics of ATM technology and describe ATM equipment suitable for constructing an event builder. Then, we report on our work, carried out within the framework of the RD31 project, on modeling and performance analysis of ATM-based data acquisition architectures.

2. OVERVIEW OF ATM SWITCHING

2.1 The principle

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.

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 is used by a network of switching elements to route the cell to its destination.

Fig. 1 The format of the ATM cell.

Cells belonging to many different 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 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” onto the new high bandwidth B-ISDN infrastructure.

Fig. 2 The principle of the asynchronous transfer mode (ATM).

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^(*) 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 connections by manipulating the VCI mapping tables (a “cross connect” configuration).

(*) Throughout the remainder of the paper we use the shorthand VCI to refer to the combined VPI/VCI 24-bit label.

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 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 CCITT B-ISDN protocols

ATM is a rapidly evolving technology, and not all aspects are fully specified yet. The CCITT's B-ISDN standards [7] define a 3-layered protocol.

(a) 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.

(b) The ATM layer

The ATM layer handles the switching of cells from source to destination according to the VPI/VCI field 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). The ATM layer guarantees the delivery of cells in the correct sequence.

(c) The ATM adaptation layer

The ATM adaptation layer (AAL) consists of two sublayers which hide the complexity of the ATM layer from the user. The segmentation and reassembly sublayer (SAR) segments packets (i.e. messages) received from the application into cells at the source, and reassembles them at the destination before passing them to the application. The convergence sublayer (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 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 [9] for data transport applications is AAL5. This protocol is very popular because of its simplicity and has now been adopted for standardization by the CCITT.

Figure 3 shows how an AAL5 packet, which carries up to 64 kbyte of user data and is terminated by an 8-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 connections is performed on the basis of the VPI and VCI fields of the cell headers.

Fig. 3 The AAL5 packet segmentation onto the ATM layer.

2.3 Blocking and congestion control 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 so 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. Cells are discarded when a buffer overflow does occur. 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. One option for handling congestion is to incorporate a flow-control mechanism inside the switching fabric. This holds up the in-flow of cells to a congested zone by propagating a back-pressure signal or message upstream.

However, because the typical telecommunications traffic mix does not require it, flow control is not implemented in the majority of ATM switching fabrics. Additional reasons for this 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 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.

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. At the network input port a policing function enforces the agreed traffic rates for each VCI connection. 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.

3. EVENT BUILDING USING AN ATM CROSS-CONNECT

Figure 4 shows the use of an ATM switching fabric as a cross-connect to provide the $N \times M$ semi-permanent connections needed to perform event building from N sources (S) into a farm of M processors (P). In the cross-connect configuration the connections are not established dynamically, and therefore the complexities of signalling and admission control are avoided.

Fig. 4 Event building with an ATM cross-connect.

Each source uses M different VCI labels to identify the connections to the M destinations. At each trigger, the destination assignment logic generates and broadcasts the information used by the sources to select the appropriate connection for the event. The sources segment the data into cells, insert the appropriate VCI label in the headers, and drive them into the switching fabric. As mentioned before, the switch maps the VCI labels into new values which are used by the destination to identify from which of the N sources the cell was sent.

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. In order to operate without loss of data, an event builder using an ATM switch with no internal flow-control mechanism will therefore need bandwidth control at the sources and/or control of the relative cell injection times at the different sources.

4. ARCHITECTURE OF THE SWITCH FABRIC

This section describes a multi-path self-routing (MPSR) broadband switching fabric developed by Alcatel [10] for public network applications. 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 Large Hadron Collider (LHC). It has been used to study the performance of ATM-based event builders reported in this paper.

Additional reasons for choosing a switch architecture from a telecommunications equipment manufacturer are that their products are designed for a 10–15 year life-cycle which matches with the expected life time of future large-scale experimental facilities,

and because an evolution path to higher performance is foreseen. In addition, they provide 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.

Because it is designed for telecommunications traffic, the Alcatel switch has no internal flow control. Applications such as event building will generate congestion unless traffic shaping is applied at the sources. Section 6 discusses the implications of this on the data acquisition system design.

4.1 The MPSR switching fabric architecture

The switching fabric is a folded, multiplane, multistage structure built from switching modules (SM) that switch 128×128 links running at 155 Mbit/s each. In its largest configuration, shown in fig. 5, the fabric will consist of a 16-plane, 3-stage structure (AS, PS1 and PS2 stages). It supports up to 16 384 external 155 Mbit/s links. It will be built from integrated switching elements (ISEs) that switch 32 inlets to 32 outlets. The currently available fabric uses a lower level of silicon integration in which 16×16 ISEs are used to build 64×64 SMs. This permits the implementation of a fabric supporting up to 2048 external 155 Mbit/s links.

