CERN-LHCC-97-5

SCP

EUROPEAN ORGANIZATION FOR NUCLEAR RESEARCH

CERN / LHCC / 97-05
LCB Status Report / RD-31
21 January 1997

RD-31 Status report '97

NEBULAS: High performance data-driven event building architectures based on asynchronous self-routing packet-switching networks

C. Bizeau, M. Costa, J.-P. Dufey[1], M. Letheren[1], A. Marchioro, A. Pacheco, C. Paillard
CERN, Geneva

D. Calvet, K. Djidi, P. Le Dû, I. Mandjavidze
CEA DSM/DAPNIA, Saclay

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

A. Manabe, Y. Nagasaka, M. Nomachi[2], O. Sasaki, Y. Watase
National Laboratory for High Energy Physics (KEK), Japan

P. Sphicas, S. Sumorok, S. Tether
Massachusets Institute of Technology, Cambridge, USA

K. Agehed, S. Hultberg, T. Lazrak, Th. Lindblad, C. Lindsey, H. Tenhunen
The Royal Institute of Technology (KTH), Stockholm

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

M. Benard[3]
Hewlett Packard, Geneva

[1] Joint spokespersons.
[2] Research Center for Nuclear Physics (RCNP), Osaka University, Japan
[3] Research Grants Programme, HP European HQ, Geneva.

TABLE OF CONTENTS

1. EXECUTIVE SUMMARY

The goal of the RD-31 project is to demonstrate high-performance, parallel event building architectures that can satisfy the requirements for the level-2 and level-3 trigger systems of the LHC experiments. These architectures can be constructed around commercial parallel, multi-way switching fabrics. Many industrial switching fabrics are now available for switching traffic in broadband telecommunications networks or local area networks based on the Asynchronous Transfer Mode (ATM) standard. Event building architectures based on this technology have been studied extensively by RD-31.

RD-31 was approved in November 1992 and the last status report was presented in March 1995. The LHCC assigned as milestones the tasks (1) "Extend the simulation of event building protocols to include congestion control and high-level read-out protocols using physics data" and (2) "Implement a system to build events from multiple data sources to multiple processor destinations using existing ATM switches". A close collaboration between RD-31 and ATLAS/CMS was encouraged for detailed simulation studies of the baseline designs and to allow reuse of expensive test equipment.

Event builder demonstrator hardware and software developments: The development of a VME-ATM interface, based on commercially available chip sets which implement the ATM protocols, has been completed. This interface operates correctly with standard equipment and achieves full performance. It is used as destination module in the event builder demonstrator. Data generators have been developed to implement the source modules and provide a simple way to emulate various traffic shaping techniques. The software protocol layers and management functions, required for event building, have been improved and tested on the ATM interface. An ATM-PCI mezzanine card has been developed and tested on PCs running Windows NT and on VME processors running LynxOS. A new generation of commercial VME computer boards based on the PCI bus and equipped with a commercial ATM-PCI mezzanine card have been evaluated in view of the implementation of "intelligent" source memories and destination processors.

Several event builder demonstrators have been assembled, based on a telecom switch from Alcatel and a LAN switch from Bell Labs. They have been used to evaluate various traffic shaping techniques in push architecture. The main result is that high throughput has been achieved on an 8 X 8 event builder demonstrator with no data loss, by using a traffic shaping technique (rate division) that is provided as a standard in the ATM adapters. Simulation by software of larger systems (up to 256 X 256) confirms that high loads with very low data loss probability are possible. The interconnection of both switches has given interesting results on the maximum performance achievable and has been used to emulate the traffic of a 64 X 64 switching fabric. Preliminary results with pull architecture have been obtained and the implementation of a complete demonstrator is under way.

Simulation: An important issue in using switching fabrics for event building is how to control the traffic patterns so as to avoid internal congestion. An in-depth investigation of congestion control by the so called "traffic shaping" techniques has been conducted. Our software models have been tuned and validated by using the results obtained on event builder demonstrators. A very good agreement has been reached and has enhanced the confidence in the extrapolations to large systems.

The simulation of data acquisition protocols has been devoted mainly to an investigation of a "pull" architecture, where the destinations play an active role and collect the data selectively. Detailed studies of this method have been performed for the proposed ATLAS Trigger/DAQ "Architecture C".

Continuation of work: We believe that the goals assigned to RD-31 as a generic R&D project have been achieved. We hope that the expertise gained so far will be invested in the implementation of more advanced demonstrators. Proper sharing of information, expertise and, possibly, resources between the different projects is preferable to duplication of effort within competitive developments. In this respect, several of the groups involved in RD-31 are ready to participate in common developments and to exchange information on a regular basis. Finally close contacts with industry and researchers outside of the LHC community have been extremely valuable and should be continued.

2. INTRODUCTION

The RD-31 proposal [1] was originally approved on 26 November 1992, a first status report to the DRDC was presented in January 1994 [2]. A second status report to the LERB was presented in March 1995 [3]. Considering the new orientation induced by the approval of the LHC project and the approval of the first two experiments, ATLAS and CMS, it was agreed that RD-31 would present a final status report towards the end of 1996.

A group from the National Laboratory for High Energy Physics (KEK), Japan, has joined the collaboration. Some further changes in individual collaborators are reflected by the updated list of signatures on the cover page.

We recall here the milestones set by the LHCC for the last phase of the project:

- Extend the simulation of event building protocols to include congestion control and high-level readout protocols using physics data.
- Implement a system to build events from multiple data sources to multiple processor destinations using existing ATM switches.
- A close collaboration between RD-31 and ATLAS/CMS was encouraged for detailed simulation studies of the baseline designs and to allow reuse of expensive test equipment.

A synthesis of the most important results obtained by RD-31 has been presented recently at the International Workshop on Data Acquisition Systems (DAQ96) in Osaka [4]. It is included in this report (Appendix 1). The article published in the IEEE Trans. on Nucl. Science (NSS 95 issue) [5] describes the more basic aspects of ATM technology and the principles of its application to event building and gives technical details; it is reproduced in Appendix 2. Finally our contribution to the 2nd Workshop on Electronics for LHC experiments (Hungary, September 1996) [6] which summarizes our views on the larger systems is also part of the present report (Appendix 3). An ATM tutorial [7], the B-ISDN standards [8] and the ATM standard [9] can be consulted by the interested reader.

3. SUMMARY OF ACTIVITIES

The main activities and achievements since the previous status report were:

- Achievement of high throughput (reception) on the ATM interface developed by RD-31. Three units in total have been manufactured.
- Design, implementation and manufacturing of 8 VME based traffic generators controlled by a VME-based PC.
- Control software for the traffic generators and program to generate the data emulating various traffic shaping schemes.
- Set-up of event building demonstrators with the traffic generators as sources, and the RD-31 ATM interfaces as destinations. (Successively 2 X 2, 4 X 4 and 8 X 8 event builders operated in push mode).
- Investigation of various traffic shaping techniques on the demonstrator: the *true barrel shifter*, the *randomizer* and the *rate division*.
- Test of a pull architecture on the demonstrator (6 sources, 1 destination).
- Design and test of a PMC-ATM network interface card.
- Design and implementation of ATM device drivers (conventional and “zero copy”) for a PCI-ATM interface based on IDT NicStar chip under Lynx-OS and Windows NT.

- Performance measurements of the PCI-ATM board on the CES RTPC and RIO2 boards and on a Pentium PC.
- Tests of performance under high load using a two stage network of 8-port switches (Alcatel and Bell Labs) in order to investigate very high loads and 64 X 64 configurations.
- Implementation of a source data module with data transfers occurring simultaneously on VME (input) and ATM (output to the network). Performance measurements.
- Study of large event builders (up to 256 ports) by means of software models previously validated and calibrated on the demonstrators.
- Study of pull architecture in view of an application to level 2 trigger in ATLAS using Regions Of Interest (ROIs).

4. CONTINUATION OF WORK

Our confidence in the feasibility of ATM based event building has been enhanced. Cross checks of results on other equipment and larger configurations are needed. Pull architecture still needs to be demonstrated, in particular in view of an application for phased event selection.

Several projects are now under way to implement ATM-based demonstrators: the ATLAS DAQ "Prototype -1" and the CDF upgrade done in collaboration with CMS. The RD-31 "Atlas team" (Saclay) will continue the architecture design and simulation studies for the proposed Atlas trigger/DAQ "Architecture C". They will develop their demonstrator in conjunction with the development of an "intelligent", flexible dual-port source memory that can support the required data flows. RCNP (Osaka, Japan) plans to evaluate event building on larger systems by using its new ATM-based network.

The implementation of an ATM-based event builder in an experiment would be a challenging project. This would be the occasion to investigate methods of management and control of the event builder in order to ensure its maximum availability.

The Phenix experiment at RICH is evaluating ATM for its event builder and RD-31 has acted as a consultant in a first phase. Further consulting is likely to be solicited.

We believe that the goals assigned to RD-31 as a generic R&D project have been achieved. We hope that the expertise gained so far will be invested in the implementation of more advanced demonstrators. Proper sharing of information, expertise and, possibly, resources between the different projects is preferable to duplication of effort within competitive developments. In this respect, several of the groups involved in RD-31 are ready to participate in common developments and to exchange information on a regular basis. Finally close contacts with industry and researchers outside of the LHC community have been extremely valuable and should be continued.

5. LIST OF PUBLICATIONS

5.1 Status reports

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.

J. Christiansen et al., "NEBULAS - A high performance data-driven event building architecture based on an asynchronous self-routing packet-switching network", CERN / DRDC 93-55.

M. Costa et al., "NEBULAS - A high performance data-driven event building architecture based on an asynchronous self-routing packet-switching network", CERN / LHCC/95-14.

5.2 Conference Proceedings, Journals, etc.

M. Letheren et al.,"An Asynchronous Data-Driven Event Building Scheme based on ATM Switching Fabrics", *IEEE Trans. on Nuclear Science*, vol. 41, No 1, Feb. 1994. Also available as CERN / ECP 93-14.

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", in *Proceedings of Open Bus Systems 1993*, 29-30 November 1993, Munich, Germany. Also available as CERN/ECP 93-24.

J. Christiansen et al., "The NEBULAS Project: A study of ATM-based Event Building for Future High Rate Experiments", presented at the Real-time Data Conference, Moscow, 1994.

I. Mandjavidze, "Review of ATM, Fibre Channel and Conical Network Simulations", *Proceedings of the 1st International Data Acquisition Conference*, Oct. 26-28, 1994, Fermilab, Batavia, Il, USA.

I. Mandjavidze, "Software Protocols for Event Builder Switching Networks", *Proceedings of the 1st International Data Acquisition Conference*, Oct. 26-28, 1994, Fermilab, Batavia, Il, USA.

A. Marchioro, I. Mandjavidze, "Pros and cons of Commercial and Non-Commercial Switching Networks, in *Proceedings of the International Data Acquisition Conference*, Fermilab, Oct. 1994.

L. Gustafsson et al., "A 155 Mbit/s VME to ATM interface with special features for event building applications based on ATM switching fabrics", in *Proceedings of the International Data Acquisition Conference*, Fermilab, Oct. 1994.

M. Costa et al., "An ATM-based event builder test system", *Proceedings of the First Workshop on Electronics for LHC Experiments*, pp 340-344, Lisbon, Sept. 11-15, 1995.

M. Costa et al., "Results from an ATM-based Event Builder Demonstrator", in *IEEE Trans. on Nucl. Sciences*, Vol. 43, No. 3, pp 1814, 1820, June 1996.

D. Calvet et al., "A Study of Performance Issues of the ATLAS Event Selection System based on an ATM Switching Network", in *IEEE Transactions on Nuclear Science*, vol. 43, No 1, pp. 90-98. February 1996.

M. Letheren, "Architectures and Technologies for Data Acquisition at the LHC Experiments", invited talk, in *Proceedings of the 2nd Workshop on Electronics for LHC Experiments*, pp 125–136, Balatonfüred, Hungary, September 23–27, 1996.

M. Costa et al., "Lessons from ATM-based Event Builder Demonstrators and Challenges for LHC-scale Systems", in *Proceedings of the 2nd Workshop on Electronics for LHC Experiments*, pp 208–214, Balatonfüred, Hungary, September 23–27, 1996.

C. Bizeau et al., "On the Integration of High Performance ATM-based Event Builders", in *Proceedings of the NSS 96*, Anaheim Nov. 3-9, 1996.

M. Letheren, "An overview of Switching Technologies for Event Building at The Large Hadron Collider Experiments", *Proc. of the 2nd International Data Acquisition Workshop (DAQ96)*, RCNP, Osaka, 13–15 Nov. 1996.

C. Bizeau et al., "On the Feasibility of High Performance ATM-based Event Builders", in *Proc. of the 2nd International Data Acquisition Workshop on (DAQ96)*, Nov. 13-15, 1996, RCNP, Osaka, Japan.

D. Calvet et al., "Performance Analysis of ATM Network Interfaces for Data Acquisition Applications", *Proc. of the 2nd International Data Acquisition Workshop (DAQ96)*, RCNP, Osaka, 13–15 Nov. 1996.

D. Calvet et al., "Evaluation of a Congestion Avoidance Scheme and Implementation on ATM Network based Event Builders", *Proc. of the 2nd International Data Acquisition Workshop (DAQ96)*, RCNP, Osaka, 13–15 Nov. 1996.

D. Calvet, F. Servaz., "Design of a PMC/ATM Interface", *Proc. of the 2nd International Data Acquisition Workshop (DAQ96)*, RCNP, Osaka, 13–15 Nov. 1996.

5.3 RD-31 Notes

RD-31 93-03 L. Gustafsson, "Evaluation of different ATM test tools to be used in an ATM switch demonstrator system", IRS, Uppsala, February 1993.

RD-31 93-06 I.Mandjavidze, "Modelling and performance evaluation for event builders based on ATM switches", December 1993.

