Support for Customising ATM Control

S. Rooney*, J.E. van der Merwe, I.M. Leslie and S.A. Crosby
University of Cambridge
Computer Laboratory
New Museums Site, Cambridge CB2 3QG, U.K.
E-mail: {*jgr20,jev1001,iml,sac*}*@cl.cam.ac.uk*
Telephone: 44 1223 334650 Fax: 44 1223 334678

January 14, 1998

1 Extended Abstract

1.1 Motivation

The need for a single network supporting a large number of diverse services is one of the major factors driving networking research. New services, e.g., video-on-demand, require precise guarantees from the network both in regard to the timeliness of the arrival of their data and its correctness. There is general agreement that other than by over provisioning, the only way to achieve this is through the reservation of resources on the set of network elements across which the data flows. These resources are usually termed a *connection* and the establishment of a connection is often termed *signalling*.

Two visions of multi-service networks can be distinguished: those which are extensions of telephony [1] and those that are extensions of data transfer between computers [2]. The latter are based on the Internet protocol suite, while the former assumes ATM as the universal network protocol. IP is more flexible and robust in the face of network failure than ATM, while ATM allows more precise guarantees to be made about network resource usage[1]. Recent work [3] has shown how compromises can be reached between the two and this trend can be expected to continue. Indeed our working assumption is that large scale multi-service networks will use both native ATM, when precise guarantees are required, and IP when only a best-effort service is needed.

In this paper, we define a *network control architecture* as the set of policies, algorithms, mechanisms and protocols used to control and manage the various devices in a network. Control in a packet forwarding network involves determining whether and to where a packet gets forwarded. Normally, the control policy is viewed at two different levels of granularity and time scales:

- at the network node level (also called in-band control);
- at the control architecture level (also called out-of-band control).

In an IP network all control is performed at the first level[2]. Each IP router makes a control decision for each arriving packet individually. This has the great advantage of being both simple

*Managing author.

[1] It is interesting to note that the distinction between the two is diminishing as IP becomes more 'ATMized' with the addition of connections and signalling protocols, while features such as frame forwarding in ATM switches are blurring the distinction from an ATM perspective.

[2] While IP networks use out-of-band protocols, for example to exchange routing information [4], this describes the overall state of the network and is not used to influence the forwarding of packets on a short time scale.

and robust, but as any attempt to use a service with temporal constraints such as the MBONE [5] at 'rush hour' will demonstrate, it does not permit much in the way of resource guarantees. ATM and extensions to IP, such as RSVP, have the notion of a connection or a flow. A connection is an abstraction of a set of diverse resources across the network; transmitted packets are associated with connections. In such a control architecture each network node determines whether packets arriving on a connection can be forwarded using only the resources currently allocated to that connection (traffic policing) and on a longer time scale the control architecture determines if and across which nodes the connection should be established (connection routing and call admission control).

The ATM standards have defined a set of control primitives for use between the end-system and the ingress/egress switch [6] and between switches [7]. While the standards recognise that different services have distinct resource requirements for their connections, there is an implied assumption that the out-of-band control policy exercised over those connections can be made common to all services. The standards attempt to define a single ATM control architecture valid for all possibles services within planetary scoped ATM networks. It is impossible to know the precise control requirements of all future services and in consequence ATM control architectures will be subject to constant evolution and revision as new services are introduced with distinct control needs. For example, the control needs of mobile ATM systems is still a topic of research, as they are better understood their integration with the single fixed standard ATM control system will require further revision to the standards. Even if it were possible to define all the control needs of all possible services within a single control architecture, it is arguable whether it would be appropriate to do so; such a control architecture would be extremely large and complex, and all services would have to pay the price for its general nature. For example, implementing complex error recovery strategies incurs a significant overhead, some services might prefer to trade greater efficiency against robustness, but all services will have to pay the price for the needs of the most demanding service. Finally, the standards attempt to avoid influencing the nature of their implementation, by saying nothing about the interaction between the controlling plane and the physical switch. This arise from a belief that this is forcibly switch-specific and properly the concern of the switch vendors. Currently, ATM switch vendors propose an integrated approach where both the cell forwarding and connection establishment functions are sold as one complete package. This obscures the fact that they are to a large extent independent and imposes architectural decisions on network operators. Although the ATM standards do not require this integrated approach, neither do they prescribe it; no standard interface is defined between the control and switching plane, allowing switch vendors to implement proprietary solutions. The consequence of this is that switch control software is difficult to port from one switch to another and is impossible for a third party to maintain and evolve.

