Levels of Aggregation in Flow Switching Networks[1]

Tom Worster and Avri Doria[2]
General DataComm, Inc., 5 Mount Royal Av., Marlborough, MA

Introduction

One of the principal design considerations in a flow switching network is the definition of a *flow.* Flow switching uses a network of connection oriented switches that allow flows to be cut-through on their own connection. The granularity of a flow can vary over a wide range from the very fine, for example an individual application level transaction, to the very coarse, such as the aggregated traffic on the next hop between adjacent routing nodes. There are technical trade-offs to be considered together with the constraints of current technology when addressing the question of the appropriate level of aggregation. It appears that all the levels of aggregation have their place in flow switching and that the aggregation level should be a parameter available to network designers for optimization.

Flows and flow switching networks

Flow switching networks are characterized by the combination of conventional store-and-forward routing and a connection oriented, cut-through switching network such as an ATM or frame relay network[3]. The routers are made aware of the topology of the cut-through network by, for example, locating one router with each cut-through switch and interconnecting the routers with default channels that are congruent with the links in the switching network. The network of routers may perform store-and-forward routing as well as running dynamic routing protocols but these functions are augmented with the capability to establish cut-through connections in the network of switches. These connections are used to carry the traffic of persistent network *flows*. Once a flow has been redirected onto its cut-through the routers are no longer burdened with forwarding the flow's datagrams. This paper concerns the definition of such flows.

The most general definition of the term flow in this context is somewhat unhelpful: a flow is that which is cut-through on its own connection. However, since the purpose of declaring a flow and cutting it through is to save the cost of making forwarding decisions, a flow ought to be a sequence of datagrams that, in the absence of topology changes, take the same route though the network. To define flows more specifically we must first distinguish the various flavors in which

---

1. To be presented at the Electronics Industries Forum, Boston May 6-8, 1997.
2. {worster, avri}@gdc.com
3. To date most of the industry's emphasis in flow switching has been on ATM although the general models are applicable to almost any connection oriented network technology.

they come. The first distinction we can make is between flow definitions that are application based, topology based, and policy based [1].

Very often the applications will be best placed to make an *a priori* determination as to whether or not a flow will be persistent. For example, a web server knows the size of all the files available for download and it could discriminate flows based on that parameter. Application driven flows require a protocol with which the applications may request connections[4] but wide deployment of such a protocol is some way off in the future.

Topology based flows, such as those in Cisco's Tag Switching [3], IBM's ARIS [4, 5], and Cascade's IP Navigator [6], are established in the course of running the routing protocols. In the process of computing routing tables the routers establish a network of cut-through connections that correspond to the network's routes. It is not simple for a router to determine in advance what the expected usage of a route will be so a cut-through per route is necessary. Such an approach might be suitable for a network in which it is expected that a large proportion of all possible routes will carry a significant amount of traffic.

We describe flows that are established by the network nodes in response to the usage of routes as being established by a network policy. Examples of this approach are Toshiba's Cell Switch Router [7] and Ipsilon's IP Switching [8]. The definition of flows based on network policy provides a simple migration path to a flow switching network since the flow discrimination and switching is transparent to hosts and applications. Flows are usually defined according to heuristics set in the network by which network elements are able to discriminate flows. An example of such a policy might be that a flow is declared whenever more than ten datagrams are seen on a route with the same source and destination addresses within a period of one minute. Policy defined flows usually expire by an automatic time out mechanism if they are not used.

The design of heuristics for policy based flow discrimination is not trivial. Many factors can be introduced including fields of the datagram header, counters and timers with parameters that may depend on the header fields, time of day, resource depletion and many other factors. Flexibility in the design of the flow discriminator is highly desirable since the characteristics of IP and Internet traffic evolve through time so that a policy that works well today may not work so well in a few months. Self optimizing algorithms for determining policy are not inconceivable especially if they are able to track the non-stationary characteristics of IP traffic. We can expect to see research papers on these subjects in the near future.

Whether a topology or policy based flow definition leads to a more economical use of cut-through connections will depend on the characteristics of a network. In a large campus network it is unlikely that a significant portion of the full enumeration of host pairs will communicate directly; more commonly hosts will correspond with a limited number of servers and gateways. In other words, the traffic matrix in such a network will be sparsely populated. Alternatively, in the core of a virtual private network bounded by the interfaces between the public service provider and customer's sites there are a relatively small number of sources and destinations and the traffic matrix may be more uniformly populated.

