

Internet-Draft	Arup Acharya, C&C Research Labs, NEC USA
Expiration Date:	Rajiv Dighe, C&C Research Labs, NEC USA
January 1998	Furquan Ansari, C&C Research Labs, NEC USA
	July 1997

IPSOFACTO: IP Switching Over Fast ATM Cell Transport

[<draft-acharya-ipsw-fast-cell-00.txt>](#)

Status of This Memo

This document is an Internet-Draft. Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

To learn the current status of any Internet-Draft, please check the "1id-abstracts.txt" listing contained in the Internet-Drafts Shadow Directories on ftp.is.co.za (Africa), ftp.nordu.net (Europe), munnari.oz.au (Pacific Rim), ds.internic.net (US East Coast), or ftp.isi.edu (US West Coast).

Abstract

This document describes a method for mapping IP flows to ATM switches. No signaling is necessary to setup a path through ATM switches. Switch controllers run a IP routing protocol and execute IP forwarding. The Ipsofacto component is responsible for mapping a IP flow to a switched path. The focus of this document is primarily on switching IP multicast. Mechanisms for switching unicast flows are also described.

Table of Contents

1. Introduction
2. Basic operation
3. Related Schemes
4. Switch configuration
5. Protocol Independent Multicast-Dense Mode
6. Protocol Independent Multicast-Sparse Mode
7. Switching Unicast flows
 - 7.1 TCP Optimizations
 - 7.2 Fixed Length IP routing tag
 - 7.3 Optimizing Mobile-IP flows
 - 7.4 Aggregation through IP-in-IP encapsulation
 - 7.5 Merging aggregated flows through a multipoint-to-point tunnel

7.6 Handling Route Changes

Acharya, Dighe, Ansari Expires: January 1998

[Page 1]

8. Conclusions
9. Security Considerations
10. Intellectual Property Considerations
11. Acknowledgments
12. References
13. Authors' Addresses

1. Introduction

IPSOFACTO is a methodology for executing the IP protocol stack directly on ATM switches. The ATM signaling stack is not used. This draft describes how IP multicast and unicast may be mapped directly to ATM switches.

The target scenario consists of (a) a cloud of ATM switches running the IP protocol stack and optionally, (b) hosts that are directly connected to this cloud. Each switch in this cloud runs the IP protocol stack along with an Ipsofacto component for mapping IP flows to switched paths. The description in this draft assumes configuration (a).

2. Basic Operation

The basic premise of operation for Ipsofacto is that all unused VCs on a input port of a switch are mapped to the switch control processor (Fig. 1).

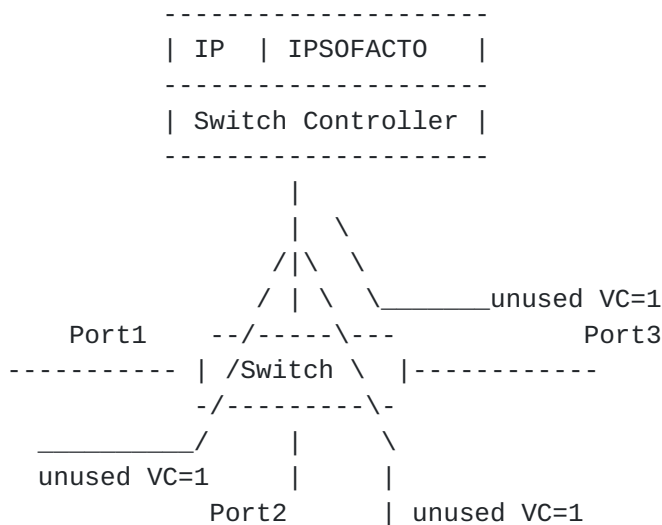


FIGURE 1

A cell-level switched path for data forwarding is established in the following way. A sender selects an unused VC on an outgoing link to forward the first packet of a new flow. This is received by the switch processor at the downstream end of the link, which then selects

outgoing links based on its IP routing tables, and an unused VC for each selected link. This first packet is then forwarded by the processor on the selected outgoing links by picking an unused VC on each link (Fig. 2).