Fig. 5 The architecture of the MPSR switching fabric.

Traffic switching units consisting of link termination (TLK) boards and access switches interface bidirectional 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.

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

(a) 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. There is no need for setting up a routing path and for managing bandwidth on internal links. Connections can be thought of as being “virtual” rather than physical.

(b) 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).

(c) 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 an 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%. This deliberate internal under-loading is designed to sharply reduce the probability of congestion and consequent cell loss.

(d) Multipath

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 multipath 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 multipath 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.

4.2 The integrated switching element

The switching and path randomization functions are performed by an ISE16 that interconnects 16 serial inlets to 16 serial outlets, each operating at 155 Mbit/s. The ISE16 uses a shared buffer memory architecture, shown in fig. 6. The ISE 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*.

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 four, 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.

Fig. 6 The architecture of the ISE16.

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 multicasting 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 ref. [11].

4.3 Traffic switching units

The traffic switching unit consists of link termination (TLK) boards and access switches. On input the access switches randomize the traffic over the multiple folded switching planes, and on output they contribute to the basic routing function.

The TLK boards provide the traffic expansion required to ensure internal link under-loading. They also handle the framing protocol used on the 8 external links. Electrical or optical serial link interfaces using the synchronous digital hierarchy (SDH) standard [12] at 155 Mbit/s are available. For high bandwidth applications an alternative TLK board interfaces two optical links at 622 Mbit/s. The TLK board also handles mapping of ATM cells into and out of the internal MSC format. On input a VCI label translation table maps the VCI into the internal routing information required by the MSC. For input traffic a policing function is provided for every VCI.

The non-deterministic delay of cells routed over the different internal paths means that cell sequencing within a connection might not be preserved, hence a resequencing function is provided in the TLK boards before the cells enter the output queue. This operates by time stamping cells in the TLK 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.

5. INTERFACING TO ATM

The ATM adaptation layer of the B-ISDN protocols will be implemented in the interfaces to the source and destination modules. In the event building application, sources will probably be high-performance microprocessors running a real-time kernel and managing the building of local event fragments. Destinations could be either based on the high-performance real-time embedded microprocessor, UNIX workstations, or a shared event memory accessible by a farm of processors [2].

The single-board microprocessor solution could be used to handle the high event rates expected at the second level of the triggering hierarchy, while workstations might be used for the more compute-intensive analysis performed by the third-level trigger farms.

5.1 The workstation interface

Third-level processing farms are expected to be compute-bound, and therefore high-performance I/O may not be necessary. Nevertheless, in order to understand the limits of the technology, and the protocol processing overheads, it is planned to evaluate event building for workstations using an experimental high-performance ATM interface developed collaboratively between Hewlett Packard Laboratories and the University of Pennsylvania. This interface consists of two halves: a network buffer platform for HP series 9000/700 workstations (called the Afterburner) and an ATM segmenter, reassembler and AAL processor (the ATM link adaptor).

The Afterburner is designed to support a single-copy protocol stack (i.e. the data are copied only once) and can transmit and receive network packets at burst rates of up to 1 Gbit/s. Single copy stacks are one solution to the poor observed performance of networking throughput seen with the arrival of 100 Mbit/s links. The main obstacles to high network throughput in current workstations are the data copy rate of the main memory system and multiple copying in conventional implementations of protocol stacks. The details of this problem are expounded in refs [13,14]. A proposal, made by Jacobson [15], is to move the data only once, directly from the application buffer to the network interface card, and modify the implementation accordingly. The Afterburner architecture, shown in fig. 7, is based on this idea and an implementation of an FDDI interface [14].

Fig. 7 The architecture of the Afterburner network buffer.

In order to support a single-copy stack, the card contains a 1 Mbyte buffer of triple-ported video RAM (VRAM) which provides one standard random access port and two high-speed serial ports. The random access port is used by the CPU while the serial ports provide a fast-transmit pipe and a fast-receive pipe, each 32 bits wide and capable of bursting at 33 MHz. The VRAM is divided into blocks between 2 kbytes and 64 kbytes in size.