---

4. RSVP, the Resource Reservation Protocol [2] is a possible example of such a protocol.

Flow granularity

One of the principle questions in defining flow policy is how coarse or fine the granularity of the flow should be. Granularity refers to how many communications share the same cut-through: in a finely granular flow very few or only one communication shares a cut-through and a coarse or aggregated flow carries the datagrams of very many communications. The following list is a selection of possible policies that produce different levels of granularity:

Source and destination addresses and port numbers. A flow that only carries the datagrams between a unique pair of hosts and a unique pair of sockets might represent, for example, a single TCP connection. A web server would typically consume several flows to present one web page since each page is usually composed of several files.

Source and destination addresses and destination port number. This is a flow definition specifically intended for www traffic. Once a flow is created from a server to a browser it will be used for all the files accessed from that server until the flow expires through lack of use.

Source and destination addresses. Without port numbers in the flow definition all datagrams between a pair of hosts will share the same flow. In client-server networking this can save a lot of flows, but if only one application is using the network then the aggregation level is the same as with the previous example.

Network routes. A mesh of connections may be established that correspond to the routes in the network. These may be point-to-point connections in networks that are small enough that the ensuing number of connections is tolerable, or multipoint-to-point connections in which one connection corresponds to each destination.

Routes to egress routers. The next hop routing protocol (NHRP [9]) uses the concept of a connection to an egress router. Given a destination address, a next hop router returns the address of the egress router to which a cut-through connection may be established. This cut-through may be used for all traffic that shares the same egress router.

Source and destination subnetwork address. Flows between subnetworks can aggregate the traffic of a number of hosts into a single flow. If hierarchical addressing is used in conjunction with classless interdomain routing (CIDR [10]) then the level of aggregation can be controlled by selecting the length of the network prefix.

Next hop routes. In the extreme of aggregation all the traffic that shares the same next hop could be put onto the same cut-through. At this point the benefit of using cut-throughs has disappeared since the next hop router will have to examine the datagram to be able to make a forwarding decision.

Comparison of aggregation levels

When considering the advantages and disadvantages of any aggregation method one has to bear certain general points in mind:

Motivations

- The basic motivation for flow switching is to simplify forwarding. For example, with ATM switching hardware, once the cut-through connection is established there is no per-datagram processing cost for forwarding user data.
- A secondary motivation for flow switching is the performance benefits experienced by users through the use of flow switching technologies such as ATM. An end-to-end ATM connection can potentially deliver lower latency than store-and-forward routers as well as higher throughput. Such capabilities allow the deployment of high quality, real time multimedia and streaming data services.
- The ability to provide quality of service commitments on throughput and delay is an important goal for the IP community and flow switching, since it is based on a connection oriented network technology, should be able to provide the network mechanisms to support such commitments. A flow switching proposal should therefore regard quality of service as a major goal.
- An associated motivation is that once flows are switched with ATM hardware the other capabilities of ATM can be utilized to the benefit of the network and its users. These capabilities include hardware multicast, traffic monitoring (i.e policing), priority queueing or per connection queueing, buffer and/or bandwidth reservations, and accounting capabilities (although internet protocols that are able to use some of these capabilities are yet to be developed). In a flow switching network that extends close to or even includes the hosts, it will often be possible to provide end-to-end delay and throughput commitments[5].

Limitations

- One basic limitation of the use of cut-throughs is that cut-through connections are a limited resource. The cell switching hardware of ATM uses connection tables to translate and forward incoming ATM cells. The memory space for these tables is limited—a typical number of connections for a 155 Mbit/s link is several thousand and not several tens of thousand.
- The processing cost of establishing and removing cut-through connections is also a limiting factor. Processing capacity is spent to install, maintain and remove the connection state in the router's local switching hardware and then again to advertise the fact to the rest of the network. The protocol used to advertise cut-through state is equivalent to the signalling protocols used in telephony or B-ISDN networks. In the case of IP Switching the protocol is called IFMP and runs only between a node and its nearest neighbors to synchronize the mapping of IP level flows to ATM labels (i.e. VPI/VCI value) [11]. In Tag Switching the signalling protocol is called the Tag Distribution Protocol [12] and it begins at the network ingress points and extends along the network's routes.
- Link bandwidth is another limiting factor. Although it is not new to flow switching, the fact that flow switching is used may allow link bandwidth to be increased. In ATM switches there is often a relationship between the connection table size and the link bandwidth. This is based on the old B-ISDN rationale that ATM is a broadband technology and therefore ATM connec-

5. This is possible if the non-flow switched network local to the host is engineered to not be a bottleneck.

tions with very low bandwidth are somewhat pointless or, at the very least, to be discouraged. Hence there is no need for tens of thousands of connections on a 155 Mbit/s link. This assumption may need revision in the flow switching application.