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

RD-31 94-01 K. Agehed et al., "Progress Report on the Design and Performance of a VME-ATM Module using Dual-ported Memories", KTH, Stockholm, January 1994.

RD-31 94-02 I. Mandjavidze, "Data Flow Protocol Overhead in an ATM based Event-Builder", CERN, January 1994.

RD-31 94-03 I. Mandjavidze, "A new traffic shaping scheme: the true barrel shifter", February 1994.

RD-31 94-04 I. Mandjavidze, "Modelling of the CMS Virtual Level 2", CERN, August 1994.

RD-31 94-05 J. Christiansen et al., "The NEBULAS Project: A study of ATM-based Event Building for Future High Rate Experiments", presented at the Real-time Data Conference, Moscow, 1994.

RD-31 94-06 A. Marchioro, I. Mandjavidze, "A data-driven event building scheme based on a conic self-routing packet-switching Banyan network".

RD-31 94-07 D. Calvet, "A MODSIM Model of the AT&T Phoenix switching Fabric", Aug. 1994.

RD-31 94-08 M. Costa, "ATM Event Building Software", December 1994, revised February 1995.

RD-31 94-09 T. Lazraq et al., “ATM traffic shaping in event building applications”

RD-31 94-11 L. Gustafsson et al., “A 155 Mbit/s VME to ATM interface with special features for event building applications based on ATM switching fabrics”, also In *Proceedings of the International Data Acquisition Conference*, Fermilab, Oct. 1994.

RD-31 94-12 A. Marchioro, I. Mandjavidze, “Pros and cons of Commercial and Non-Commercial Switching Networks”, also in *Proceedings of the International Data Acquisition Conference*, Fermilab, Oct. 1994 (to be published).

RD-31 95-01 M. Costa et al., “Randomizer Protocol”, February 1995.

RD-31 95-02 J.-P. Dufey, “Problem statement for Fibre Channel event builder modelling”, January 1995.

RD-31 95-03 I. Mandjavidze, “Modelling of an ATM implementation of the CMS Virtual Level 2 Architecture”, February 1995.

RD-31 95-04 C. Paillard, “An STS-OC3 SONET/ STM-1 SDH ATM Physical layer implementation and Application to an ATM Data Generator”, February 1995.

RD-31 95-05 M. Costa, “An ATM based Event Building test system using ATM traffic generators”.

RD-31 95-06 S Tether, "Implementation of a C++ based Simulation Package", February 1995.

5.4 Other publications

M. Letheren et al., "Switching Techniques in Data Acquisition Systems for Future Experiments", CERN School of Computing, Arles, France, 20 August-2 September 1995, CERN 95-05.

M. Costa et al., “ATM-based event building”, ATLAS Internal Note, DAQ-NO-024, December 1994.

M. Costa, “Application of ATM to Event Builders for High Energy Physics Experiments: a demonstrator system”. Travail de fin de stage du cours “Communication Networks”, Ecole Polytechnique Federale, Lausanne, Mars 1995-Mars 1996.

5.5 WEB site

http://www-rd31.cern.ch/rd31/rd31.htm

6. REFERENCES

[1] 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.

[2] J. Christiansen et al., NEBULAS - A high performance data-driven event building architecture based on an asynchronous self-routing packet-switching network, CERN / DRDC 93-55

[3] M. Costa et al., "NEBULAS - A High Performance Data-driven Event Building Architecture based on an Asynchronous Self-routing Packet-switching Network", *CERN/LHCC/95-14*, CERN, Geneva, March 1995.

[4] C. Bizeau et al., "On the Feasibility of High Performance ATM-based Event Builders", in *Proc. of the 2nd International Data Acquisition Workshop on (DAQ96)*, Nov. 13-15, 1996, RCNP, Osaka, Japan.

[5] M. Costa et al., "Results from an ATM-based Event Builder Demonstrator", *IEEE Trans. on Nucl. Sciences*, Vol. 43, No. 3, pp 1814 -1820, June 1996.

[6] M. Costa et al., "Lessons from ATM-based Event Builder Demonstrators and Challenges for LHC-scale Systems", in *Proceedings of the 2nd Workshop on Electronics for LHC Experiments*, pp 208–214, Balatonfüred, Hungary, September 23–27, 1996.

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

[8] International Telecommunications Union, ITU, Geneva; recommendations I.150, I.211, I.311, I.321, I.327, I.361, I.362, I.363, I.413, I.432, I.610.

[9] The ATM Forum, "ATM User Network Interface Specification", Version 3.0.

APPENDIX 1

On the Feasibility of High Performance ATM-based Event Builders

C. Bizeau, M. Costa, J.-P. Dufey, M. Letheren, A. Pacheco, C. Paillard
CERN, 1211 Geneva 23, Switzerland
D. Calvet, P. Le Dû, I. Mandjavidze
CEA Saclay, 91191 Gif-sur-Yvette CEDEX, France
M. Weymann, A. Wiesel
Creative Electronic Systems, Geneva, Switzerland

ABSTRACT

It is shown that an ATM switch with 8 ports can efficiently support the traffic generated by event building applications. Using only standard traffic control techniques, a bandwidth utilization of nearly 100% can be reached, without data loss. Extrapolations of the feasibility of larger systems by software models is encouraging. Source and destination modules can be implemented on commercially available components and good performance can be achieved in terms of throughput and event building rate.

1. Introduction

ATM is an attractive technology for the implementation of high data rate event builders. However, some fears have been expressed regarding the fact that ATM networks are prone to congestion and consequently to data losses and that no safe data transport protocol is provided by the ATM standard. It should be pointed out that, when the ATM standard has been defined, it has been a deliberate choice not to impose guaranteed data delivery because certain services (voice, video) cannot tolerate the delays and jitters introduced by secure transport protocols, whereas they are not sensitive to occasional data losses. If required, this security must be provided in a protocol layer above the standard ATM layers.

We show that, for event building applications, the traffic can be regulated by standard methods in such a way that the ATM network is reliable even at high loads, and that a transport protocol is not required. To confirm this we have made tests on small scale demonstrators and we have used software models to study larger configurations.

ATM provides high performance for small as well as large packets; on an STM-1 link (155 Mbit/s), single cells carrying 48 bytes of user information can be delivered every 2.7 µsec, corresponding to a maximum frequency of 370 KHz. This capability of ATM allows new architectures for event builders: the events can be built individually, independently of the event fragment size and it is possible to use the switching network to transport the event builder's control protocol messages. Consequently systems can be designed in which partial event building is used in a phased event selection process [1].

We have studied the problem of matching the performance of the various event builder components, the source and destination modules and their interfaces to the switching network, in order to take advantage of the intrinsic performance of ATM. High performance is achievable in the network interface if a modest effort is invested in the development of drivers. We have also made measurements of throughput of a source when VME traffic is routed through to the ATM network; the performance is found to be adequate for many of the event builders required by experiments planned for the near future.

Section 2 of this paper discusses the performance requirements of a generic event builder system and identifies the critical points. Section 3 deals with the performance of ATM adapters and the effect of software overheads due to the operating system and the event builder protocol. Sections 4 and 5 give results from measurements on small scale demonstrators and extrapolate to larger switches. The performance of a source module based on commercial components is presented in Section 6.

The basics of ATM technology, the principles of ATM-based event building and details of the hardware and software used in the demonstrators can be found in [2].

2. Generic Event Builder Model and Performance Requirements

The data flow structure of a generic event builder is presented in Figure 1. A complete event building system, in its simplest form, the "push" architecture, requires that data are collected from one or several Read-Out Buffers (ROB) in the sources before being sent as an event fragment of suitable size through the network via the Network Interface (NI). The destination, in turn, submits the full event for analysis to a processor in a farm, or a symmetric multiprocessor with shared memory. In the backwards direction (destination to source) only control messages are sent, in particular when point to point flow control and reception acknowledgements are required (transport protocol).

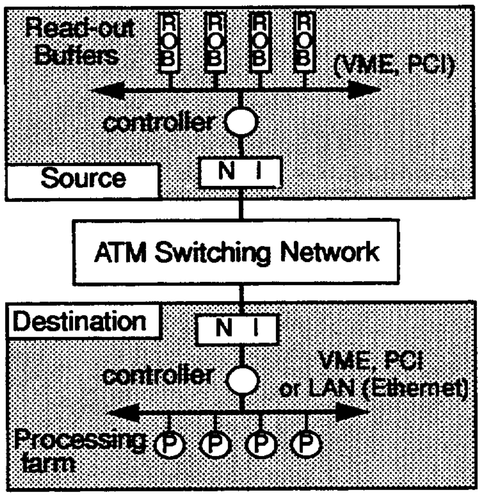


Figure 1: Data flow structure of a generic event builder.

In the "pull" architecture, the destination initiates the transfer of data by the sources, one of the processors in the farm being in charge of the selection and/or analysis of an event. The control messages issued by this processor to request data from the sources are routed via the switching network.

We briefly analyse the requirements for the critical elements of this architecture. Results from our investigations of these issues will be given in the following sections.

2.1 *The NIs (Network Interfaces)*

The user network interface gives access to the network resources and provides adaptation to the ATM standard. It is under the control of the user application which implements the higher layers of the protocol stack. The submission or reception of a data packet by the controller generates overheads due to the adapter control software (driver). Additional overheads are due to the event building protocol and, if guaranteed data delivery is required, to the transport protocol.

2.2 *The Switching Network*

The switching network routes the main data streams and possibly interleaves control messages. Depending on the traffic pattern, it may be necessary to reduce the load to avoid congestion. However, as will be explained in the section 4, an efficient traffic shaping scheme permits high load with very low probability of congestion.

2.3 *The Source and Destination Modules*

The accretion of data from several ROBs in the source module is desirable in order to use efficiently the network: if the link offers a user data throughput of T [Mbyte/s] then the trigger rate f [KHz] determines the maximum size of an event fragment F [KByte], namely F = T / f (with a minimum value F = 48 bytes, determined by the fixed length of an ATM cell). For example, an STM-1 link (155 Mbit/s) provides an effective user data bandwidth of 16.8 MByte/s and, for a trigger rate of 1 KHz, the event fragment size can be up to 16.8 KByte. It is desirable to use as much as possible of the link bandwidth by aggregating data to a value close to the maximum. The achievable fragment size may be lower if a load limitation is imposed to avoid congestion in the switching network. In any case it is inadvisable to operate the link at 100% load, even in a network with perfect traffic control.

In a source, a general requirement for the bus linking n ROBs is that its throughput, measured for sub-fragments of size ~ F/n, should be at least equal to the network link throughput. This is quite a difficult condition to meet when the sub-fragment size is a few hundred bytes. This problem of accretion is not specific to ATM. A destination module receives fragments with a frequency f and distributes events to the farm of processors at a frequency f/N, N being the number of destinations. The event size is of the order of N*F. The link with the processor farm must offer a bandwidth of the same order as the network link, but the required high bandwidth efficiency does not need to be achieved with small packets.

3. Performance of the ATM User Network Interfaces

The ATM standard includes adaptation layers for the different services. For data transfer services, the adaptation protocol layer, called AAL5, is defined for blocks with variable size, up to 64 KByte. On the transmission side, the AAL5 protocol specifies that a trailer with a CRC is added and that the data block is segmented in fixed size ATM cells. Reassembly of the original data block and CRC check are performed on the receiving side. The ATM layer is in charge of the cells and routes them according to their virtual connection identifier. Many virtual connections can be active simultaneously in a single NI. At reception, cells are sorted out according to the virtual connections so that reassembly at AAL5 level can occur. ATM does not provide a transport protocol. Corrupted packets are detected and signalled, but retransmission is not performed because it is not required in every application. If needed, it has to be provided in the upper layers.

Commercial chip-sets provide hardware implementations of the ATM and AAL5 protocol layers. The complex operations of routing, segmentation and reassembly are performed in a very efficient way with negligible overheads.

3.1 Software

The ATM user network interface is controlled by a processor which runs the software to submit or receive packets at AAL5 level. The overhead due to these basic operations is small but, in the presence of an operating system, a copy of the data between the user space and the kernel space usually takes place and can substantially increase the overhead.

Additional overheads originate in the higher layers, on top of the ATM and AAL5 layers: the optional *transport protocol* layer and the *event building protocol* which provides for the identification of event fragments, their assembly into events, and determines when an event is completed. A short description of our implementation of the event building protocol layer and the data structures can be found in [2].

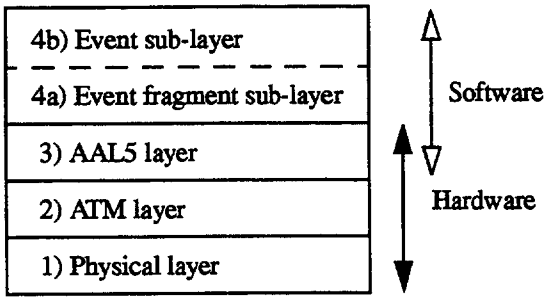


Figure 2: Protocol Layer Structure of the event builder architecture

Figure 2 shows the layered structure of the protocols: layers 1 to 3 are the standard physical (SONET/SDH), ATM and AAL5 layers [3]. We use the AAL5 protocol to transfer the event fragments and, in the case of pull architecture, to carry the request messages as well.

A *transport protocol* can be inserted between layers 3 and 4, to take care of retransmission of lost or corrupted packets. It has been proposed to use TCP/IP in order to ensure lossless data transmission. As a de-facto standard TCP/IP simplifies considerably the interfacing of applications with the switching network. However, this choice has several drawbacks: TCP/IP has been implemented for low speed networks and gives poor performance on high speed links such as the ATM STM-1 at 155 Mbit/s, especially for small data packets. It does not help to reduce the probability of congestion in the switching network and it requires a non-trivial tuning of the communicating systems in order to avoid dead lock situations. We have found that a suitable traffic shaping mechanism avoids data loss (more precisely that the probability of data loss is very low) under normal operation of the event builder (see next section) and that consequently we can omit the transport protocol.