The transmit section has two FIFO queues which contain a list of blocks that are empty (Tx_free) and blocks that are waiting to be transmitted (Tx_ready). In the TCP protocol implementation, the socket layer acquires an empty block and copies the user data into it. The TCP checksum is calculated as the data is copied. Then, the lower layers build the packet header in another block and the driver places the addresses of the full blocks in the Tx_ready FIFO. The hardware on the card then manages the transmission of the packet. The same operation occurs in reverse order for the receive section. Afterburners operating in a back-to-back mode have been measured as achieving > 25 Mbyte/s TCP throughput, using packets around 14 kbyte in size.

The mediation between these pipes from Afterburner buffers and the ATM network is carried out by a link adaptor developed at the University of Pennsylvania [16]. On transmission, the adaptor receives large packets from the host via Afterburner and segments them into cells with appropriate AAL5 trailer generation. The re-assembly function de-interleaves incoming cells based on VCIs, re-assembles in an intermediate buffer (using a combination of content-addressable memory look-up and linked-list management) and then emits larger packets to the Afterburner. The theoretical performance of this adaptor is around 300 Mbit/s and sustained throughputs of 140 Mbit/s have been observed on machines where connection bandwidth has been the bottleneck.

5.2 A VME-ATM interface

For handling the higher event rates and shorter processing times of second-level triggering farms, an embedded real-time processor is most suitable. The RD31 project will investigate ATM-based event building in the VMEbus environment. For this purpose the MIPS R3000-based RISC I/O (RIO) board from Creative Electronic Systems [17] has been selected. The RIO has an architecture optimized for developing high-performance I/O protocols using a mixture of firmware and protocol-specific plug-in adaptor cards.

The RIO is designed to operate as an I/O server for VME processors (clients) and is provided with firmware that allows multiple clients to connect and interact with it over the VMEbus. A RAID R3000-based VMEbus master module [18], also from Creative Electronic Systems, running a real-time UNIX kernel and physics data acquisition system software [19] will be used as the client processor. The RAID can be used to perform the local VME readout required to build event fragments at the sources, and at the destinations it can analyse the built events.

Creative Electronic Systems are currently developing an ATM AAL5 adaptor for the RIO based on programmable gate-array chips in collaboration with the Ecole Polytechnique Fédérale, Lausanne [20]. The logic density achievable in the programmable logic limits the support for VCIs to a relatively small number. Therefore, the RD31 project will parallel this development with an ATM source module design based on a commercial ATM chip-set supporting a large number of concurrently active VCIs (e.g. ref. [21]). The RIO approach gives the flexibility of incorporating special traffic shaping hardware tailored to the requirements of event building. When used as an ATM destination module, the RIO off-loads the client of the overhead associated with the management of the virtual channels connecting to each of the many sources.

5.3 Front-end interfaces

Front-end interfaces could also be built around a dual-port memory which is loaded by a detector-specific controller that reads out and synchronizes event fragments from a group of channels, and formats them into AAL5 packets. The ATM interface retrieves the AAL5 packets from the other port and performs the required segmentation into ATM cells, VCI assignment and traffic shaping using a commercial ATM chip set. An on-board processor is required to manage the chip set and the higher levels of the data acquisition protocol.

6. PERFORMANCE ANALYSIS

6.1 Modelling the Alcatel MPSR switch

An extension of the object-oriented language C++ supporting concurrency [22] has been used to develop models of the ISE16 switching element, the 64 x 64 Switching Module (SM) and the TLK board. These building blocks have been used to assemble models of switching fabrics of up to 256 x 256 ports.

Figure 8 shows the probability distribution for shared buffer memory occupancy and cell “waiting times”^(*) for ISE16s 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 [23] have been used to dimension the shared buffer memory of the ISE16 and to engineer the maximum internal transfer delay used for re-sequencing the cell streams.

Fig. 8 ISE16 buffer occupancy and waiting time in a 128 x 128 switching fabric operated under a “telecommunications” traffic pattern at a load of 80%.

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

(*) The difference between the observed delay and the delay expected when no congestion occurs within the switching fabric.

than 10^{-10} for the largest switching fabric operating under a “telecommunications” traffic pattern and a load of 80%. In order to achieve this level of performance the ISE16 shared buffer memory is sized at 256 slots (2 kbytes).

6.2 Low latency event building

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 ref. [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 decision time.

Figure 9 shows how an ATM switching fabric could be used for RoI building and event building for the global second-level trigger proposed in ref. [24]. 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 600 μ s on average, of which data switching accounts for approximately 100 μ s.

Fig. 9 Switching for RoI building and event building for the global second-level trigger.