- Another more subtle limiting factor can be found in the dynamics of the policy based flow switching mechanism. In simple terms, efficiency is reduced if the latency in detecting a flow and establishing its cut-through is a significant fraction of the flow's duration. For non-real time applications the ratio of concern is the fraction of bytes sent before and after cut-through, rather than the temporal fraction.

The need for aggregation

Research indicates that traffic in the Internet core is very highly multiplexed. The degree is somewhat surprising: analysis by Newman [8] on measurements from a network access point (the Mae West NAP) suggest that, in a time window of one minute, the number of individual host-to-host communications per Mbit/s of data is around 1400. This number corresponds approximately to the total number of connections needed to flow switch the traffic if the connection expiry timer is set to one minute. This definition of flows is based on a source and destination address and the flow is declared when the first datagram of the source-destination pair is seen. With Ipsilon's policy based flow definition this is clearly not a suitable heuristic but it does serve to indicate the degree of multiplexing seen in the core Internet. Newman's research suggests that a suitable heuristic would be to declare the flow after a number, n (with $10 < n < 20$), of datagrams are seen between the source-destination pair within a time interval of one minute. With the parameter n set to twenty the total number of connections comes down to about 300 flows per Mbit/s. For 155 Mbit/s links the total number of connections required by this heuristic is about an order of magnitude more than the average ATM switch offers.

Newman's work also suggests that the dynamic connection set-up rate required to support these flows is about 2 cut-throughs/s per Mbit/s. If the IP switching node has a distributed control architecture such that the signalling capacity of the node scales with the number of ports, then 2 cut-throughs/s per Mbit/s might be tolerable. However, today's IFMP processors are centralized and handle several hundred cut-throughs/s—not several thousand.

It can, perhaps, be argued that measurements at a NAP like Mae West are unrepresentative of average traffic conditions in the Internet but the evidence is that current technology lies about an order of magnitude behind the requirement for host-to-host policy based flows in the Internet. In these situations a more aggregated flow definition is clearly required.

Characteristics of aggregate flows in a flow switching network

For the sake of discussion we postulate a simple environment for aggregation: a flow switching network at the core of an Internet that has conventional IP external interfaces (i.e. not flow switched interfaces), flow switched internal interfaces, gateway functions at the edge nodes and a manager of this network with the power to choose an appropriate aggregation policy. (We also assume that the network equipment allows the choice of aggregation level.) In this example a high

level of aggregation would probably be chosen—perhaps a mesh of flows based on routes to network prefixes would be appropriate.

Now, assuming that the number of flows is manageable, how far does this model meet the goals discussed above for flow switching? The goal of simplifying forwarding has been met, at least within the network, but conventional routing techniques have to be applied at the edges. On the ingress links the datagrams have to be examined to determine their flow. The highly aggregated flows will have to be split up at some stage even if this is external to the flow-switched network.

As far as improving the quality of service perceived by users and applications, this scenario has not brought the benefits of end-to-end flow switching due to the store-and-forward techniques and segmentation and reassembly used at the network edge.

To utilize the differentiated quality of service classes provided by ATM's priority queueing hardware in this aggregated flow switching scenario requires the establishment of more cut-through connections. The total number requires it proportional to the number of service classes— twice as many service classes required double the number of cut-throughs. Non aggregated flows, on the other hand, can have the service class individually assigned and supporting large numbers of service classes does not affect the number of cut-throughs.

A related problem exists while trying to support multicast groups. The aggregated flows in this network correspond to a superposition of point-to-point communications that happen to share the same route across this network. That there is a strong probability that many communications will use the route is the reason for setting up the aggregate flow. IP multicast trees, on the other hand, are less likely to be congruent over the whole network. It is desirable that underlying hardware multicast capabilities, such as those of ATM, could be used to support IP multicast trees and ATM can do this if a multipoint-to-multipoint connection is established to support a given IP multicast tree[6]. But such a multipoint-to-multipoint connection can only be used to support additional multicast trees if all these trees have identical topology within the flow switching network. So although multicast can be supported in the network, the multicast trees will not be aggregated together in the same way as point-to-point traffic.