The event building layer is subdivided in 2 sub-layers (4a and 4b). Sub-layer 4a, the *Event Fragment Sub-layer,* ensures the independence of the sub-layer 4b from the specific network layers and allows to define a maximum packet size independently from the AAL5 standard. The *Event Sub-layer* has the task of linking together the event fragments to form an event. The event sublayer is able to build several events concurrently. Its task is also to recognize when an event is completely assembled.

3.2 Performance of the ATM Interfaces

Figure 3 shows performance measurements made on a 155 Mbit/s ATM interface based on the SARA chip set from Transwitch [4] and with a 25 MHz MIPS R3000 processor used as controller. In order to minimize the overheads, the event building software runs in stand alone mode, with a minimal kernel and the destinations poll for the arrival of new packets rather than waiting for interrupts. Curve a) is the maximum throughput achieved when transferring AAL5 packets, in the absence of the event building layers of the protocol. Bandwidth saturation (149 Mbit/s at ATM level) occurs for a packet size above 240 bytes, which corresponds to an overhead of 16 µsec per AAL5 packet. Curve b) shows the throughput when the event building layer is active, measured on the receiving side where the overhead is greatest. The saturation occurs at a packet size of 400 bytes, corresponding to an overhead of 25 µsec. The frequency at which packets can be received

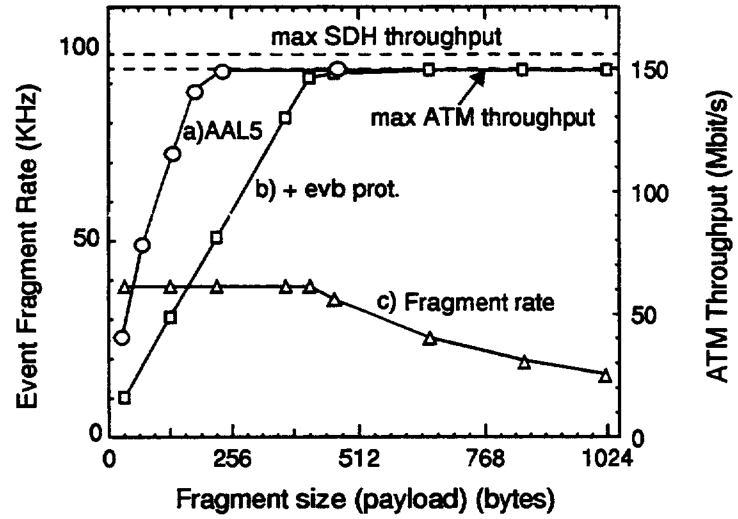


Figure 3: Performance of the ATM interface (RD31),
No operating system; with event building protocol

(Curve c) is determined by the data transfer time, but below the saturation point the software overheads dominate and limit the maximum frequency to 38 KHz.

We have evaluated the performance of PCI-ATM interfaces based on the NicStar chipset from IDT [5]. One of them is a PCI mezzanine card (PMC) from CES [6] on a RIO2 board which uses a 100 MHz PowerPC 604 [7] running LynxOS; the other one has been developed at Saclay as a PMC board [8] and its performance has been measured on a PC running Windows NT. In both cases a conventional driver and a "zero copy" driver have been developed with the aim to compare their performance at AAL5 level. In the case of the "zero copy" driver, the overhead per AAL5 packet, in transmission mode, is 10 µsec up to 4 MByte, plus 2 µsec for each additional 4 MByte packet. In receiving mode these figures are respectively 18 µsec and 5 µsec. The event building protocol is not included. A full description of this development is given in a separate paper presented at this workshop [9].

4. Performance of the 8 X 8 Event Builder Demonstrator

The experimental set up is shown in Figure 4. Two switches have been used: one from Alcatel and one from Bell Labs. The 8 X 8 Alcatel switch [10] of "telecom" type has no internal flow control: it resolves contention by means of internal buffering, both in the switching elements and in the network interfaces, and through internal bandwidth expansion provided by multiple paths. The 8 X 8 switching fabric from Bell Labs is based on the Phoenix 2 X 2 switching element which implements internal flow control by means of back pressure [11]. This mechanism holds up the traffic arriving into an internal switching element as soon as a certain level of buffer occupancy has been reached. Back pressure

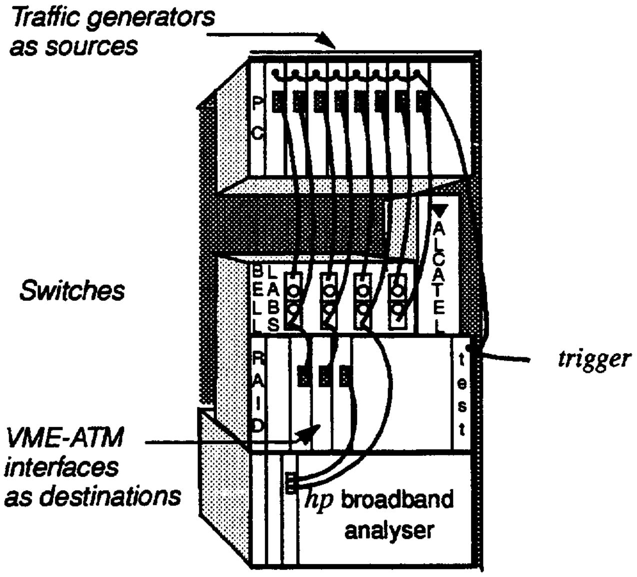


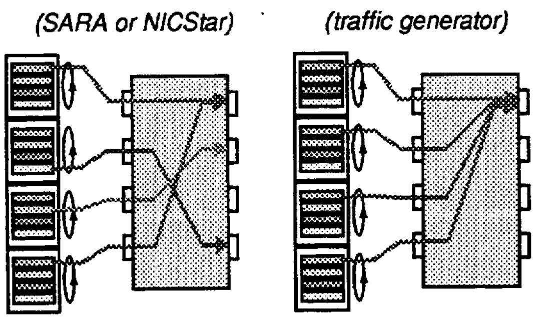
Figure 4: Event builder demonstrator test bed

cannot propagate outside of the switch. As a consequence, the input network interface is the place where buffer overflow can occur and it is equipped with a relatively large memory (85 cells). In addition the internal bandwidth is expanded by a factor 2.

As source modules we use the traffic generators that we have developed [12]. Implemented as VME modules they include the SONET/SDH physical layer of the ATM standard protocol (at 155 Mbit/s) and a 1 Mbyte memory in which the traffic to be emitted is downloaded in the form of pre-formatted ATM cells. A general purpose program can generate the ATM cell streams off-line, including the AAL5 structure, on the basis of specifications of the traffic pattern required. Various traffic shaping schemes can be emulated. The generators can operate in a continuous mode, without software control, and can achieve full bandwidth. All generators can be started simultaneously by means of an external signal daisy-chained between them. As destination module we use the first ATM adaptor described in the previous section.

For the push architecture, in the absence of a transport protocol, the traffic is unidirectional and it is possible to implement an 8 X 8 event builder on an 8-port switch by connecting at each port a source to the transmit line and a destination to the receive line (each port is full duplex and has one transmit and one receive line). Every source is linked to all destinations by virtual connections that are set up permanently. Although all destination ports receive data, only 1 needs to be equipped with an actual interface for measurement purposes.

Events are collected individually in a destination: no packing in "super-events" is performed. The destination is assigned in round robin fashion to successive events. The event fragments are either of fixed length, the same in all sources, or follow a normal distribution with a variance proportional to the mean value.



a) uncorrelated sources

b) completely correlated sources

Figure 5: Rate division traffic shaping

The ATM standard defines various types of traffic that can be selected for a virtual connection (VC). We have adopted CBR (Constant Bit Rate): in the case of a symmetric N X N event builder a constant rate, equal to 1/N of the physical link rate, is assigned to every VC. This ensures that the average aggregate bandwidth arriving at a destination does not exceed the nominal bandwidth available at the output port. The implementation of CBR is made in the SAR (Segmentation And Reassembly) chip. The mechanism is illustrated in Figure 5 for the case of 4 sources and 4 destinations: in the source adapter, a queue per destination (i.e. per VC) is maintained. These queues are scanned in a round robin fashion, 1 cell being extracted from each queue each time. The sources are not correlated in time because each adapter has its own clock (Fig. 5a). This *rate division* scheme combined with the random correlation between the sources acts as traffic shaping that smoothes the effect of the trigger. The hardware implementation of CBR in the SAR chip does not produce any additional overhead.

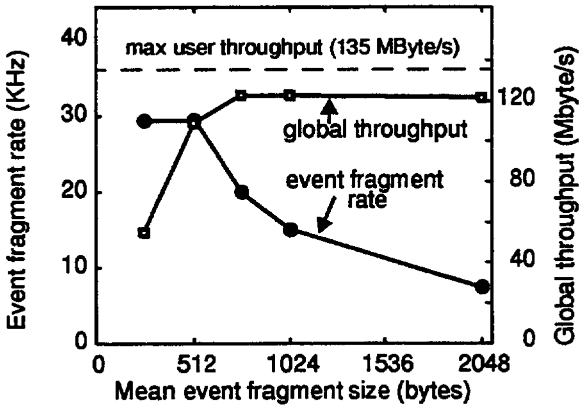


Figure 6: 8x8 event building; push data flow; gaussian size distribution (σ^2=30%)

Figure 6 shows the measured performance of the Bell Labs switch with 8 traffic generators as sources. The event fragments have variable sizes normally distributed with σ^2=30% of the mean value. The maximum reachable aggregate throughput (100% load) for user data is approximately 135 MByte/s, after removal of the overheads at the physical, ATM and AAL5 layers. We reach 120 MByte/s (i.e. 89% load). The data in the traffic generator's memory cannot be filled in an efficient way for variable length event fragments and this accounts for the slightly reduced performance.

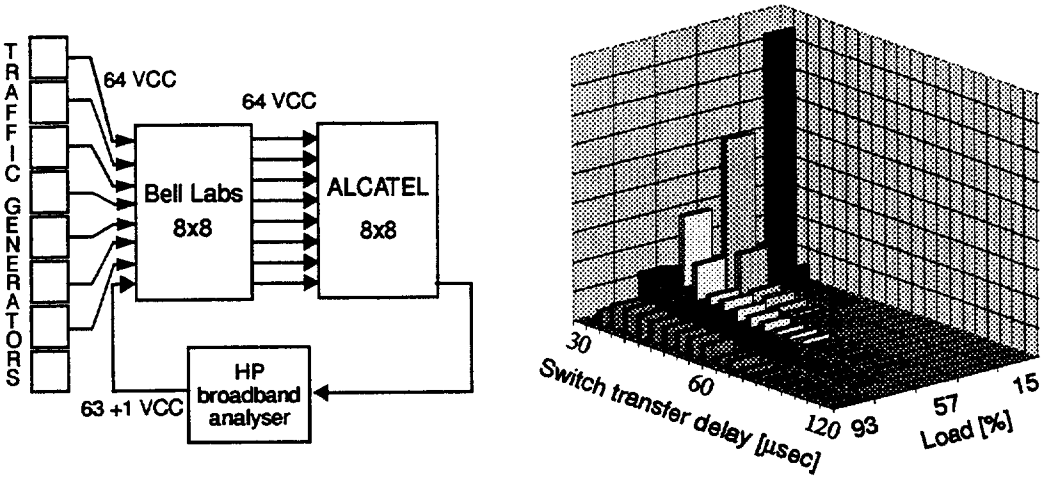
We have also investigated the worst case, where all sources are exactly synchronized, as shown in Figure 5b), and where a wave front of N cells is sent to the same destination every N time slots (a time slot is the time to transfer a cell). This probability is very small and, furthermore our test has shown that the switch is capable of dealing with this situation at a load close to 100% without cell loss if CBR traffic policing is used. Details on the setup and the results can be found in [13]. This result, based on the worst possible case, proves that the rate division traffic shaping method is sufficient to regulate the event building traffic on an 8 port switch.

5. Larger Switches

In order to evaluate larger switches we have emulated a 64 X 64 switching fabric composed of 16 switches, 8 X 8 each, connected in 2 stages. The traffic that would be seen by one of the output stage switches (implemented with the Alcatel switch) can be emulated by a single 8 X 8 switch (the Bell Labs switch), as shown in Figure 7a). The broadband tester injects probe cells to measure the cell delay through the system and to detect any cell loss.

The measured statistical distributions of delay experienced by a cell when traversing both switches is shown in Figure 7b for various values of the aggregate load. These distributions show an increasing dispersion when the load increases, indicating queuing in the switch. Nevertheless, no data loss is observed at loads as high as 93%.

Using software models of the various types of switches that we could validate against measurements made on the small event builder demonstrators, we have studied the behaviour of large switches (256 X 256, 155 Mbit/s per port) with event building traffic in the



a) Test setup

b) Cell transfer delay for various loads

Figure 7: Emulation of a 64 X 64 event builder based on 16 switches,

push architecture. The traffic was composed of variable size event fragments with Erlang distribution, an average size of 4 KByte and a maximum of 10 KByte. Three rates were simulated: 1 KHz (23% load), 2 KHz (46% load) and 3 KHz (70% load). The input ports implemented the rate division (CBR) traffic shaping. They were not synchronized, each one moving in time with a random jitter relatively to the other sources.

