Felicia Holness and John Griffiths

Multiprotocol Label Switching within the Core Network

The phenomenal growth of Internet traffic is attributed to the increasing acceptance of the Internet protocol (IP) as the predominant protocol in enterprise networks. The traditional design of a router upon which the Internet is based is becoming unable to support the current demand from users. This service has worked exceptionally well for non real-time traffic such as electronic mail but for real-time traffic such as voice and video, IP has only performed well across a lightly loaded network. To provide services for real-time traffic, new classes of services are being introduced into the Internet.

For networks to be able to support this increased user demand for more bandwidth and for multimedia and real-time traffic, IP routers need to be faster and to provide quality-of-service (QoS) guarantees. This demand for more bandwidth and QoS support has fuelled interest in the use of asynchronous transfer mode (ATM) as an underlying link-layer technology in the Internet. The aim is to exploit the potential benefits of ATM while maintaining the inherent strengths of IP.

Hop-by-hop routing normally works well for non-delay sensitive data traffic but, in cases where prioritisation is required, it gives unacceptable performance since the current Internet is unable to differentiate between different sorts of packets, and thus stipulate that the network provide designated levels of servicing. Since voice and video are delay sensitive they will add additional pressure to the routers. Additionally the increase in traffic is creating the need for higher bandwidth which the packet-forwarding model of router processing cannot provide effectively. The goal would be to expedite movement of information without forcing users to abandon router-independent protocols. Ideally, one would want to move the forwarding function of routing all the way out to the periphery of the network, leaving a protocol-independent network core that would be focused on providing reliable high-performance standards-based connections.

The emergence of the multiprotocol label switching (MPLS) architecture[1] aims to improve the scalability and performance of the prevalent hop-by-hop routing and forwarding across packet networks. Its primary goal is to standardise a technology that integrates the label swapping forwarding paradigm with network layer routing. This label swapping is expected to improve the flexibility and robustness in delivering new routing services without alteration to the forwarding paradigm. The word* multiprotocol *suggests that this architecture will be applicable to any network-layer protocol in addition to IP; label switching indicates the underlying forwarding mechanism. This paper will start with a basic tutorial on IP switching techniques and then explain how it can be utilised to accelerate IP packets through the core of the network. The paper concludes with the description of a proposed architecture, quality-of-service label distribution protocol (QoSLDP) whose objective is to provide bandwidth guarantees.

Introduction

An IP switch is a device that can forward IP packets at layer 3 as well as switching packets at layer 2. The mechanisms used by an IP switch enable it to be able to classify packets which will be forwarded at layer 3 and which will be switched at layer 2 and then to redirect those packets over a layer 2 switched path[1, 2].

Felicia Holness:
E-mail: f.holness@elec.qmw.ac.uk
John Griffiths:
E-mail: j.m.griffiths@elec.qmw.ac.uk,
Department of Electronic Engineering
Queen Mary and Westfield College
University of London
London, E1 4NS.
United Kingdom
Tel: +44 171 415 3755
Fax: +44 181 981 0259

The device consists of an IP controller running network layer routing protocols (for example, BGP, OSPF, RIP) on top of a switching component. Most devices use an ATM switching fabric where the ATM switch maintains a table of input ports, input labels, output ports and labels like a normal ATM switch. There are two distinct functional components to an IP switching system: a control component and a forwarding component. The control component uses standard routing protocols to build a forwarding table. On receiving packets, the forwarding component searches the forwarding table to decide on the next hop for the packet, and switches the packet from the input interface to the output interface.

The ingress and egress components, otherwise known as a *label edge router* (LER), are positioned at the edge of an IP switching system; they can consist of software residing in a router. The concept of an IP switching ingress and egress is as follows: assume that the IP switching system decides to forward some packets over an established end-to-end layer 2 switched path. Packets arriving at the ingress are classified for either default layer 3 IP forwarding or layer 2 switching based on IP layer criteria such as IP source/destination addresses. Packets that fall into the layer 2 classification are forwarded over the established layer 2 path, while all other packets are forwarded at layer 3 using normal IP forwarding procedures. Upon arrival at the egress, either at layer 2 or layer 3, they will be passed to layer 3 for normal IP processing and forwarded to the destination.

In conventional IP routing, packets travel from a source to the destination by hop-by-hop forwarding through routers. At each router hop the IP header is examined and the next hop decision is made on a number of functions such as the destination address, TTL counter, and so on. This introduces delay for packets traversing the path.