The deployment of flow switching in this scenario has brought some direct, cost saving benefits to the network provider. Although cost savings can be passed on to the network's customers, they do not necessarily perceive any direct benefits. Aggregation was necessary to control the number of flows since the postulated network is in a core Internet, but aggregation worked to undermine the other goals of flow switching.

Aggregation as a partial strategy

If the network design is to achieve the goals of using ATM technology to provide quality of service commitments on delay and throughput, real-time, and broadband capabilities then fine grained cut-throughs will have to be supported. Since we have acknowledged that aggregation

6. This assumes that the cell interleaving problem of using AAL5 together with ATM channel merging is resolved one way or another.

will be necessary in some networks the question becomes: how can coarse and fine grained flows be handled at the same time?

As far as the underlying switching mechanism is concerned the content of a flow is irrelevant. An ATM network only handles the connections which among them may carry highly aggregated flows at the same time as fine grained flows, or any other kind of flow. It is entirely within the responsibilities of the layer three routers that control the ATM network to determine what is appropriate for cut-through. So if the layer three routing is able to aggregate most traffic but determine when a fine grained flow is required then both kinds of flows can be handled at the same time.

The approach of switching fine grained and coarse grained is desirable at least in the short term since the deployment of applications capable of using and justifying quality of service commitments is unlikely to displace best effort traffic very soon. So with a technology that can switch both fine and coarse grained flows and can allow the network manager to set the aggregation policy, the deployment of flow switching can allow cost benefits for existing networks at the same time as providing a platform for the roll out of new IP applications and protocols that can benefit from quality of service commitments.

IP Switching currently uses three flow types numbered zero, one and two [13]. The type zero flow is used to connect the routers with their immediate neighbors and they carry aggregated next hop traffic that is forwarded using the routers. Types one and two flows represent application-to-application and host-to-host cut-through flows respectively. With type one flows the source and destination IP addresses and the source and destination TCP/UDP port numbers are bound to the flow creating a very fine grained flow. Type two flows are bound only to the addresses and the port numbers are encapsulated together with the data in the AAL-5 PDU. It has been suggested that by extension one could define a type three flow in which additionally the source and destination host addresses are encapsulated and the flow is bound only to the CIDR prefix. Such type three flows could be used to encapsulate data from gateway interfaces running BGP4 while other interfaces could be set up with finer grained flows. This is an example of a technical approach to providing users with the benefits of fine grained flows at the same time as letting the network take advantage of aggregation on a selective basis.

Augmenting and diminishing the aggregation level

The problem with the approach outlined above, of allowing coarse and fine grained flows to coexist in a network, is that it will not scale adequately to a potential future scenario in which quality of service commitments are widely used in large networks such as the Internet. To solve this problem we must find a way to switch fine grain flows end-to-end and aggregate them without losing the quality of service commitment.

If fine grained flow switching extends end-to-end in a network but the network's core uses aggregated flows then a mechanism for augmenting the aggregation level must be placed somewhere in the network. The complementary function that diminishes the aggregation would most likely be located in the same place. The device could tentatively be named audi (from *au*gment/*di*minish).

One way to do this would be to terminate the cut-through connections on the audi and perform a flow discrimination at the datagram level. The flow discrimination policy would be set for coarse and fine grained flows in the augmenting and diminishing directions respectively. To borrow a name from a sublayer of AAL5, this method of implementing an audi could be termed RAS[7] (reassembly and segmentation) since the datagrams would be reassembled from the cut-through connections and segmented again once the flows are reclassified. Since flows terminate at a RAS audi the knowledge of which flow a datagram belonged to is lost in the subsequent network.

A RAS based audi is unattractive since it will cause delay and will be relatively expensive compared with ATM switching. It may also become a bottleneck and as such present a problem in allocating network resources. Such an audi would require special hardware so it could not easily be turned on or off or be relocated in the network through as an aspect of network configuration.

An alternative the RAS approach is to use a hierarchy of flow levels. This approach, known as label stacking in the MPLS study group [1], takes a number of fine grained flows and merges them together in the cut-through switching hardware. The new aggregate flow is given a label on which it can be switched while the labels of the individual flows are preserved.