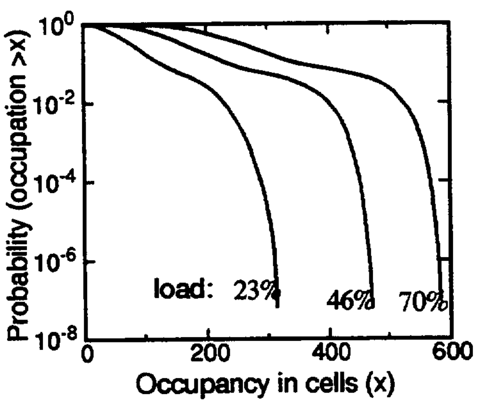


Figure 8: Probability of buffer occupancy in a switching element of a 256 X 256 switching fabric.

Figure 8 shows the buffer occupancy tail distributions for a switching element in a 2 stage regular 256 X 256 switching fabric built from 16 X 16 switching elements. The figure shows tail distributions that, for a given value of buffer occupancy (x-axis), give the probability (y-axis) that this buffer occupancy is exceeded. Provided that a switching element buffer can store at least 600 cells, the probability to lose cells is very low for loads less than 70%. Current LAN switches already provide larger values of buffer.

6. Simultaneous Traffic in Sources and Destinations

So far we have discussed the performance of the event builder assuming that the event fragments were already available in the source memory and discarding events in the destination as soon as they were assembled. We next consider solutions for the complete data transfer from the read-out buffers to the source network link on one hand and from destination link to the analysis processors on the other hand.

6.1 *Source Modules*

Event fragments have to be built by accretion of smaller blocks of data located in several read-out buffers connected to a source module by a local bus. In order to achieve the best performance, the bandwidth of this bus has to be at least of the same order as the network link bandwidth, and must provide good performance for data blocks of the size delivered by the ROBs. At present we can envisage VME and PCI as standard local buses. PCI seems to offer the best performance characteristics, however its limited physical range restricts its use to a small number of ROBs (typically up to 4). VME offers a bandwidth large enough to match with ATM STM-1 links. Its limitation is due to large overheads for the initialisation of block transfers.

We have measured the performance of a source module using VME to link the ROBs. As source controller we have used a CES RTPC board [14] with the ATM 8468 interface from CES [5]. The ROBs were emulated by a single slave board (in our case a FIC 8234 from CES) from which a variable number of data blocks was transferred into the source controller memory. The RTPC uses a 100 MHz PowerPC 604 with a second level cache of 512 KByte. The VME interface is connected to the PCI bus of the RTPC. Block transfer between the slave memory and the source controller is performed by means of the Block Mover Accelerator (BMA) hardware controller which also provides for chained block transfer driven by a list of descriptors stored in the RTPC memory.

We used the VME and ATM drivers provided by CES under LynxOS. They offer asynchronous access to VME and ATM (at the AAL5 level) thus allowing concurrent transfers on both. The test program consisted of 2 threads, one for VME read-out and one for the transfer of event fragments on the ATM link, each one passing control to the other once it had initiated a transfer.

The results are shown in Figure 9. The chained block transfer in VME is very efficient and no significant throughput variation is observed if the event fragment is composed of one or several blocks (up to 8). The measured overhead when starting a chained block transfer is about 120 µsec plus 5 µsec for each subsequent block. It is possible to implement a simpler version of the VME driver if higher performance for small fragments if required.

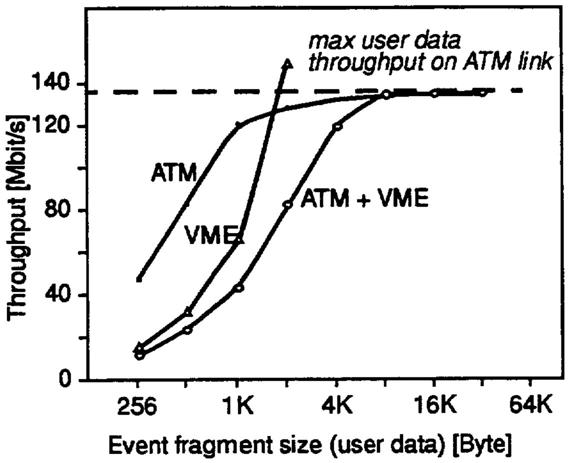


Figure 9: Throughputs for ATM, VME and simultaneous transfer in the CES RTPC board.

The PCI option is being evaluated in a separate project that implements and tests a PMC-based solution for the ATLAS Read-out Buffers [15].

6.2 Destination Modules

The problem of data transfer to the analysis processors is in principle not difficult: it is relatively easy to achieve high performance for the large blocks formed by complete events. However, an additional overhead is imposed by the fact that the event fragments, of variable length, arrive in unpredictable order and a copy operation may be necessary in order to store the event in contiguous virtual memory space.

One or two Fast Ethernet ports could be sufficient to carry the traffic of a destination to the processors. In the example of a 1 MByte event, the transmission time, is of the order of 0.1 sec. This is compatible with the use of TCP/IP which is difficult to avoid for Ethernet and commercial UNIX workstations or PCs. We have measured a bandwidth occupation as high as 80% on a 10-Base isolated Ethernet link with TCP/IP packets of 4 KBytes, under Lynx-OS.

7. Conclusions

The 155 Mbit/s ATM technology, as implemented in today's commercial products, can deliver high performance: the switches support high aggregate throughput; the adaptors can operate up to the full bandwidth, the "saturation point" depending on the overheads that can be minimized if optimal choices of software are made and if powerful CPU's are selected. The ATM technology is versatile and can support both data and control traffic.

We have shown on 8 X 8 event builder demonstrators that high throughput can be achieved (~120 MByte/s), with no data loss, by using a simple traffic shaping technique (rate division) that is provided as a standard in network adapters. Software models of larger switching fabrics (up to 256 X 256 at 155 Mbit/s/port) predict that the event building traffic can easily be supported by current technology for aggregate loads at least up to 70%. The switches that will soon be available from industry will probably provide effective congestion avoidance techniques.

The requirement for source and destination modules to implement simultaneous traffics in a very efficient way, is one of the most challenging tasks in the design of event builders. The VME based computer boards available now can sustain simultaneous VME and ATM traffic with a performance that is suitable for the present event builder requirements.

8. References

[1] D. Calvet et al., "A Study of Performance Issues of the ATLAS Event Selection System based on an ATM Switching Network", in *IEEE Transactions on Nuclear Science*, vol. 43, No 1, pp. 90–98, February 1996.

[2] M. Costa et al., "Results from an ATM-based Event Builder Demonstrator", *IEEE Trans. on Nucl. Sciences,* Vol. 43, No. 3, pp 1814 -1820, June 1996.

[3] The ATM Forum, "ATM User Network Interface Specification", Version 3.0.

[4] Transwitch Corp., Shelton, Connecticut, USA, SARA chip set, Technical Manual, version 2.0, Oct. 1992.

[5] IDT Inc., Santa Clara, CA, USA, IDT77201 NICStAR chip, User Manual Vers. 2.0, November 30, 1995.

[6] Creative Electronic Systems SA Geneva, ATM 8468, PCI-ATM Mezzanine Card, DOC 8468/PG, Version 1.0, May 1966.

[7] Creative Electronic Systems SA, Geneva, RIO2 8060, PowerPC based RISC I/O Board, Technical Manual vers. 1.0, DOC 8060/UM, October 1995.

[8] D. Calvet, F. Servaz., "Design of a PMC/ATM Interface", *Proc. of the 2nd Int. Data Acquisition Workshop (DAQ96),* RCNP, Osaka, 13–15 Nov. 1996.

[9] D. Calvet et al., "Performance Analysis of ATM Network Interfaces for Data Acquisition Applications", *Proc. of the 2nd Int. Data Acquisition Workshop (DAQ96),* RCNP, Osaka, 13–15 Nov. 1996.

[10] M. Henrion et al., "Technology, Distributed Control and Performance of a Multipath Self-Routing Switch", in *Proceedings of the XIV International Switching Symposium,* vol. 2, pp. 2–6, Yokohama, Japan, October 1992.

[11] V.P. Kumar et al., "Phoenix: A building block for fault tolerant broadband packet switches", *Proc. of the IEEE Global Telecommunication Conference,* December 1991, Phoenix, USA.

[12] C. Paillard, "An STS-OC3 SONET/STM-1 SDH ATM Physical layer implementation and Application to an ATM Data Generator", RD-31 note 95–04, February 1995.

[13] M. Costa et al., "Lessons from ATM-based Event Builder Demonstrators and Challenges for LHC-scale Systems", in *Proceedings of the 2nd Workshop on Electronics for LHC Experiments,* pp 208–214, Balatonfüred, Hungary, September 23–27, 1996.

[14] Creative Electronic Systems SA, Geneva, RTPC 8067LK, PowerPC Single Board Computer, Technical Manual vers. 2.0, DOC 8067LK/UM, May 1996.

[15] O. Gachelin et al., "ROBIN: A Functional Demonstrator of the ATLAS Trigger/DAQ Read-Out Buffer", *Proceedings of the 2nd Workshop on Electronics for LHC Experiments,* pp 204–207, Balatonfüred, Hungary, 23–27 September, 1996.

IEEE Transactions on Nuclear Science, Vol 43, No.3, June 1996

APPENDIX 2

Results from an ATM-based Event Builder Demonstrator

M. Costa, J.-P. Dufey, M. Letheren, A. Marchioro, R. McLaren, C. Paillard
CERN, 1211 Geneva 23, Switzerland

L. Gustafsson
Uppsala University, ISV, Uppsala, Sweden

A. Manabe, M. Nomachi
National Laboratory for High Energy Physics, Oho 1-1, Tsukuba 305, Japan

D. Calvet, K. Djidi, P. Le Dû, I. Mandjavidze
CEA Saclay, 91191 Gif-sur-Yvette CEDEX, France

T. Lazrak, Th. Lindblad, H. Tenunen
The Royal Institute of Technology (KTH), Stockholm, Sweden

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

M. Benard
Hewlett Packard, Geneva, Switzerland.

Abstract

ATM switching fabrics are good candidates to implement high performance parallel event builders for the future data acquisition systems in particle physics experiments. We are studying their feasibility through simulations and implementation of event builder demonstrators. We present results from performance measurements made with a demonstrator based on a commercial ATM switch and on network interfaces that we have developed. The measurements are compared with the simulation studies and their scalability is discussed.

I. INTRODUCTION

The RD31 project is evaluating ATM (Asynchronous Transfer Mode) (see for instance [1]) as a possible technology for implementing high rate and high data throughput event builders [2].

We have developed detailed computer simulation models of commercial or generic switches and various software tools that are used to implement models of most types of event builder architecture and data flow that are foreseen for the future experiments. The simulations carried out so far have shown that the ATM technology is a good candidate for several types of event building applications. Results can be found in [2] and [3].

In parallel with simulation studies, we develop small demonstrators in order to validate our understanding of the standards, to measure the performance of actual implementations and to evaluate various traffic shaping schemes.

The aim of this contribution is to present performance measurements made with a 4 X 4 event builder demonstrator based on a commercial ATM switch and on ATM interfaces developed by RD31. We shall first summarize the main features of ATM. The characteristics of event building over an ATM switching network and the specific problems caused by this type of traffic will be outlined. We will then present the measured switch performance under various traffic conditions, using different traffic shaping schemes. A comparison of the results with the simulation studies will be made in order to evaluate the scalability of the event builder. The performance of two event building algorithms will be compared in terms of their impact on global latency and event building frequency.

II. ATM TECHNOLOGY AND EVENT BUILDING WITH AN ATM SWITCH

A. *ATM Technology*

ATM is a connection oriented packet switching technology based on fixed length packets, called *cells*, of 53 bytes (5 bytes of header and 48 bytes of payload). Cells are routed through the network via virtual connections (VC) which define the characteristics of point to point links. The standard requires the sequence of cells to be preserved on a VC. Multiple VC's can be opened simultaneously on a physical link. There is no connection overhead when a source switches from a VC to another.

Above this "ATM layer", an *adaptation layer* is defined which offers a choice of standard adaptations to various types of services. For data transmission in event building we have selected the AAL5 (ATM Adaptation Layer 5) protocol which specifies the transmission of data packets with variable length up to 64 Kbyte. In a source, an AAL5 packet is complemented with an 8 byte trailer that is terminated with a CRC (Cyclic Redundancy Check) and it is segmented into cells. Reassembly and CRC check occur at the destination.

AAL5 does not include data retransmission in case of error (e.g. cell loss). It is the responsibility of the higher level layers to implement it if needed. Presently, no flow control is provided by the standard, but the ATM Forum is in the process of standardizing a flow control mechanism for variable bit rate services [4].

The physical layer can be implemented with various technologies that have been standardized by ITU or the ATM Forum. We have selected the ITU SDH standard [5] which offers bit rates of (roughly) 155, 622 and 2488 Mb/s. ATM cells are placed asynchronously within frames that are emitted every 125 µsec. A small overhead is due to the SDH control bytes (1/27 at 155 Mb/s).

Various commercial chip sets implement the segmentation and reassembly (SAR) of AAL5 packets, the management of the VC's and the physical layer. For switches, the standard defines the services that have to be provided, but the implementation is not specified and many choices are possible, depending on the application domains. For telecommunications applications, where low latency real time services are supported, cells may be discarded in case of congestion and buffering at every node of the switching network must be sufficient to ensure a low cell loss probability under random traffic conditions. For LAN applications, backpressure or credit based flow control mechanisms may be provided in order to avoid cell loss.

B. *Event Building with ATM*

An event builder is made up of multiple source and destination modules interconnected via a switching network. For every event the source modules collect the data from a subdetector to form an event fragment. Fragments are sent to a destination across the switching network. A controller assigns a destination for every event. Typically, for the Level 2 trigger, a sub-set of the sources is requested to send the whole, or a part, of the event fragment to a destination. For the Level 3 trigger, the complete event is assembled in the destination. The switching network supports simultaneous parallel data streams and multiple events are built concurrently.