The unsatisfactory nature of the ATM standards' will inevitably impede the wide scale deployment of ATM. There is a need for control systems which are less monolithic and rigid than those proposed by the ATM standards.

1.2 Contribution

Recently within the networking research community there have been a realisation that the control architectures for multi-service networks need to be more flexible. The open signalling concept [9, 10] attempts to loosen the association between the control plane and the switching plane allowing both to evolve independently. Open signalling control architectures can therefore be much more readily deployed and upgraded than existing ones.

Open signalling has proposed to formalise the separation between the switch controller and the switch fabric using a switch independent interface, thereby allowing both levels of control to be developed and evolved independently. It is envisaged that this will allow more flexible ATM control and the faster introduction of new services. Our early work on open signalling led us to the realisation that some of the basic assumptions regarding switch control could be rethought.

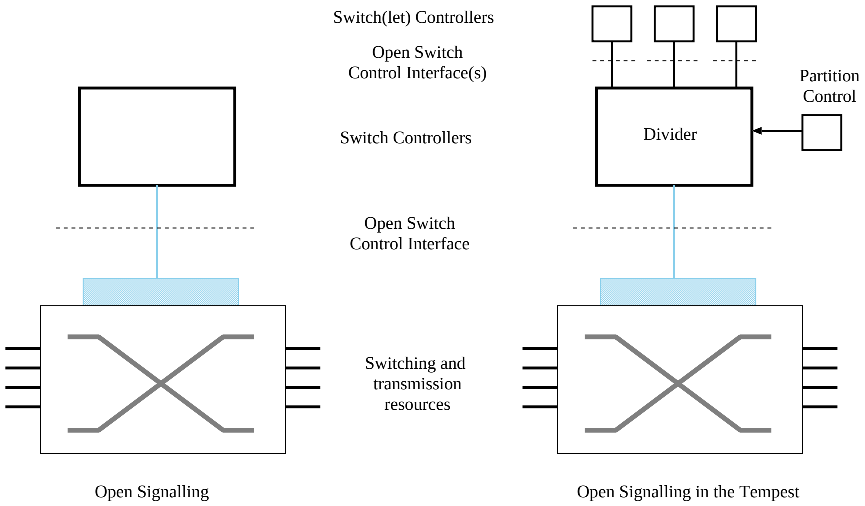

Figure 1: Open Signalling and Open Signalling in the Tempest

This paper describes the necessary infrastructure for allowing multiple control architectures to run simultaneously over the same physical ATM network. We call the set of required components *The Tempest* [11, 12]. The Tempest depends on our ability to simultaneously control a given ATM switch with multiple controllers by strictly partitioning the resources of that switch between those controllers. We call such a partition a *switchlet* as to all intents and purposes the switchlet appears as a small, but complete, switch to the controller. The set of switchlets that a controller or group of controllers possess forms its *virtual network*. The distinction between the Tempest approach and open signalling generally is shown in Figure 1.

Ariel is a switch control interface which can be used for open signalling. This basic functionality has been extended by a crucial component of the Tempest framework, namely the Prospero Switch Divider [11]. Prospero communicates with the Ariel server on the switch and as such is solely in control of the switch as a whole. It then partitions the switch resources into switchlets. The Prospero Divider exports an Ariel control interface for each switchlet created in this fashion. Figure 2 depicts this arrangement.