One way of doing this with ATM technology is to use the VCI field as the fine grained label and the VPI field as the coarse label. In this way, standard ATM switching hardware can be used to implement the audi—it is only required that the controlling router be capable of determining aggregated routes. A generalized concept of label stacking could involve adding and removing labels at several points along a flow's route, although this is probably not feasible with standard ATM cell switching hardware.

A version of label stacking is already available. An IP network may be implemented using IP Switching and layering this network over a core ATM network using ATM virtual path connections. The VPCs should be established to correspond to the required interconnect topology between the IP Switches. Multiple virtual path interfaces, each of which corresponds to an IP interface in the layer three network, can be established on one ATM interface. This can be established most conveniently by giving the gateway points ATM addresses and using soft permanent VPCs. Soft permanent VPCs have the advantage of requiring the installation of hard state only at the gateway points.

Conclusions

It appears that aggregation of traffic in some IP networks is necessary in order to control the total number of flows but this is not required in all networks. Aggregation tends to negate some of the benefits of flow switching such as hardware multicast, traffic control and quality of service commitments although these can potentially still be provided through if suitable aggregation methods are developed. A combination of aggregate and fine grained flow switching is an attractive compromise solution for the near term but is limited in its ability to support large numbers of fine grained flows. Mechanisms exist for changing the level of aggregation within a network that provides end-to-end switching of fine grained flows but some careful attention has to be paid to the

7. As opposed to SAR function in AAL5.

traffic management issues at the aggregation points. With these thoughts in mind it appears that an implementation of flow switching should allow the network manager to determine aggregation policy and where the aggregation points should be located.

References

[1] R. Callon, P. Doolan, N. Feldman, A. Fredette, G. Swallow, and A. Viswanathan, “A Framework for Multiprotocol Label Swapping,” work in progress, ftp://ftpeng.cisco.com/mpls/mpls, Apr 1997.

[2] R. Braden, L. Zhang, S. Berson, S. Herzog, and S. Jamin, “Resource ReSerVation Protocol (RSVP)—Version 1 Functional Specification,” Internet Draft draft-ietf-rsvp-spec-14.txt, Nov 1996.

[3] Y. Rekhter, B. Davie, D. Katz, E. Rosen, and G. Swallow, “Cisco Systems' Tag Switching Architecture Overview,” RFC 2105, Feb 1997.

[4] A. Viswanathan, N. Feldman, R. Boivie, and R. Woundy, “ARIS: Aggregate Route-Based IP Switching,” Internet Draft draft-viswanathan-aris-overview-00.txt, Mar 1997.

[5] N. Feldman and A. Viswanathan, “ARIS Specification,” Internet Draft draft-feldman-aris-spec-00.txt, Mar 1997.

[6] “White Paper—IP Navigator,” Cascade Communications Corp., http://www.casc.com/products/datasheets/ipnav.pdf, Dec 1996.

[7] “White Paper on Cell Switched Router,” Toshiba Corporation, ftp://ftp.wide.toshiba.co.jp/pub/csr/white_paper.ps.gz, Nov 1996.

[8] P. Newman, G. Minshall, and T. Lyon, “IP Switching: ATM Under IP,” Submitted to IEEE/ACM Transactions on Networking, http://www.ipsilon.com/~pn/papers/newman0001.pdf, Dec 1996.

[9] J. V. Luciani, D. Katz, D. Piscitello, and B. Cole, “NBMA Next Hop Resolution Protocol (NHRP),” Internet Draft draft-ietf-rolc-nhrp-11.txt, Mar 1997.

[10] V. Fuller, T. Li, J. Yu, and K. Varadhan, "Classless Inter-Domain Routing (CIDR) an Address Assignment and Aggregation Strategy", RFC 1519, Sep 1993.

[11] P. Newman, W. L. Edwards, R. Hinden, E. Hoffman, F. Ching Liaw, T. Lyon, and G. Minshall, Ipsilon, “Ipsilon Flow Management Protocol Specification for IPv4,” RFC 1953, May 1996.

[12] P. Doolan, B. Davie, D. Katz, Y. Rekhter, E. Rosen, “Tag Distribution Protocol,” Internet Draft draft-doolan-tdp-spec-00.txt, Sep 1996.

[13] P. Newman, W. L. Edwards, R. Hinden, E. Hoffman, F. Ching Liaw, T. Lyon, and G. Minshall, “Transmission of Flow Labelled IPv4 on ATM Data Links,” RFC 1954, May 1996.