The layer 2 switched path bypasses intermediate layer 3 hops from source to destination; this path is called a *cut through* or *short cut* path[3]. The basic functions of a cut through path are:

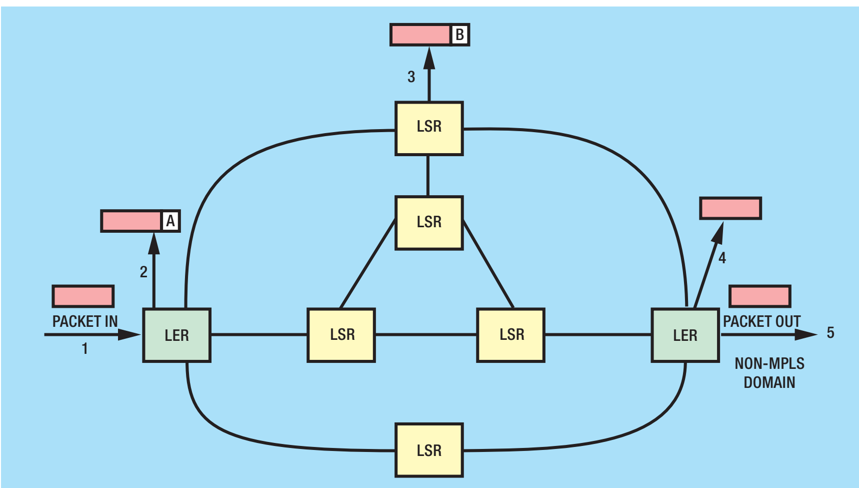
- it bypasses intermediate layer 3 routing;
- if the cut through is removed, packets can still be forwarded by conventional hop-by-hop routing;
- it can be point-to-point, point-to-multipoint or multipoint-to-point; and
- a cut through path can be built based upon data or control traffic.

Label Distribution Protocol

The main component of the multiprotocol label switching (MPLS) architecture is the label distribution protocol (LDP)[4, 5]; it is defined as a set of protocols by which a label switch router (LSR) communicates with an adjacent peer by means of exchanging labels. The LDP protocol is a set of procedures and messages exchanged and maintained between adjacent LSRs in the process of establishing label switch paths (LSPs) through a network by mapping network layer information to a label representing a data-link layer path. Each LSP has a forwarding equivalence class (FEC) associated with it; at the ingress of an MPLS domain all incoming packets will be assigned to one of these FECs. The classification of packets into specific FECs identifies the set of packets, which will be mapped to a path through the network. Examples of FEC elements are a complete host address or an address prefix.

The LDP protocol has four types of messages, discovery, session, advertisement and notification. The cumulative operation of these messages allows LSRs to indicate their presence in the network to establish and maintain sessions between peers and to perform operations on the labels they exchange between themselves. A label is a short fixed-length identifier used to identify a packet assigned to an FEC. Adjacent LSRs agree on the label that will be used to represent an FEC flowing in a particular direction; packets belonging to that FEC will have this label appended to it. The label can either reside within the data link header or exist between the layer 3 and layer 2 headers. The downstream LSR makes the decision to bind a label L to a particular FEC F and sends this assignment to the upstream LSR. This label is then used by the upstream node to forward all packets belonging to that FEC to the downstream node. It is these mechanisms of label assigning and distributing that are employed in an MPLS network. Figure 1 shows a simple example of an MPLS domain. (1) An unlabelled packet reaches the ingress of an MPLS domain, where the ingress node (2) examines the network layer header and decides which FEC to assign it to. Once this decision is made the packet is assigned a label representing the path it should take. The labelled packet is then forwarded to the next LSR within the MPLS domain. When it is received at (3) the LSR examines its labelling information base and uses the label to determine the next hop for the packet, and the new label that should replace the incoming label. The packet is then forwarded to the egress node using the newly assigned label. At the egress node (4) the LSR removes the label, examines the network layer header and forwards the packet from the MPLS domain into the non-MPLS network.

Figure 1—MPLS domain



Partnership of MPLS and ATM

This section outlines a number of scenarios in which MPLS and ATM can coexist; it explains the problems with the various implementations and direction in which this research will take.

It is possible for MPLS to operate over any data link layer, but ATM is by far the most attractive technology because it is the only link layer protocol that can transport video, voice and data with the offer of a defined QoS.