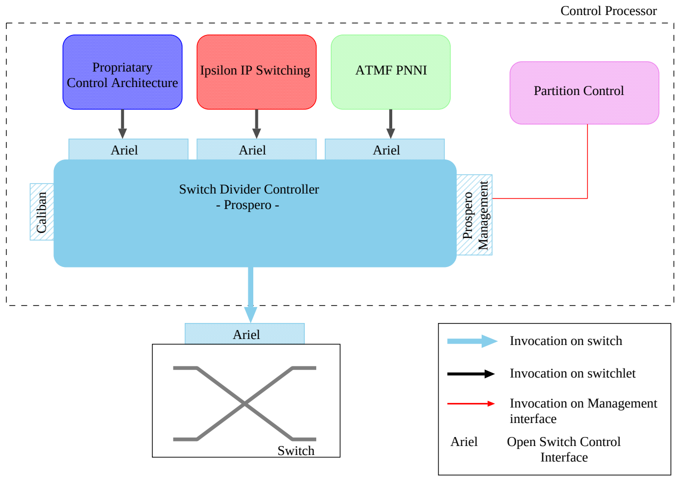

Figure 2: Creating switchlets

The Tempest allows us to run both standard and non-standard control architectures on the same network. Third parties may lease a virtual network from the Tempest network operator and control it in the way best suited to their needs; our work [13] has identified the advantages of *service-specific* control architecture dedicated to one type of high-level service e.g. video conferencing. At an even finer level of granularity we have shown how end-users can dynamically associate code with network resources in a running control architecture [14] in order to achieve *application-specific* control. We call the association of the user defined control policy and the allocated network resources a *connection closure*.

The running of multiple control architectures simultaneously is a new idea and techniques for ensuring the general health of the network need to be rethought. On-going experiments have been looking at issues as diverse as the application of mobile software agents moving between network nodes in order to achieve more adaptive management [15] and the use of measurement based effective bandwidth estimation to perform resource management at various levels of granularity in our framework.

The Tempest frees network operators from having to define one single unified control architecture capable of satisfying all possible future services control requirements. It also allows operators the freedom to differentiate their service offerings in a proprietary fashion, while maintaining a base level of interoperability with other operators where this is required. At the same time it allows "end-users" to become network operators, allowing them to make decision not only about the resources allocated to their connections, but also the way that those connections themselves are established and managed. The Tempest changes the question from, *How should I implement my service with this control architecture ?*, to, *What is the best control architecture for my service ?*.

In summary, the Tempest work is premised on the fact that there will be many different control architectures on a widely deployed multi-service network. It demonstrates how this can be elegantly achieved through the switchlet concept. Further, it shows how end-users can take advantage of switchlets to program their network either by writing dedicated service specific control architecture or by the use of control architectures which are capable of integrating application specific code dynamically.

References

[1] ITU-T, "Draft Recommendation Q.2931, Broadband Integrated Service Digital Network (B-ISDN) Digital Subscriber Signalling Systems No. 2, User-Network Interface layer 3 specification for basic call/connection control," *ITU publication*, 1994.

[2] L. Zhang, S. Deering, D. Estrin, S. Shenker, and D. Zappala, "RSVP: a new resource ReSerVation protocol," *IEEE Network*, vol. 7, pp. 8—18, September 1993.

[3] P. Newman, G. Minshall, T. Lyon, and L. Huston, "IP Switching and Gigabit Routers," *IEEE Communications Magazine*, vol. 35, pp. 64—69, January 1997.

[4] J. Moy, "OSPF Version 2," *RFC 1583*, March 1994.

[5] S. Keshav, "Experience with Large Videoconferences on Xunet II," in *INET'94/JENC5*, 1994.

[6] ATM-Forum, "ATM User-Network Interface Specification - Version 4.0, (UNI 4.0)," *The ATM-Forum: Approved Technical Specification*, July 1995. af-sig-0061.000.

[7] ATM-Forum, "Private Network-Network Interface Specification - Version 1.0 (P-NNI 1.0)," *The ATM Forum: Approved Technical Specification*, March 1996. af-pnni-0055.000.

[8] T. Chen and S. Liu, "Management and Control Functions in ATM Switching Systems," *IEEE Network*, vol. 8, pp. 27–39, July/August 1994.

[9] A. Lazar, "Programming Telcommunication Networks," *IEEE Networks*, vol. 11, pp. 8–18, September/October 1997.

[10] P. Newman, "Ipsilon's General Switch Management Protocol Specification Version 1.1," *Internet RFC 1987*, August 1996.

[11] J. van der Merwe and I. Leslie, "Switchlets and Dynamic Virtual ATM Networks," *Integrated Network Management V*, pp. 355—368, May 1997.

[12] S. Rooney, "An Innovative Control Architecture for ATM Networks," *Integrated Network Management V*, pp. 369—380, May 1997.

[13] J. van der Merwe and I. Leslie, "Service Specific Control Architectures for ATM," *Accepted for Publication in: IEEE JSAC 1998*, 1998.

[14] S. Rooney, "Connection Closures: Adding application-defined behaviour to network connections," *Computer Communication Review, 27(2):74—78*, vol. 27, pp. 74—88, April 1997.

[15] D. Halls and S. Rooney, "Controlling the Tempest: Adaptive management in advanced ATM control architectures," *Accepted for Publication in: IEEE JSAC 1998*, 1998.