6.3 Congestion control in ATM event builders

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 ref. [2]). Here it becomes important, for economic reasons, to efficiently utilize the available bandwidth. In order to be able to run with a high load the traffic patterns must be shaped in such a way 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.

The performance of 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 data clustering in the detector, the local event sizes generated in the sources also followed an exponential distribution. A more realistic approximation to the expected traffic patterns is planned by using Monte Carlo data.

(a) Event-based barrel shifter

The first scheme studied, shown in fig. 10, emulates a circuit-switched barrel shifter, and is similar (but not identical) to the event building schemes proposed under ref. [3]. 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 their 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.

Fig. 10 An event-based barrel shifter scheme.

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.

(b) Cell-based barrel shifter

Figure 11 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 connection should be limited to the output bandwidth divided by the number of sources.

Fig. 11 A cell-based barrel shifter scheme.

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 to be routed 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 ($2.7 \mu s$).

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 queueing time) therefore also depends on these same factors.

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.

(c) Traffic 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 12 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)

Fig. 12 Source traffic randomization scheme.

Before injection into the switch, the 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 time correlation between traffic generated in different sources.

In this scheme the tail distributions for ISE16 buffer occupancy and network latency are very similar to the case for the “telecommunications” traffic patterns shown before 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.

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 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.

Figure 13 shows, for an average total event size of 75 kbyte and a trigger rate of 25 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 40% in one case and 80% in the other case. The average event building latency for these two cases is 15.5 ms and 29 ms. 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.

Fig. 13 Source and destination buffer occupancy for two switch configurations, both building events of an average size of 75 kbyte at a trigger rate of 25 kHz; (a) a 128 x 128 switch operated at 80% load, (b) a 256 x 256 switch operated at 40% load.

Figure 14 shows how the average event building latency (including queueing in the sources) scales with the size of the event builder for a constant load of 80%.

Fig. 14 Scaling of the total event building latency at a constant load of 80% (trigger rate = 25 kHz; local event fragment size = 4.6 kbit) using the traffic randomization scheme.

6.4 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. All simulation results presented here use this optimal destination assignment scheme.

6.5 ATM switches incorporating flow control

The absence of flow control within ATM switches ensures low latencies but complicates the design of the source modules that feed data to the event-builder.

Following the recent growth of interest of workstation and LAN manufacturers in ATM technology, the validity of the choice of input rate control rather than flow control has become the subject of much debate. A switch using internal flow control could provide loss-free cell delivery. However, internal flow control will lead to blocking effects under heavy loads, which may result in latency distributions with long tails. In fact, some experimental ATM switching fabrics incorporating flow-control schemes are now under development [25]. The RD31 project will continue modelling work in order to be able to make a comparative evaluation of the cost and performance of data acquisition system architectures based on switches employing these two different strategies.

7. SUMMARY

ATM is a statistical multiplexing technique that generally needs careful bandwidth management. ATM switches designed for telecommunications applications normally use no internal flow control and simply discard cells if congestion occurs. Therefore, when operating under the heavy loads and “many-to-one” traffic patterns characteristic of event building, a specific traffic shaping scheme is needed at the sources. The destination assignment scheme is also an important factor in traffic shaping. A specific destination assignment algorithm optimized to the switch architecture minimizes cell loss.

We have studied several source traffic shaping schemes involving input rate control, synchronized skewing of cell traffic between sources and randomization of cell injection times. The most promising scheme for large event builders optimizes the efficiency with which the available switch bandwidth is used through a combination of source rate control and randomization of cell injection times. Its distributed control architecture and avoidance of synchronization between different sources should simplify implementation and result in a robust system design.

A consequence of the traffic shaping scheme is that, because congestion is controlled, the throughput delivered to any given destination is independent of the switch size, and therefore it should be possible to expand the switch to large dimensions without

losing efficiency. The efficiency of utilization of the switch (and therefore the cost) can be traded against event-building latency and the required memory in sources and destinations.

Emerging commercial chip sets for building ATM interfaces contain the basic functions needed to build source interfaces with multiple queues and selectable queue servicing rates. The additional hardware required for traffic shaping is the randomization buffer and its controller.

If low event-building latency is important, the system should be designed so that the density of traffic at the inputs is low and queueing is avoided in the sources. If the load is light, traffic shaping may not be necessary and low event building latencies can be achieved. On the other hand, constructing an event builder from an ATM switch that supports an internal flow-control mechanism would guarantee lossless data transport even under high loads. In that case, the requirements for traffic shaping could be relaxed, but probably at the expense of a long tail to the latency distribution.