There are a number of different ways in which ATM[6, 7] can support MPLS; the first is as a peer model. An ATM switch is converted into an LSR called an *ATM-LSR* that runs an instance of the LDP protocol and network layer routing protocols, and is capable of forwarding packets at layer 3 and label swapping at layer 2. ATM-LSRs are directly connected to each other and they exchange routing protocols and LDP messages along serial links. In this instance, all ATM Forum protocols have been removed and the switches use LDP to establish LSPs through the network. The peer MPLS/ATM[3] implementation places less routing burdens on the system than an IP/ATM overlay model because each ATM-LSR only forms routing adjacencies with its neighbour, and not each router at the other end of a VC. However, in this model the loop-free QoS-based path selection and the native QoS support is lost. If one is not exploiting ATM QoS this model might seem to defeat the initial objectives.

The second example is the integrated approach in which ATM-LSRs communicate with each other through native ATM switches. ATM-LSRs may establish PVCs or SVCs using native ATM signalling to send IP traffic, control information or LDP messages over. When a stream is mapped to a VC, an identifier, VCID, is associated with it if it passes through intermediate switches. The integrated approach allows MPLS to be integrated into an existing ATM network[8].

In the third model the physical ATM model is divided into two topologies, one maintained by PNNI and the other by IP and MPLS. This requires the partitioning of the VPI/VCI label space into MPLS and ATM portions. Additionally, some ports may be configured for MPLS and others for ATM. Switch resources are then under the control of coresident, but mutually exclusive, MPLS and ATM switch control points. The switch component would run a VC routing protocol like PNNI. Thus a network employing the integrated approach would support concurrent MPLS and ATM topologies, routing and VC management. Rather than implement the overlay model that could delete VC resources, an installation of this model would be an option. This model also allows the provider to continue to provide ATM services while introducing an additional control plane that is optimised for scalable IP routing.

A number of scenarios in which MPLS can be used in ATM have been described. We have seen how the integration of IP and ATM has allowed IP to route its packets quickly through the core of the network, but the fundamental problem with the current version of MPLS is that it offers no QoS. With MPLS in its current form you would need to use the complex and cumbersome ATM Forum signalling. The authors propose a novel QoS scheme that will enable an IP switching network to reserve resources using LDP as its base.

QoS Label Distribution Protocol

In order to devise a QoS scheme it is essential to detail the component(s) that is deemed necessary. The authors believe that in order to offer more than one service level efficiently, there is the need for the 'reservation of router resources'[9].

QoSLDP expands on the fundamental component of the MPLS architecture—the LDP protocol. An ingress node will receive the application's QoS requirements in terms of ATM's QoS and traffic parameters; that is, cell loss priority, cell delay variation and cell transfer delay. The ingress node will formulate a *reservation request* message to its downstream neighbour with the requested QoS parameters. The node will carry out a connection admission control algorithm to determine whether or not it can accept this request. If it is unable to grant the request it will immediately reply with a *reservation reject* message. Upon receipt of a *reservation reject* message, the recipient will try alternative paths until it is accepted or until all possible paths are blocked.

If an LSR determines that it can grant a reservation request, it will select a virtual connection that can meet the QoS requirements and identify it by assigning a label to it, which it will record in its information base. It will then send the reservation request to its downstream neighbour. The downstream neighbour will determine if it can accept the request; if it can it will assign a label identifying the virtual connection. *Reservation request* messages will continue to propagate through the network until an egress node for the flow has been reached.

When a node realises that it is the egress node for a particular path, it will reserve a VC and send a *reservation accept* message to its upstream neighbour setting a field in the label indicating that this is the egress node for that particular flow.

Upon receiving a *reservation accept* message, intermediate nodes will record the label received from their downstream neighbour alongside the reserved virtual connection.

When the ingress node receives a *reservation accept* message it checks the label for an indication that its request has been received by the egress node before assigning all packets belonging to that flow with the label. An end-to-end virtual connection has been established for that flow to transverse with the requested QoS parameters.

A node that is unable to process a *reservation request* message will not propagate the request message to its downstream neighbour; it will instead send a *reservation reject* message to its upstream neighbour.

A node only propagates a *reservation accept* message to its upstream neighbour when it has received a label indicating it has originated from the egress node for the flow, thereby indicating an end-to-end virtual connection has been established for the flow. In the case where the connection is no longer required a *reservation withdraw* message will be sent to the appropriate routers.

In the QoSLDP architecture, in the event that a node is unable to allocate resources a message is immediately propagated to the initiator. If the router alongside the ingress router is unable to reserve resources it will reply with a *reservation reject* message instantly. In QoSLDP the network layer header is examined only once at the edge of the network and within the network, the routers need only examine a short label to determine where the next hop is.

QoSLDP naturally allows the encapsulation of QoS messages within LDP messaging format.