A few parameters are of interest to characterize an event builder. The *event building latency* is the time between the decision to submit the event fragments to the event builder and the recognition, by the destination, that the event building is completed. The *load factor* is the maximum fraction of the available aggregate bandwidth of the switch that can be used for event building. The *event building frequency* is the rate at which event building can be accomplished. Source and destination *queue occupancies* are also interesting in order to determine the required amount of storage.

In an ATM event builder, VC's link each source to all destinations. These connections are virtual and are kept open permanently so that no connection set up time overhead degrades the data transfer throughput to a new destination. In the simplest case of a "push" architecture, the sources "push" their event fragments, as AAL5 packets, towards the destination as soon as they are ready. Other schemes are possible such as "pull" architectures, where the destination processors request the data from the sources.

The traffic pattern creates concentration in the switch and can lead to congestion and loss of data if no flow or traffic control mechanism is provided. Our simulations show that congestion can be avoided by using a switch with internal flow control, by applying traffic shaping or by combining both.

III. An Event Builder Demonstrator

We are investigating a push architecture on a square event builder (N X N). Every source has 1 VC with each of the N destinations. It should be noted that, as no information flows backwards from the destinations to the sources, it is not required to have a module connected on all the active output ports of the switch. The destination assignment (0..N-1) is implicit, the event k being sent to destination k modulo N.

The event builder demonstrator used for the performance measurements presented here is based on an 8 X 8 telecom switch prototype from Alcatel Bell (Belgium) [6] and on 2 types of ATM adaptors that we have developed: an ATM-VME interface and a simplified ATM traffic generator/capture module. They can be combined in various ways to test event building architectures on a small scale (Fig. 1).

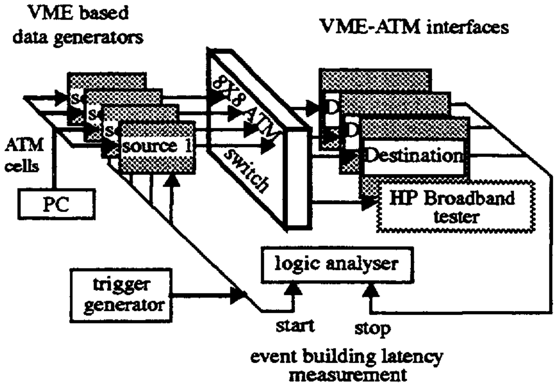


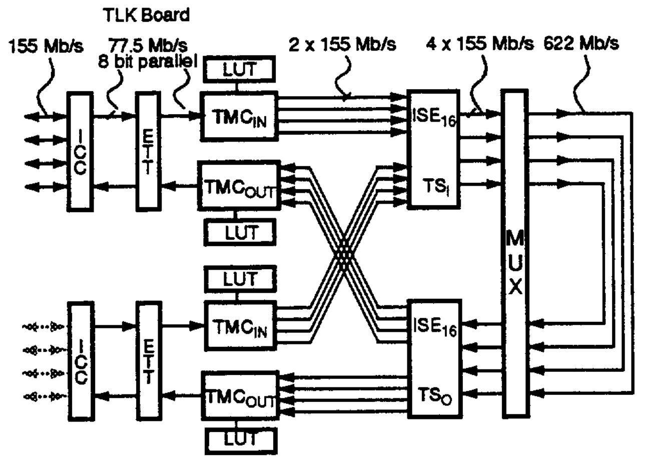
Fig. 1: Example of configuration of the event builder demonstrator.

A Hewlett Packard broadband tester [7] is connected to the switch and can act as a traffic generator or as a traffic analyser, measuring throughput and latencies and signalling the errors.

This section describes the hardware and software components of the demonstrator. Performance measurements will be presented in the next section.

A. *The Switch*

A detailed description of the switch architecture can be found in [6] and a summary of the main features in [8]. It is a Multi-Path Self-Routing (MPSR) broadband switching fabric developed by Alcatel for public network applications. The architecture allows expansion of the switch to very large aggregate bandwidths in incremental steps. Fig. 2 shows the layout



ICC - Interface Conversion Chip (Serial to parallel)

ETT - External Transmission Termination (SONET/SDH to ATM)

TMC - Transmission Mode Conversion (ATM to Multi Slot Cells)

ISE - Integrated Switching Element (Switching of MSC-s)

LUT - Look Up Table

Fig. 2: Switch lay-out.

of the 8 port version that we are using. Our present set up is limited to the use of 4 ports only.

B. The ATM Adaptor

The VME-ATM adaptor developed by RD31 is a full duplex interface that implements the AAL5 segmentation and reassembly and the ATM layer (including the management of VC's) using the SARA-S and SARA-R chips from Transwitch [9]. The SONET/SDH physical layer, at 155 Mb/s, is based on the SUNI chip from PMC Sierra [10]. The SARA chipset implements the rate-division traffic shaping discussed later, and provides error checking of the data received. The ATM network adaptor is plugged into a VME RIO board from CES [11] which includes, as host CPU, a 25 Mhz MIPS R3000.

C. The ATM Traffic Module

A simple VME module, that can be a generator or receiver of ATM traffic, has been developed with the aim of providing a low cost source and destination module. It uses the same physical layer implementation as the ATM adaptor. The ATM/AAL5 layer is replaced by a memory interconnected to the physical layer and to VME.

Presently the traffic module is used mainly as event builder source and allows to simulate easily several types of traffic shaping. The ATM cells, with the AAL5 structure, are created by a general purpose program on the basis of specifications of the required global traffic pattern and then downloaded into the traffic module memory. An external trigger, sent simultaneously to all sources, initiates the delivery of ATM cells into the network. A VME based PC can control several of these modules to download, analyse, define and edit ATM cells.

D. The Protocol Stack and the Event Building Software

The software that drives the ATM interface is structured in layers. Fig. 3 shows a simplified representation of the functional structure.

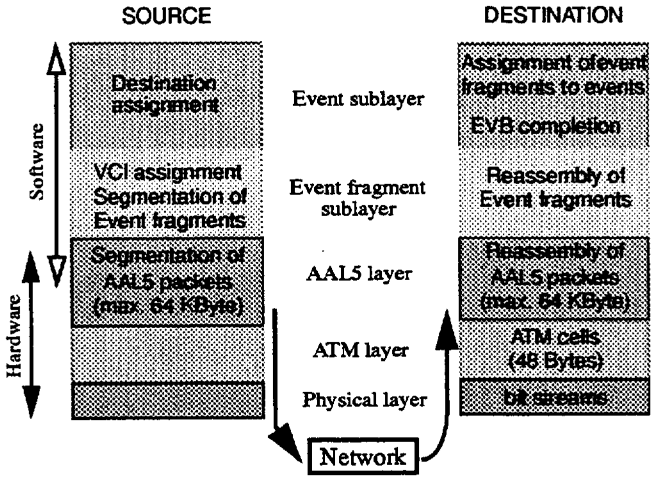


Fig. 3: Software layer structure.

The *event fragment sublayer* ensures independence from the packet size defined by AAL5 and more generally from the network technology. In particular packets can be of any length. The *protocol data unit* (PDU) in which an event fragment is encapsulated is illustrated in Fig. 5. An *event fragment PDU* is formed by complementing the payload with the event number and the destination identifier. It is then segmented in one or more *event fragment SAR* (segmentation and reassembly) *PDUs*, which must include a source identifier and a fragment sequence number. This structure is then encapsulated in an AAL5 packet.

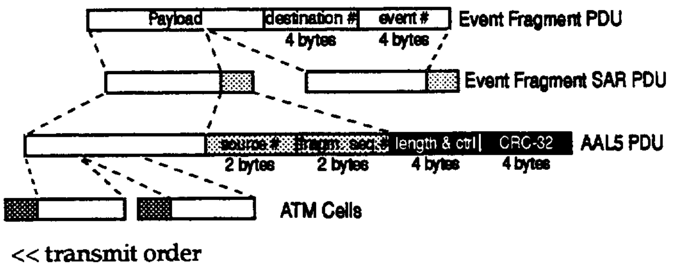


Fig. 4: ATM based event building protocol data units (PDU)

In the *event sublayer* of the destination, two algorithms have been implemented to determine when the building of an event is completed, namely the *time out* and the *notification* algorithms [12]. They will be briefly described in the next section.

Currently, no transport layer has been implemented to handle retransmission in case of errors. Errors are detected by checking the AAL5 CRC. Events with one or more errored packets are simply discarded. In the demonstrator, which contains many prototype boards, the probability of receiving an errored packet has been measured and is around 10^{-6} for packets of 1KB. If a transport layer were considered as a necessity, it could be provided either by a standard network protocol like TCP/IP or, in a lighter way, by a dedicated function in the event fragment sublayer.

As the destination will receive several event fragments, it is important for the scaling of the system that the performance is a linear function of the number (N) of fragments. This requirement has implications on the choice of data structures and algorithms, in particular on the algorithms used to search for fragments and to recognize the end of event building. In our demonstrator the time spent on one event is of the form: $t_0 + N \cdot t_f$, where t_0 is a fixed overhead for the global event building operation and t_f is the overhead for 1 fragment, including the data transfer.

In order to keep the overheads as low as possible, the event building software is designed to rely only on semaphore polling while interrupts are used only when errors occur.

IV. Performance Measurements

A. Performance of the Adaptor

The performance of the VME-ATM adaptor has been measured in transmission (source) and reception (destination). For the measurements there was no transfer of data between VME and the packet memory (it is assumed that suitable dual ported memories and DMA block transfers will be available to feed the packet memory without additional overhead).

Fig. 5 shows the measured performance for the ATM throughput and the fragment rate at emission and reception. The ATM throughput measures the amount of transmitted cell data, including the cell headers. After subtraction of the SONET/SDH overhead, the maximum theoretical limit is 149.76 Mb/s. The fragment rate measures the maximum frequency at which the interface can send, or receive event fragments of a given size.

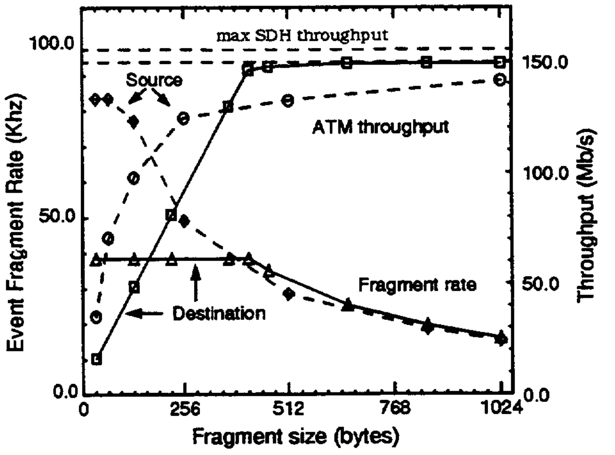


Fig. 5: Event builder source and destination performance.

In the source, for small packet sizes (up to 2 ATM cells, or 88 bytes of user data), the throughput is limited by software and hardware overheads (approximately 12 µsec). Although the bandwidth utilization is rather poor (approx. 25%), the frequency is high and indicates that event building of small event fragments can proceed at high trigger rates. The lower performance for the receiver side, and consequently the lower frequency (38 kHz), is due to a higher software overhead at reception, where the event building protocol has more tasks to carry out.

For larger packets the software overheads play no role any more as they proceed in parallel with the data transfer. The hardware link performance is determinant in this case. When the interface is used as a source, it can transmit fragments of 1KB at 95% of the link bandwidth. When it is used as a destination, the interface hardware can receive at the maximum ATM speed. This guarantees that the interface can absorb bursty traffic without losing cells.

For the larger packet sizes, the lower throughput of the source, compared with the destination, is due to a small link inefficiency between the segmentation chip and the physical layer (a 16 bit link is required instead of 8 bits as implemented). This has been corrected on the destination side, where it is crucial to reach full bandwidth to avoid cell losses. In this case, Fig. 5 shows that the throughput curve has a perfect shape, increasing linearly in the interval dominated by the software overheads (approximately 25 µsec) and reaching full link capacity above.

B. Performance of a 4 x 4 Event Builder

In the present status of our development, the event builder test system uses 4 ATM traffic modules as sources, 1 ATM-VME interface and the HP broadband tester as destinations, while the other 2 destinations receive event fragments but are left open, which, as already pointed out, has no effect on the performance measurement. The event building latency is determined with a logic analyser that measures the time elapsed between the first cell of an event fragment submitted to the physical layer in the source module and a signal generated by the software in a destination, when the event has been completed. The broadband tester checks that no cell loss occurs, measures the fragment rate and throughput as well as the latency of the switch.

i) Congestion Avoidance using Traffic Shaping

The 4 sources send event fragments of equal size to 4 destinations. We compared the behaviour of the switch for different traffic patterns by measuring the cell latencies through the switch and checking for cell losses. The traffic patterns implemented are: a) ***no traffic shaping:*** each source sends the event fragments one after the other to the destination (FIFO of event fragments); the traffic has been tested at 2 different physical link loads (50% and 82%). b) ***rate division:*** this is the traffic control provided by default by the SAR chip. The source maintains one queue of event fragments for each VC and 1 cell is extracted from each queue in round robin manner at a defined rate. c) ***barrel shifter:*** the sources are synchronized by an external signal. A logical FIFO queue is maintained for every destination. A source extracts cells from the same queue during the time interval T_c between 2 signals and changes queue, at a new signal, in a well defined way such that no two sources can send to the same destination simultaneously.