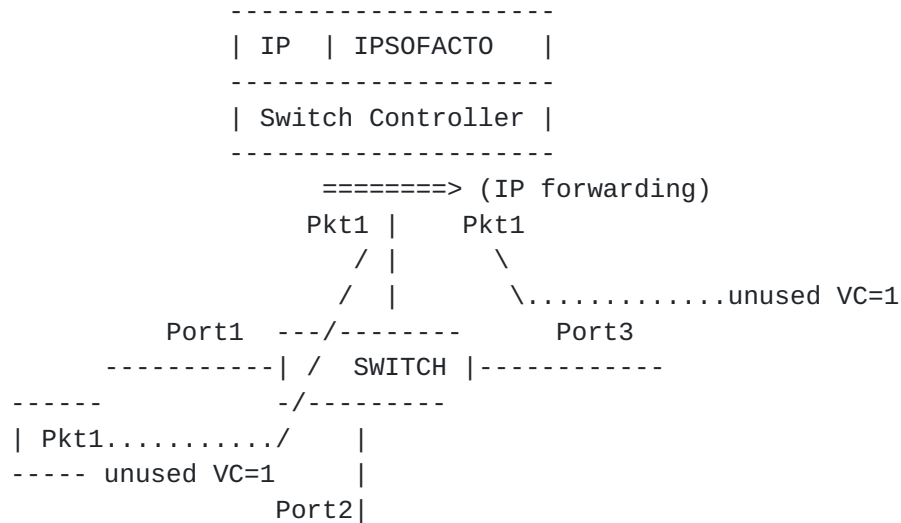


FIGURE 2

Subsequently, the switch processor adds an entry in the VC tables, <{input port, input VC}, {output port, output VC}>. Following packets are then switched at the cell level, eliminating the need for packet-level forwarding at the control processor.

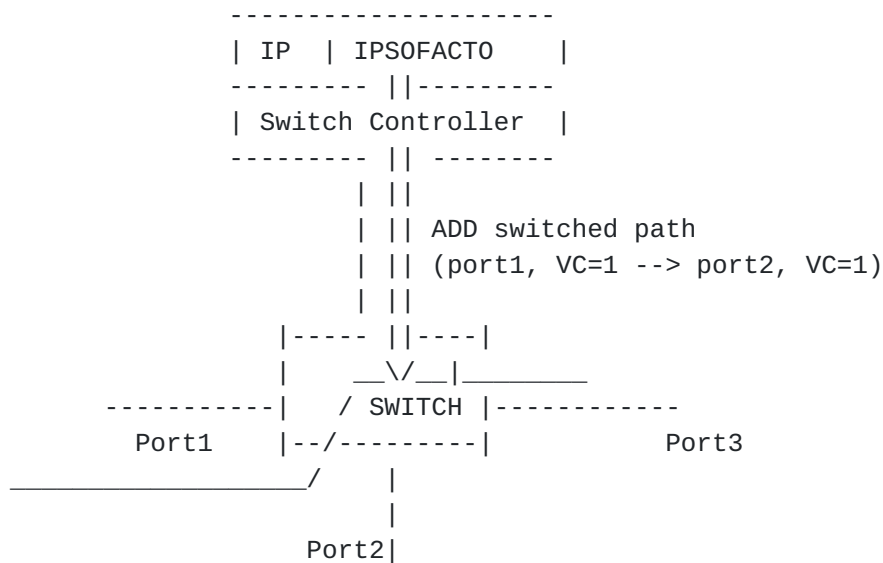


FIGURE 3

In contrast to data packets, which are forwarded through a switched path (following the first packet), a switched path is never created

for IP "control" messages: typically, IP "control" messages will be sent and received on a predefined "control VC", and will therefore, be forwarded through the switch processor. As a result of processing these control messages, a per-flow forwarding state may be established on the switch processor. Changes in the forwarding state, e.g. pruning an outgoing interface, is then used to modify the switched path (e.g. deleting an <outgoing port, VC> from the VC tables). When the forwarding state is removed from the control processor, the corresponding switched data path is released, by marking the input and output VCs as unused. For PIM, the "control" packets are distinguished from data packets, based on protocol number in the IP header. For TCP, the "control" packets may be decided based on flags in the TCP header (SYN/FIN). ICMP packets are always forwarded through the control processor.

When an outgoing VC, that is currently in use, is reclaimed back, i.e. marked as 'unused', this action is preceded by marking the VC to be in 'transient' state, sending a RECLAIM message from the upstream to the downstream node, and waiting for a RECLAIM-ACK in reply. When the downstream node receives a RECLAIM message, it marks the corresponding incoming VC as 'unused', responds with a RECLAIM-ACK, followed by sending RECLAIM messages further downstream. The RECLAIM message may need to be re-sent if a RECLAIM-ACK is not received within a specified interval. Once the RECLAIM-ACK is received, the VC is marked as 'unused' at the upstream node. Note that it may not be necessary to send a per-VC RECLAIM and RECLAIM-ACK, and instead, a single RECLAIM and RECLAIM-ACK message may contain multiple VC numbers; this cuts down the inter-switch control traffic at the expense of delayed reclamation.

A RECLAIM message is triggered as a result of some protocol action at the controller, e.g. an outgoing branch of a switched path may be removed because a PIM "Prune" message was received.

It is not necessary that the RECLAIM is only sent from the upstream to the downstream node; the downstream node may initiate the RECLAIM as well.

In this draft, the terms 'marking a VC unused' or 'reclaiming a VC' implicitly imply that RECLAIM and RECLAIM-ACK message exchange has been completed.

3. Related Schemes

Unlike MPOA [[MPOA](#)], Ipsofacto does not use ATM signaling to setup switched paths for IP flows. Ipsofacto is similar in spirit to Ipsilon's IP switching [[Ipsilon](#)] [IFMP] and Toshiba's CSR [[CSR](#)]. Ipsilon selects the VC for switching a flow from the downstream node, while CSR selects the VC at the upstream node. Both use a notification protocol (IFMP for Ipsilon, FANP [[FANP](#)] for CSR) to inform the upstream/

downstream neighbour of the selected VC. Ipsofacto informs the downstream node of the selected VC implicitly through usage. Other related proposals include Tag-switching [[TAG](#)] and ARIS [[ARIS](#)]: both pre-establish switched paths based on routing information.