In the design of QoSLDP we have made the assumption that IP applications at some time in the future will have the ability to signal their QoS needs to the ATM layer, in terms of ATM QoS parameters.

At the moment QoSLDP is still in the experimental phase and as a result of this the authors accept that performance issues will arise. A few of the problems that can be determined qualitatively are scaling issues being introduced when a single VC is used for each flow as the number of VCs in any ATM network is limited. As a result of this the number of QoS flows that can be accommodated by any one device is limited strictly to the number of VCs available to a device. Another primary concern is how to integrate the many-to-many connectionless features of IP multicast into the one-to-many point-to-multipoint connection-oriented realm of ATM.

References

1 ROSEN, E.; VISWANATHAN, A; and CALLON, R. Multiprotocol Label Switching Architecture. draft-ietf-mpls-arch-02.txt, July 1998.
2 NEWMAN, P.; EDWARDS, W.; HINDEN, R.; HOFFMAN, E.; CHING LIAW, F.; LYON, T.; and MINSHALL, G.; Ipsilon's General Switch Management Protocol Specification, Version 1.1, RFC 1987, Aug. 1996.
3 METZ, C. IP Switching Protocols and Architecture. ISBN 9 780070 419537.
4 ANDERSSON, L.; DOOLAN, P; FELDMAN, N.; FREDETTE, A.; and THOMAS, B.

LDP Specification. draft-ietf-mpls-ldp-02.txt, Nov. 1998.
5 Davie, B.; Lawrence, J.; McCloghrie, K.; Rekhter, Y.; Rosen, E.; Swallow, G. Use of Label Switching with ATM. draft-davie-mpls-atm-01.txt, July 1998.
6 Bruce, D.; Lawrence, J.; McCloghrie, K.; Rosen, E.; and Swallow, G. MPLS using ATM Switching. draft0ietf-mpls-atm-01.txt. Nov. 1998.
7 Clark, M. ATM Networks Principle and Use. Santa Clara, California, Frankfurt, Germany: Wiley Teubner 1996.
8 Awduche, D., Malcolm, J.; Agogbua, J.; O'Dell, M.; and McManus, J. Requirements for Traffic Engineering over MPLS. draft-ietf-mpls-traffic-eng-00.txt, Oct. 1998.
9 Keshav, Sharma. Issues and Trends in Router Design. *IEEE Communications Magazine*, 1998.

Bibliography

Dorling, B. *et al.* Internetworking over ATM. An Introduction. Prentice Hall, New Jersey, 1996.

Katsube, K.; Shobatake, Y.; Mogi, A.; Matsuzawa, S.; Jinmei, T.; and Esaki, H. Toshiba's Flow Attribute Notification Protocol (FANP) Specification. RFC 2129.

Marshall, G. Classical IP over ATM: a status report. Data Communiations, 1995, **24**(7). p. 5.

Nagami, K.; Demizu, N.; Esaki, H.; and Doolan, P. VCID Notification over ATM Link. draft-ietf-mpls-vcid-atm-01.txt, Aug. 1998.

Newman, P.; Edwards, W.; Hinden, R.; Hoffman, E.; Ching Liaw, F.; Lyon, T.; and Minshall, G. Transmission of Flow Labelled IPv4 on ATM Data Links. Ipsilon Version 1.0, RFC 1954, May 1996.

Newman, P.; Edwards, W.; Hoffman, E.; Ching Liaw, F.; Lyon, T.; and Minshall, G. Ipsilon Flow Management Protocol Specification for IP4, Version 1.0, RFC 1953, May 1996.

Rekhter, B.; Davie, B.; Katz, D.; Rosen, E.; and Swallow, G. Tag Switching Architecture. http://www.cisco.com/warp/public/732/tag/tagsw_ov.htm

Biographies

Felicia Holness
Queen Mary and Westfield College, University of London

Felicia Holness is currently studying for a doctorate degree at Queen Mary and Westfield College, University of London, which is being sponsored by British Telecommunication.
Prior to joining the University of London, Felicia obtained a BEng Honours Degree in Electronic and Electrical Engineering and a Masters of Science in Satellite Communications both at the University of Surrey.

John Griffiths
Queen Mary and Westfield College, University of London

John Griffiths graduated in Electrical Engineering at Manchester University. He joined the Post Office (later to become BT) Research Laboratories working on early PCM transmission systems, CATV and ISDN. Until 1994 he was manager of the Core and Global Networks Division at BT Laboratories at Martlesham. He is now Professorial Fellow at Queen Mary and Westfield College of the University of London, working on various aspects of modern broadband networks.