Fig. 6 shows the results for all these cases and a comparison

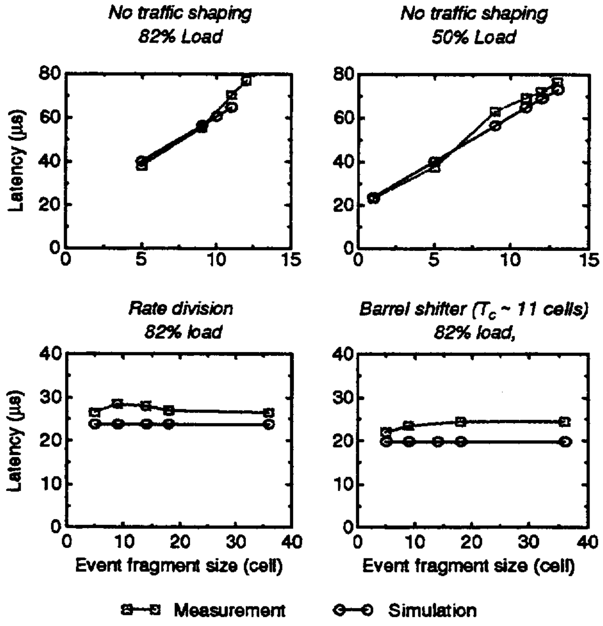


Fig. 6: Cell latency for different traffic shaping schemes.

with a simulation model of the switch If no traffic shaping is applied, the latency grows linearly with the size of the event fragment. This is an indication that the buffers in the switch are accumulating cells as a consequence of the concentration of data. In fact the last point of measurement in the top graphs is the limit beyond which cell loss occurs. Reducing the mean load per link from 82% to 50% has practically no effect in avoiding congestion.

The rate division and the barrel shifter schemes provide a good distribution of the traffic and do not result in accumulation of cells in the switch. For our experimental conditions, these two traffic shaping methods give the same results. However one can expect a difference between the 2 schemes for larger switches because the rate division alone does not break the correlation between the sources and many of them can, at the same time, send a cell to the same destination. In fact the simulation shows that already for a 16 X 16 event builder the rate division is not sufficient to avoid cell losses.

ii) Full Event Building

In a full event builder, for each event, all sources send a fragment. Fig. 5 shows measurements of the maximum event building rate and of the aggregate user throughput as a function of the fragment size. Rate division traffic shaping is applied for these measurements.

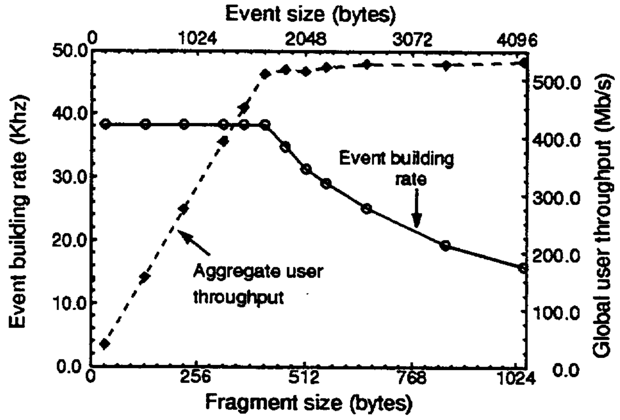


Fig. 7: 4x4 Event builder rate and throughput.

The event building rate remains constant (38Khz) for fragments smaller than 412 bytes because the software overhead in the destinations is the limiting factor. For bigger fragments the link throughput determines the maximum rate, as already discussed. The maximum aggregate throughput of user data is 527Mb/s (66MB/s). We should observe that, although the links are loaded at 99%, the switch does not loose cells. This is because only half of the inputs are in operation, resulting in a 50% total load factor of the switch. Using all the ports of the switch will lead to lower performance values because it will not be possible to use 100% of all links simultaneously. The maximum load factor will depend on the switch performance and on the efficiency of the traffic shaping.

In first approximation, for a "square" N X N event builder, the rate performance does not depend on N: the rate that a destination can sustain varies like 1/N and the number of destinations is N. However the scalability may depend on the acceptable load factor of the switch as a function of N.

We have also measured the event building latency for 2 traffic shaping methods: the rate division and the barrel shifter. Fig. 5 shows the results as a function of the event fragment size.

When rate division is applied, the event building latency increases roughly linearly with the event fragment size. The destination receives one cell from each source in round robin and the event building time is proportional to the event fragment size (or to the size of the largest fragment in the case of fragments of variable size). If N event fragments of equal size are expected, none of them is completed before the N-th round robin. Consequently, the operations on the event fragments cannot start before this delay.

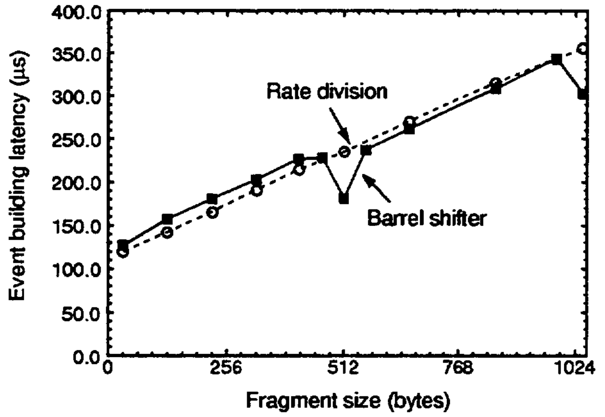


Fig. 8: 4x4 Event builder latency using the rate division and barrel shifter traffic shaping schemes.

In the barrel shifter case, we have chosen a constant time period (T_c) for all packet sizes, namely the time needed to transmit 11 cells. The particular case when event fragments have a size of exactly 11 cells is easy to understand: the event fragments arrive exactly one after the other and the operation for each one can proceed immediately and in parallel with the arrival of the next one. An additional overhead is needed to complete the event building. This efficient operation mode explains the dip observed for a size of 512 bytes (a similar effect of synchronisation can be anticipated for sizes equal to a multiple of 11 cells and is in fact visible at 1024 bytes). For event fragments with size different from a multiple of 11 cells, the latency varies periodically as function of the event number (see Fig. 5).

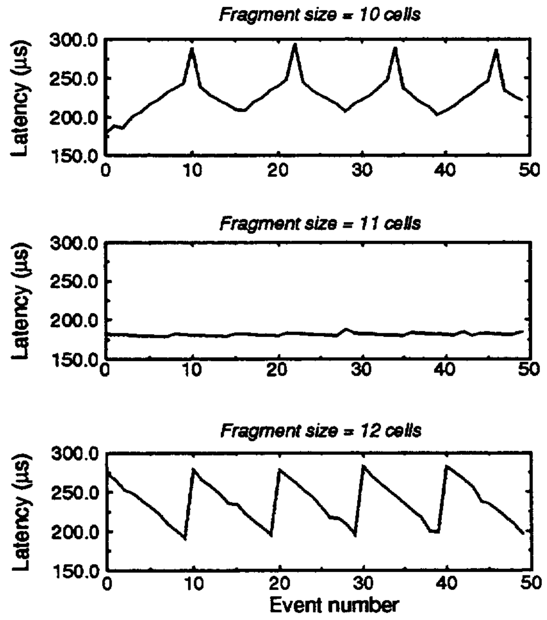


Fig. 9: 4x4 Event builder latency profile using the barrel shifter traffic shaping schemes for different event fragment sizes

As the event builder has a pure push architecture the throughput of the system does not depend on the latency and is the same using both traffic shaping schemes.

iii)Event Building Algorithms

In many applications of event building, only a subset of the sources have data to send for a given event and some mechanism must be provided in order to determine when all non empty fragments have been received in the destination.

We have implemented 2 of the proposed algorithms to determine the completion of the event building: *on time-out* and *by notification* [12]. In the time out case, after reception of the first fragment of a new event, one waits a time sufficiently long to have a low probability of missing data. In this implementation we use, as approximation of time, the arrival of a certain number of event fragments (4 in our test). In the notification case, when a source has no fragment for a given event, it sends instead a notification cell. In order to minimize software overheads, the notification is encapsulated in an OAM (Operation And Maintenance) F5 cell ([5]) instead of an AAL5 packet.

In both cases the rate division traffic shaping has been used. The event builder runs in sparse mode, corresponding to the Level 2 conditions: in our test, 2 sources (chosen randomly for each event) send event fragments of a fixed size while the other 2 either do not send any data (time-out algorithm) or send a notification cell (notification algorithm).

The event building latency and the maximum event building rate have been measured, for both cases, as a function of the event fragment size. The results are given in Fig. 10.

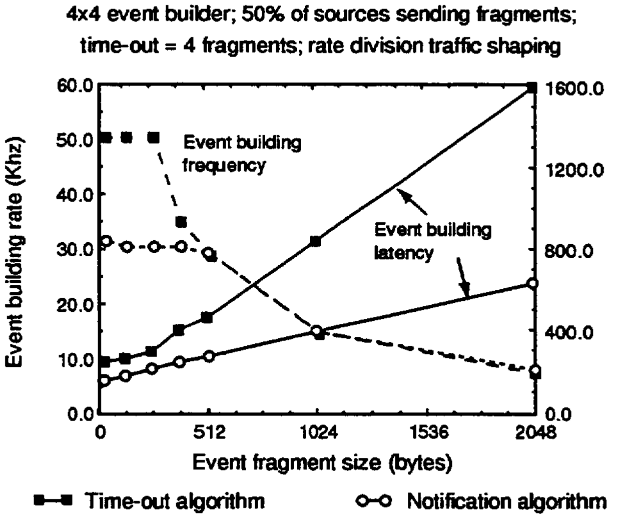


Fig. 10:Performance comparison of 2 event building algorithms.

The event building latency is mainly due to the data transfer time (which increases linearly with the event fragment size). The switch contributes for a constant amount in the case of rate division traffic shaping, as seen before. The time-out algorithm, in its present implementation, adds a latency proportional to the event fragment size. The event building latency is expected to grow with the size of the event builder.

The event building frequency has the characteristic shape seen before. In the case of a sparse event builder, the time-out algorithm does not process information from each source and the software overhead is smaller than for the notification algorithm, allowing higher event building frequency. The curves of Fig. 10 show that, with the present software overheads, the cut-off occurs at ~ 256 bytes for the time-out algorithm and at ~ 512 bytes for the notification algorithm.

The more complicated software explains the lower performance of the notification algorithm compared to the full event building.

V. CONCLUSION

A first small event builder using ATM has been set up and its performance measured. A complete event building software has been implemented in the source and destination modules. The system is completely decentralized in order to be scalable to very large event builder systems.

In the current implementation, using 25 MHz RISC processors, sources can send small event fragments at a frequency up to 80 KHz and destinations can receive them at 38 KHz (or higher if the number of destinations is superior to the number of sources). For small fragments the rate is limited by the software overheads. For larger fragment sizes the link throughput is the limiting factor.

Measurements of performance under various traffic patterns confirm the need for traffic shaping, as was previously shown through simulation. The implementation of different event building algorithms has allowed us to verify advantages and disadvantages of each of them in terms of global throughput and latency.

VI. REFERENCES

[1] M. de Prycker, *Asynchronous Transfer Mode*, 2nd ed., Ellis Horwood Series in Computers and their Applications, 1993.

[2] M. Costa et al., "NEBULAS - A High Performance Data-driven Event Building Architecture based on an Asynchronous Self-routing Packet-switching Network", *CERN/LHCC/95-14*, CERN, Geneva, March 1995.

[3] D. Calvet et al., "A Study of Performance Issues of the ATLAS Event Selection System based on an ATM Switching Network", in *Proceedings of the Ninth Conference on Real-Time Computer Applications in Nuclear, Particle and Plasma Physics (IEEE RT95)*, Michigan State University, East Lansing, MI, May 22-25, 1995. To be published in the *Conference issue of IEEE Transactions on Nuclear Science.*

[4] J.B. Lyles, "Definition of ABR Service Model", *ATM Forum document* 94-0709, 18 July 1994.

[5] The International Telecommunications Union (ITU); recommendations G.707, G.708, G.709.

[6] M. Henrion et al., "Technology, Distributed Control and Performance of a Multipath Self-Routing Switch", in *Proceedings of the XIVth International Switching Symposium*, Yokohama, Japan, October 1992, vol. 2, pp. 2-6.

Th.R. Banniza et al., "Design and Technology Aspects of VLSI's for ATM Switches", *IEEE Journal on Selected Areas in Communications*, Vol. 9, No. 8, Oct. 1991.

[7] Hewlett Packard, Broadband Series Test System, 1994.

[8] M. Letheren et al., "An Asynchronous Data-Driven Event-Building Scheme based on ATM Switching Fabrics", in *IEEE Transactions on Nuclear Science*, Vol. 41, No.1, Feb. 1994.

[9] Transwitch Corp., Shelton, Connecticut, USA, SARA chip set, Technical Manual, version 2.0, Oct. 1992.

[10] PMC-Sierra Inc., the PMC5345 Saturn User Network Interface (SUNI) manual, May 1993.

[11] Creative Electronic Systems SA Geneva, RIO 8260 and MIO 8261 RISC I/O processors. User's manual, version 1.1, March 1993.

[12] I. Mandjavidze, "Software Protocols for Event Builder Switching Networks", in *Proceedings of the International Data Acquisition Conference*, Fermilab, Batavia, Il, Oct. 26-28, 1994, p. 47.

APPENDIX 3

Lessons from ATM-based event builder demonstrators and challenges for LHC-scale systems

M. Costa, J.-P. Dufey, M. Letheren, C. Paillard
CERN, 1211 Geneva 23, Switzerland

D. Calvet, P. Le Dû, I. Mandjavidze
CEA Saclay, 91191 Gif-sur-Yvette CEDEX, France

Abstract

The RD31 project has implemented demonstrators of ATM based event builders of various sizes, up to 8 X 8. Congestion control techniques based on traffic shaping have been evaluated and compared with simulation results. The future LHC experiments will require larger systems, typically 256 X 256 or more. We shall first present the conclusions reached with small demonstrators that apply to larger systems and then discuss the use of computer models. The question of the availability of large commercial switches is raised, and some alternative approaches are suggested, such as networks of smaller switches.