Acknowledgements

The concept of using packet switching network technology for event building, which led to the RD31 project, was suggested by A. Marchioro; W. von Rüden assisted in setting up the collaboration and D. Samyn suggested the use of the concurrent C++ language for modelling and gave invaluable support with the tool. The Afterburner hardware was developed by D. Banks, C. Calamvokis and M. Prudence, and the software by C. Dalton and A. Edwards. B. Traw and J. Smith from the University of Pennsylvania are designing the ATM adaptor for the Afterburner.

For many informative discussions we thank our collaborators of the RD31 project; K. Agehed, A. Eide, P. Gomes, L. Gustafsson, S. Hultberg, T. Lazraq, Th. Lindblad, M. Mota and H. Tenhunen. Other colleagues whom we wish to acknowledge for their contributions through discussion and information exchange are: B. Brunner, R. Bock, J. Bovier, B. Carpenter, J. Carter, S. Cittolin, J-P. Dufey, J. Joosten, L. Mapelli, G. Mornacchi, C. Petitpierre and A. Wiesel.

Thanks are also due to M. Bénard of Hewlett Packard, Geneva for support through the Hewlett Packard Research Grants Programme (Europe). We appreciate the continued support given to our work by the CERN management, in particular P.G. Innocenti and F. Bourgeois.

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] 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, Conference record, Vol. 2 (1991) 833-837.
- [4] W. Bozzoli et al., High performance event distribution using HIPPI, Proc. of the International Conference on Computing in High-Energy Physics, Annecy, France, CERN 92-07 (September 1992) 192-195.
- [5] W. Greiman, S. Loken and C. McParland, Use of commercial gigabit data switches for SSC and LHC event builders, Proc. of the International Conf. on Computing in High-Energy Physics, Annecy, France, CERN 92-07 (September 1992) 184-187.
- [6] U. Behrens et al., The event builder of the ZEUS experiment, Proc. of the International Conference on Computing in High-Energy Physics, Annecy, France, CERN 92-07 (September 1992) 162-166.
- [7] International Telecommunication Union, Geneva, Switzerland, CCITT recommendations I.150, I.211, I.311, I.321, I.327, I.361, I.362, I.363, I.413, I.432 and I.610.
- [8] 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 (1992).
- [9] The ATM Forum, c/o Interop Inc., 480 San Antonio Road, Suite 100, Mountain View CA94040-1219, USA, ATM user network interface specif., Version 2.0 (June 1992).
- [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) 156-165.
- [11] M. Henrion et al., Technology, distributed control and performance of a multipath self-routing switch, Proc. of the XIVth International Switching Symposium, Yokohama, Japan, Vol 2 (October 1992) 2-6.
- [12] International Telecommunication Union, Geneva, Switzerland, CCITT recommendations G.707, G.708 and G.709.
- [13] D. Clark, V. Jacobson, J. Romkey and H. Salwen, An analysis of TCP processing overhead, IEEE Communications Magazine (June 1989) 23-29.
- [14] 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).
- [15] V. Jacobson, Efficient protocol implementation, presented at ACM SIGCOMM '90 tutorial (September 1990).
- [16] C.B.S. Traw and J.M. Smith, A high-performance host interface for ATM networks, Proc. SIGCOMM '91, Zurich, Switzerland (September 1991) 317-325.
- [17] Creative Electronic Systems SA, Geneva, Switzerland, RIO 8260 and MIO 8261 RISC I/O Processors — User's Manual, Version 1.0 (1992).
- [18] Creative Electronic Systems SA, Geneva, Switzerland, RAID 8239 RISC Processor User's Manual, Version 1.0 (1992).
- [19] 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 (1991-1993).

REFERENCES (cont'd)

- [20] Creative Electronic Systems SA, Geneva, Switzerland, private communication.
- [21] Transwitch Corp., 8 Progress drive, Shelton CT 06484, USA, SARA chipset, technical manual (November 1992).
- [22] P.A. Buhr et al., μ C++: concurrency in the object-oriented language C++, *Software – Practice and Experience*, Vol. 22(2) (February 1992) 137–172.
- [23] 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, Session 5.23 (1991) 1–8.
- [24] 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 (1990–1993).
- [25] V. Kumar et al., Phoenix: a building block for fault-tolerant broadband packet switches, *Proc. of the IEEE Global Telecommunications Conference*, Phoenix, USA (December 1991) 228–233.