I. Introduction

At the 1995 Workshop we presented the first results from a 2 X 2 event builder demonstrator based on the ATM (Asynchronous Transfer Mode) technology [1]. This work has been continued with the implementation of 4 X 4 [2] and 8 X 8 event builders. Results from performance measurements are presented here.

The demonstrator program was undertaken in order to clarify a number of questions, namely a) whether the hardware implementations of the ATM and adaptation layer (AALx) protocols would deliver the expected performance, b) whether ATM-based event building would require custom "traffic shaping" in order to avoid congestion and data loss in the switching network, c) the effect of software overheads in terms of performance and d) validation of our simulation models by comparing them to measurements. In addition commercial products were still emerging at that time and the performance of the first adapters was poor. However the future availability of large switches did not appear at that time as a severe problem: telecom switches were in test phase and were designed to scale to large sizes (thousands of ports).

We now have implemented demonstrators for push architectures (8 X 8) and made preliminary tests of pull architecture (6 sources and 2 destinations). The most important results are that ATM hardware can deliver full performance and that custom traffic shaping is not required for this size of event builder. We have implemented a scalable event building software and measured the overheads. The future availability of very large switches however is not clear and we are investigating the possibility to interconnect medium size switches. Our event builder simulation models have been tuned to give the best fit with the measured performance, for various traffic shaping methods. When applied to larger event builders (256 X 256), they show that standard traffic shaping techniques supported by commercial adaptors (rate division) is sufficient to avoid congestion at high loads.

The basics of ATM technology, the principles of ATM-based event building and details on the hardware and software used in the demonstrators can be found in [1]. We first present some updates on the hardware and software and then we discuss the results from performance measurements for various event builder tests and architectures. Comparisons with simulation results are shown.

In the second part of the paper we tackle the problem of large switches: trying to quantify the notion of "large" and to identify possible solutions. We give predictions from software models on buffer occupancy for these solutions.

II. Experimental set up

A detailed description of the experimental set up is given in [1]. We summarize here the main elements and their important characteristics and indicate the modifications that have been made since the last reports.

Hardware

The experimental set up is shown in Figure 1. Two switches have been used: one from Alcatel and one from Bell Labs. The 8 X 8 Alcatel switch [3] of "telecom" type has no internal flow control and it resolves contention by means of internal buffering, both in the switching elements and in the network interfaces, and through internal bandwidth expansion provided by multiple paths. The 8 X 8 switch from Bell Labs is a switching fabric based on the Phoenix 2 X 2 switching element which implements internal flow control by means of back pressure [4]. This mechanism holds up the traffic arriving into an internal switching element as soon as a certain level of buffer occupancy has been reached. Back pressure cannot propagate outside of the switch. As a consequence, the input network interface is the place where buffer overflow can occur, therefore it is equipped with a rather large memory (85 cells). In addition the internal bandwidth is expanded by a factor 2.

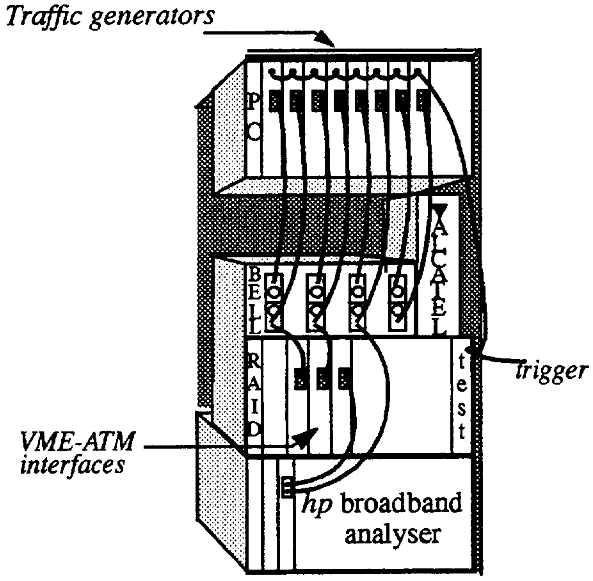


Fig. 1: Event builder demonstrator test bed

As source modules we use the traffic generators that we have developed [5]. Implemented as VME modules they include the SONET/SDH physical layer of the ATM standard protocol (at 155 Mbit/s) and a 1 Mbyte memory in which the traffic to be emitted is downloaded in the form of pre-formatted ATM cells. A general purpose program can generate off-line the ATM cells stream, with the AAL5 structure, on the basis of specifications of the traffic pattern required. Various traffic shaping schemes can be emulated. The generators can operate in a continuous mode, without software control, and can achieve full bandwidth. It is possible to vary the rate of emission by inserting dummy (or idle) ATM cells which occupy bandwidth on the link but are discarded by the switch's network interface. All generators can be started simultaneously by means of an external signal daisy-chained between them. A new mode of operation has been implemented for the pull architecture tests: a generator can be triggered by an incoming cell to send the next AAL5 packet in memory; then it remains idle until the next request is made.

As destination module we use an ATM adaptor that we have developed [1]. It is based on the Segmentation and Reassembly chip set SARA from Transwitch [7]. It is a mezzanine card that plugs into the CES RIO-1 VME board that contains a 25 MHz R3000 RISC processor [6].

Software

Figure 2 shows the layered structure of the protocols: layers 1 to 3 are the standard AAL5, ATM and physical (SONET/SDH) layers [11]. We use the AAL5 protocol to transfer the event fragments and, in the case of pull architecture, to carry the request messages as well. The AAL5 standard defines the transfer of data blocks of variable length, up to 64 KByte. The segmentation and reassembly (SAR) of these data blocks in ATM cells is performed by hardware in the SARA chip.

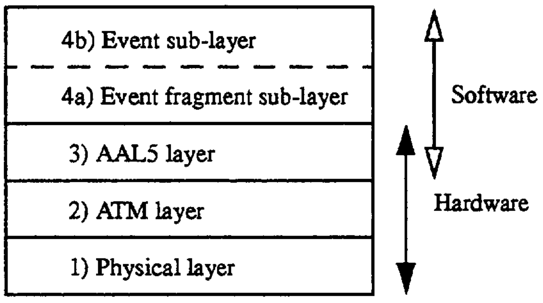


Fig. 2: Protocol Layer Structure of the event builder architecture

The event building layer is subdivided in 2 sub-layers (4a and 4b). Sub-layer 4a, the *Event Fragment Sub-layer,* ensures the independence of the sub-layer 4b from the specific network layers and allows to define a maximum packet size independently from the AAL5 standard.

The *Event Sub-layer* has the task of linking together the event fragments to form an event (they reach the destination in any order). The event sublayer is able to build several events concurrently. Its task is also to recognize when an event is completely assembled.

We do not use a *transport protocol* between layer 3 and layer 4, that normally takes care of retransmission of lost or corrupted packets. We have found that this does not happen (more precisely that the probability of data loss is very low) under normal operation of the event builder. Nevertheless a light transport protocol can be envisaged.

In order to minimize the overheads, the event building software runs in stand alone mode, with a minimum kernel and the destinations poll for the arrival of new packets rather than waiting for interrupts. Although we use data generators, the software that runs in a source has also been implemented and tested and the overhead has been measured.

III. Event builder performance

Performance of the ATM interface

Figure 3 shows performance measurements made on the 155 Mbit/s ATM interface based on the SARA chip set and with the R3000 at 25 MHz. The throughput curves are limited at the ATM layer to a maximum possible throughput of 149 Mbit/s, due to the physical layer framing overhead. Curve a) is the maximum throughput achieved when transferring AAL5 packets, in the absence of the event building layers of the protocol. Bandwidth saturation occurs for a packet size of 240 bytes which corresponds to an overhead of 16 µsec per AAL5 packet. Curve b) shows the throughput when the event building layer is active, measured on the

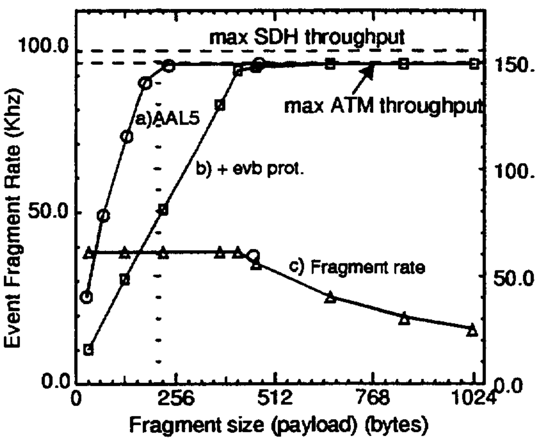


Fig. 3: Performance of the ATM interface (RD31)

receiving side where the overhead is greatest. The saturation occurs at a packet size of 400 bytes, corresponding to an overhead of 25 µsec. Curve c) shows the maximum frequency at which packets can be received. For a packet size less than the saturation value, the overhead dominates and the maximum frequency is a constant (38 KHz). For larger packets, the frequency is determined by the data transfer time.

Test of an 8 x 8 event builder with push architecture and traffic shaping

For the push architecture, in the absence of a transport protocol, no acknowledgement of event fragment reception is given. The maximum possible configuration, with the switches that we are using, is 8 sources and 8 destinations. An event builder test system has been set up using 8 ATM traffic generators as sources. Every source is linked to all destinations by virtual connections that are set up permanently. Although all destination ports receive data, only 1 needs to be equipped with an actual interface for measurement purposes.

Events are collected individually in a destination: no packing in "super-events" is performed. The destination is assigned in round robin fashion to successive events. The event fragments are either of fixed length, the same in all sources, or follow a normal distribution with a variance proportional to the mean value.

The simplest way to send the event fragments is to submit them, as soon as they arrive, via a single FIFO queue per source. This generates bursty traffic where all sources may send to the same destination for the duration of the event fragment transfer. For this scheme we have measured the individual cell transfer latency across the switch as a function of the event fragment size. Figure 4a shows results for the case of a 4 X 4 event builder based on the Alcatel switch.This latency grows linearly, suggesting that cells are queuing as a consequence of increasing buffer occupancy

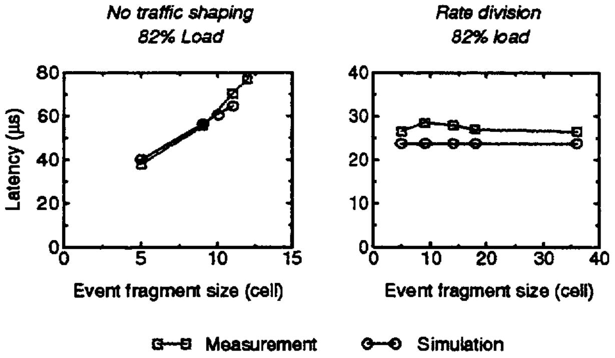


Fig. 4: Cell latency without and with traffic shaping (rate division).

inside of the switch. Eventually the latency becomes infinite if cells are discarded when a buffer overflows. This method is not appropriate for event building as it imposes a limit to the event fragment size.

The ATM standard defines various types of traffic that can be assigned to a virtual connection (VC). We have adopted CBR (Constant Bit Rate): in the case of a symmetric N X N event builder a constant rate equal to 1/N of the physical link rate is assigned to every VC. This ensures that the average aggregate bandwidth arriving at a destination does not exceed the nominal bandwidth available at the output port. The implementation of CBR is done in the SAR chip. The mechanism is illustrated in Figure 5: in the source adapter, a queue per destination (or per VC) is maintained. These queues are scanned in a round robin fashion, 1 cell being extracted from each queue each time. The sources are not correlated in time because each adapter has its own clock and this *rate division* scheme provides a kind of traffic shaping that smoothes the effect of the trigger. The hardware implementation of CBR in the SAR chip does not produce any additional overhead. Figure 4b) shows the cell latency as a function of the event fragment size in the case where CBR is applied. The latency remains constant, indicating that no queuing occurs inside of the

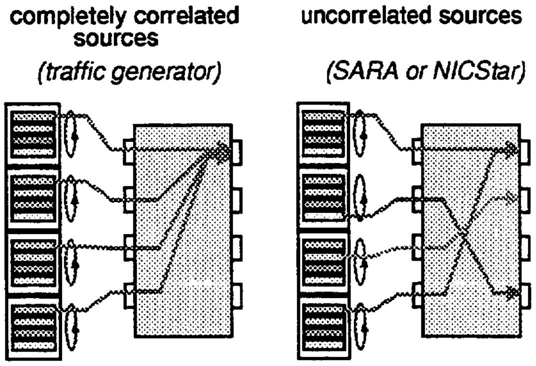


Fig. 5: Rate division traffic shaping

switch. Both graphs in Figure 4 also show the cell latency curves predicted by the software model of the switch in both cases. The agreement with the measured values is good and suggests that the model is reliable.

There is still the possibility that all sources are exactly synchronized and, consequently, that a wave front of 8 cells is sent to the same destination every 8 time slots (a time slot is the time to transfer a cell). However, the network interfaces having independant clocks, this probability is very small and, furthermore, the switch is capable of dealing with this situation, as was shown with the test set-up of Figure 6 designed to determine the performance of the Alcatel switch under maximum throughput with rate division and synchronized sources. The Bell Labs switch has been used to emulate 8 synchronized sources for the Alcatel switch: 1 input was fed from the HP tester. Every cell entering was broadcast to all 8 outputs. Consecutive cells were assigned to different VC's, one per output of the Alcatel switch. Despite of this bursty traffic a load close to 100% was achieved without cell loss. Figure 7 shows the distributions of the delay experienced by a cell when traversing both switches for various values of the aggregate load. These distributions show an increasing dispersion when the load increases, indicating queuing in the switch. Nevertheless, no data loss was observed at loads as high as 96%. This result, based on the worst possible case that can happen with rate division, proves that this traffic shaping method is sufficient to regulate the event building traffic on 8 port switches.

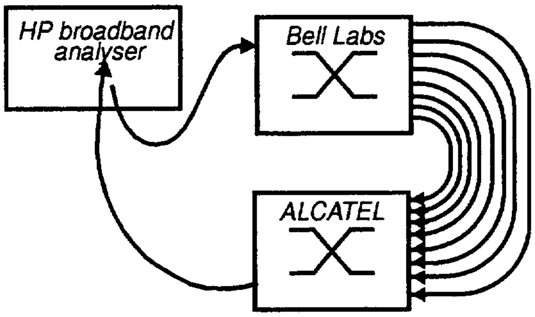


Fig. 6: Test set-up to emulate full event building

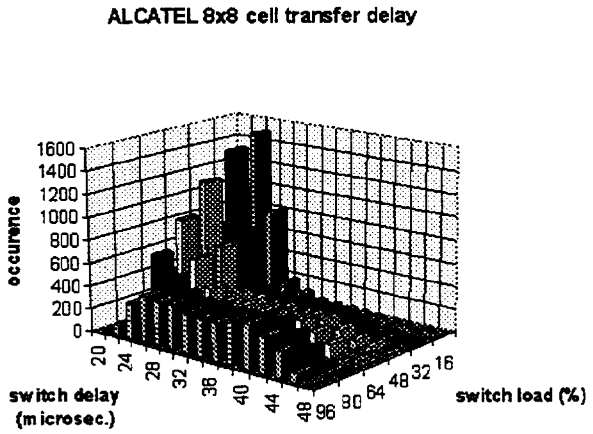


Fig. 7: Distribution of cell switch transfer delay for various loads (set-up of Figure 6)

Figure 8 shows the performance of the Bell Labs switch with 8 traffic generators as sources. The event fragments have variable sizes normally distributed with σ^2=30% of the mean value. The maximum reachable aggregate throughput (100% load) for user data is approximately 135 MByte/s (after removal of the overheads at the physical, ATM and AAL5 layers). We reach 120 MByte/s (i.e. 89% load) The data in the generators memory cannot be filled in an efficient way for variable length event fragments which accounts for this loss.

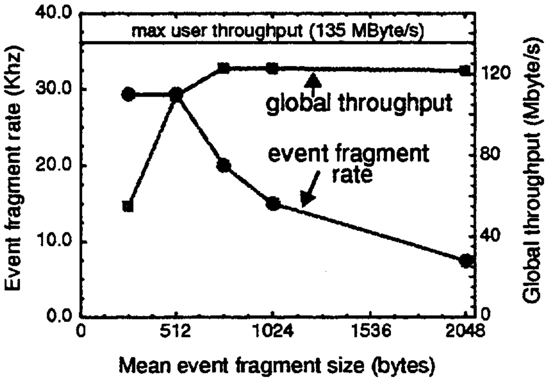


Fig. 8: 8x8 event building; push data flow; gaussian size distribution

IV. What we have learned so far

- The ATM technology, as implemented in commercial products, can deliver high performance: the switches support high aggregate throughput; the adaptors can operate up to the full bandwidth, the "saturation point" depending on the overheads that can be minimized if optimal choices of software are made and if powerful CPU's are selected.
- It is possible to implement very efficient event builders. We have been able to load 8 X 8 switches close to 100%, with the standard rate division traffic shaping method. On larger systems, the maximum load achievable will depend on the internal characteristics of the switches. This point will be developed in Section V.
- ATM technology is versatile and provides an easy way to implement a mixture of data and control traffic.
- Efficient implementation of an event builder requires from the designers a very good understanding and practice of the ATM technology.

Table 1: Requirements of LHC experiments

Experiment	Physics Throughput	Max. freq./port	Switch size (@~50% load)	Load	# duplex ports (155 Mbit/s)
ATLAS (Level 2 only)	12 Gbit/s	20 KHz			
ATLAS (Level 3 only)	10 Gbit/s	1.5 KHz	128 X 128	58%	256
ATLAS (Architecture C: Level 2 and 3)	10 Gbit/s	20 KHz	256 X 256	30%	512
CMS (Virtual Level 2)	250 Gbit/s	100 KHz	1000 X 1000 (@622 Mbit/s)	46%	8'000
ALICE (50 Hz, 40 MB/ev) or 1 KHz, 0.5 MB/ev)	16 Gbit/s	1 KHz	200 X 200	60%	400

"Buy, Plug'n Play" may work for multimedia applications running at human reaction time scale, it will give very poor performance for fast event builders.

V. Feasibility of large event builders

How large is "large"?

Table 1 shows the requirements of 3 LHC experiments, ATLAS, CMS and ALICE. The number of ports has been calculated assuming a load on the switch around 50% and full duplex connections even in the case of push architectures. In addition one has taken into account the effective user data throughput which is 135 Mbit/s on a 155 Mbit/s (OC-3c) connection. In the case of a 2 stage network of switches, the number of ports has to be doubled once more.

With the exception of CMS, the event builders require very similar aggregate physics data throughputs of 10 to 20 Gbit/s, corresponding to switches of 40 to 80 Gbit/s aggregate throughput or 256 to 512 ports at 155 Mbit/s.

The ATLAS architecture C is noticeable as it implements event building in pull mode for Level 2 and Level 3, combining the data and the control traffic on the same switch. It takes full advantage of phased event selection to reduce considerably the amount of data to be transferred.

Possible solutions for large switches

LAN switches are usually implemented on a backplane and, consequently, are not scalable to very high throughputs. They are non-blocking up to this maximum design throughput. Larger switches are now available which can be implemented as an assembly of several smaller backplane switches with central or distributed buffers. Some LAN switches implement flow control. Switches with an aggregate bandwidth of 10 Gbit/s can be found on the market now and 20 Gbit/s will probably be available soon. This is still too low for the class of 10-20 Gbit/s physics data event builders that require in fact switches with 40-80 Gbit/s aggregate throughput.

The *interconnection of switches* is an approach that can be envisaged. For the event builders requiring 128 ports, 8 interconnected switches of 64 OC-3 ports would be sufficient (Figure 9). In the worst case, if we assume that all sources are synchronized, an output port receives bunches of ~128 cells. The aggregate number of cells to be stored in order to absorb the burstiness of all output ports is of the order of 4000 cells per switch. Large switches currently available (64 ports) have by far enough buffering.

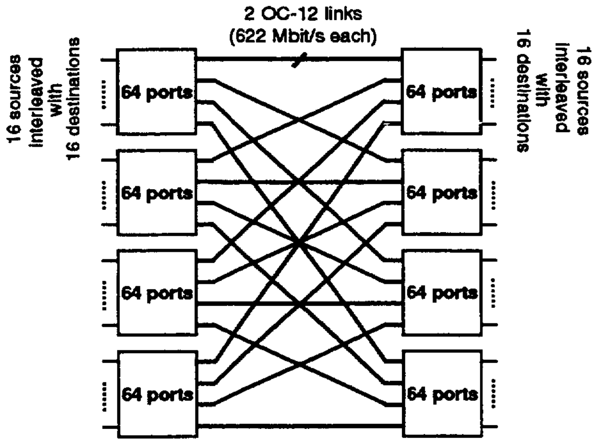


Fig. 9: A 128 X 128 switching fabric made of 8 switches, 64 ports (OC-3) each

Currently only *telecom switches*, based on fabrics of switching elements (8 X 8, 16 X 16,...) have an architecture capable of scaling to many thousands of ports. However very large configurations have not been deployed so far. Their ability to avoid congestion over common paths in the switch relies on bandwidth expansion, multiple paths and local buffering in the switching elements.

The ATM Forum has designed a standard flow control, ABR (Available Bit Rate) to feedback the information on possible congestion to the sources. No evaluation of the applicability to event building has been made so far in our project, mainly because the first commercial boards and switches implementing ABR are not available yet.

Evaluation of the feasibility of large event builders

Using software models of the various types of switches that we could validate on the small event builder demonstrators, we have studied the behaviour of large switches (256 X 256, 155 Mbit/s per port) with event building traffic in the push architecture. The traffic was composed of variable size event fragments with Erlang distribution, an average size of 4 KByte and a maximum of 10 KByte. Three rates were simulated: 1 KHz (23% load), 2 KHz (46% load) and 3 KHz (70% load). The input ports implemented the rate division traffic shaping. They were not synchronized, each one moving in time with a random jitter relatively to the other sources.

Three switch architectures have been evaluated: a) a 2 stage regular switching fabric with 16 X 16 switching elements without data flow control, b) an 8 stages Banyan switching fabric composed of 2 X 2 Phoenix switching elements implementing back pressure flow control, and c) an Alcatel switching fabric with 16 X 16 switching elements in Clos architecture with an internal bandwidth expansion (multi-path) of 1.56. The aim of the simulation was to determine the occupancies of queues and/or buffer and identify where buffer overflow is likely to occur. The figures that follow show probability curves (or tail distributions) that, for a given value of buffer occupancy (x-axis), give the probability (y-axis) that this buffer occupancy can be exceeded.

Figure 10 shows the aggregate buffer occupancy for a switching element in the switching fabric a). Provided that a real implementation has a buffer exceeding 600 cells in the switching element, one can see that the probability to lose cells is very low. Presently LAN switches exceed these values of buffering.

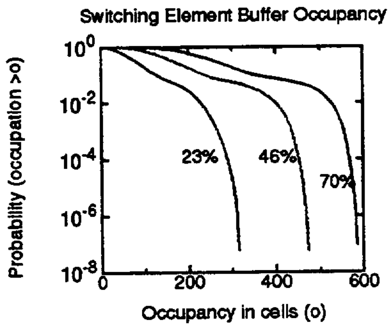


Fig. 10: 256X256, 2 stages of 16X16 elements

Figure 11 shows the results for a switching fabric with back pressure internal flow control and an internal bandwidth expansion to 320 Mbit/s. Eight stages of 2 X 2 switching elements are used. Back pressure produces accumulation in the input network adapter (the output network adapter is much less loaded). The present implementation of this adapter provides a buffer of 85 cells and is not sufficient to guarantee a low cell loss probability at 70% load. However it is very likely that larger implementations of this type of technology will provide larger buffers and, in addition, will rely on more sophisticated flow control techniques than the simple back pressure technique which suffers from "head of line blocking". For example per VC queuing in a shared buffer might be implemented.

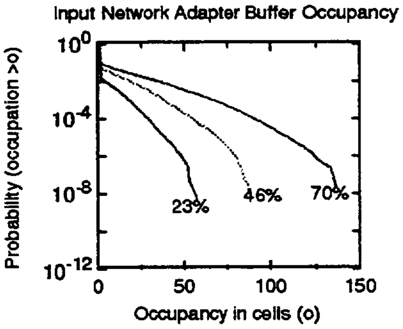


Fig. 11: 256X256, 8 stages of 2X2 switching elements
Back pressure flow control.

Finally Figure 12 shows the buffer occupancy probabilities in the case of the Alcatel architecture. In our simulation, the switching elements are 16 X 16 and the shared buffer memory contains up to 256 slots of 64 bytes. One can see that, in this case also, the probability of buffer overflow is very small for a load of 70%. In the latest implementation of switches based on this architecture, larger buffer sizes are provided.

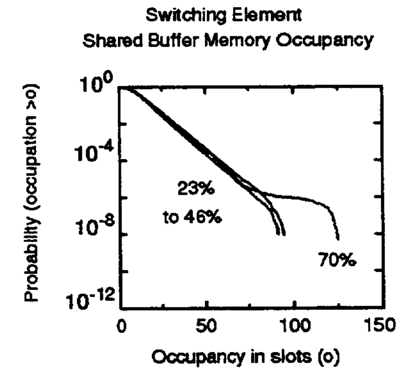


Fig. 12: 256X256 ALCATEL type switch.

In our models these architectures were deliberately conceived with conservative assumptions. Nevertheless, they show that high event building throughput can be achieved in a large switch with a simple standard traffic shaping. Large switches that will be provided by industry will be based on more efficient traffic flow control techniques and will implement larger buffers, resulting in even better traffic characteristics.

VI. CONCLUSION

We have shown on small scale event builder demonstrators that high throughput can be achieved, with no data loss, if one uses a simple traffic shaping technique (rate division) that is provided as a standard in network adapters.

Our models of large switching fabrics (at least up to 256 X 256 at 155 Mbit/port) show that the event building traffic can easily be supported by current technology for aggregate loads up to 70%. Furthermore, the switches that will be available from industry will probably provide even more secure conditions concerning congestion avoidance.

References

[1] M. Costa et al., "An ATM-based event builder test system", Proceedings of the First Workshop on Electronics for LHC Experiments, Lisbon, Sept. 11-15, pp 340-344.

[2] M. Costa et al., "Results from an ATM-based Event Builder Demonstrator", IEEE Trans. on Nucl. Sciences, Vol. 43, No. 3, June 1996, pp 1814, 1820.

[3] Henrion, M. et al., "Technology, Distributed Control and Performance of a Multipath Self-Routing Switch", in Proceedings of the XIV International Switching Symposium, Yokohama, Japan, October 1992, vol. 2, pp. 2-6.

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

[5] C. Paillard, "An STS-OC3 SONET/ STM-1 SDH ATM Physical layer implementation and Application to an ATM Data Generator", RD-31 note 95-04 February 1995.

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

[7] Transwitch Corp., Shelton, Connecticut, USA, SARA chip set, Technical Manual, version 2.0, Oct. 1992.

[8] Creative Electronic Systems SA Geneva, ATM 8468, PCI-ATM Mezzanine Card, DOC 8468/PG, Version 1.0, May 1966.

[9] IDT Inc., Santa Clara, CA, USA, IDT77201 NICStAR chip, User Manual Vers. 2.0, November 30, 1995.

[10] Creative Electronic Systems SA Geneva, RIO2 8060, PowerPC based RISC I/O Board, Technical Manual vers. 1.0, DOC 8060/UM, October 1955.

[11] The ATM Forum, "ATM User Network Interface Specification", Version 3.0