4. Switch configuration

Each port of a switch is assigned to be an IP interface. Incoming cells on all unused VCs are directed to the control processor. When a packet is received on a unused VC, the incoming VC and port number information associated with the packet is made available to the Ipsofacto layer. Ipsofacto layer keeps track of all unused VCs on each port, and maintains a flow-table which maps a flow to a switched path. e.g. <source IP address, destination IP address, TTL>

---> <{input port, input VC}, list of {output port, out VC} pairs>

The flow-id is constructed from the source/destination IP addresses and port numbers along with the incoming TTL value on the packet.

Although not strictly necessary, we will assume that there is a dedicated control VC between neighbouring switches. Typically, the control VC will be used to forward IP multicast "control" messages (i.e. PIM packets with protocol number 103) between adjacent switches.

In effect, each switch together with its IP/Ipsofacto layers appears as a network router to its adjacent switches/hosts.

5. Protocol Operation: Mapping PIM Dense Mode [[PIM-DM](#)] flows in Ipsofacto

When there exists no forwarding cache entry at a switch controller for a multicast flow, the first data packet for that flow will install a forwarding cache for the flow at the controller, as specified by the protocol operation for Protocol-Independent-Multicast(Dense Mode). The VC# and the port number (selected by the upstream switch controller) on which the data packet is received, is available to Ipsofacto. Additionally, for each outgoing interface in the newly created multicast forward cache, Ipsofacto selects an unused VC to forward the packet to the downstream switch controller. At this time, it adds an entry in the flow-table, mapping the flow-id <source IP address, destination IP address, TTL> to a switched path {<input port, input VC>, list of <output port, output VC>}, and adds entries in the hardware VC routing tables corresponding to {<input port, input VC>, list of <output port, output VC>}. Subsequent packets in the flow are not processed by the controller, and are switched at the cell level making use of the hardware multicast capability of the ATM switching fabric.

In essence, the IP multicast routing protocol is used to establish the multicast forwarding state at the switch controller, while Ipsofacto maps this state to a point-to-multipoint VC within the switching fabric.

In PIM-DM, the forwarding state for a flow is associated with an expiration timer. This timer is restarted whenever a data packet is being forwarded. However, in Ipsofacto, packets subsequent to the first are not forwarded through the switch controller. Therefore, the rule for handling a timer expiration is as follows: when the timer

expires, the controller checks the corresponding switched path for any cell forwarding activity since the last expiration. If cells have been

forwarded in the interim period, the timer is restarted. Else, both the forwarding state and the switched path are deleted. The list of outgoing interfaces associated with a forwarding cache entry, may be modified by arrival of Prune and Graft messages. The corresponding switched path is modified as follows: for a Prune message, the <output port, output VC> for the interface on which the Prune is received, is deleted from the VC tables, i.e. the corresponding branch of the point-to-multipoint VC is deleted. If after deletion of the branch, no more outgoing branches remain (i.e. the forwarding state is now a negative cache entry), then the switched path is deleted, and the incoming VC is marked as unused, i.e. it points to the switch controller. When a Graft message for a group G is received, PIM-DM will add the received interface to the outgoing interface list for all (S,G) forwarding entries. Ipsofacto complements this action, by picking an unused VC and adding a branch <output port, output VC> to the corresponding switched path, for each of the (S,G) entries modified by the Graft message. If the (S,G) entry is a negative cache entry, then no switched path will exist for the (S,G) entry and Ipsofacto will not take any action; a new switched path will be created when with the arrival of the first packet.

Note that Ipsofacto does not change PIM protocol actions associated with PIM control messages; Ipsofacto only determines how the resulting multicast forwarding state needs to be mapped to a switched path.

Ipsofacto supports Distance-vector multicast routing protocol (DVMRP) [[DVMRP](#)] in a similar fashion.

6. Mapping PIM Sparse Mode [[PIM-SM](#)] operation under Ipsofacto

PIM-SM creates a shared forwarding tree rooted at a Rendezvous Point (RP) with receivers at the leaves. It allows for individual receivers to bypass the shared tree and instead receive data packets on a source-specific tree. Unlike PIM-DM, forwarding entries at intermediate routers are created by sending explicit Join messages towards the RP (shared tree) or the source (source specific tree), from last-hop routers.

Ipsofacto maps the shared forwarding entry on the RP-tree to per-source switched paths at intermediate switches. This is a slight departure from traditional packet-level routers which maintain a common (*,G) packet-level multicast forwarding state: under Ipsofacto, the forwarding state at the switch processor is shared by all (*,G) flows, but cell-level switched paths are kept separate for each (S,G) flow. The reason for separating the switched paths is that PIM-SM uses the shared tree to reach all receivers by default, while simultaneously using part of the shared tree to reach a subset of the receivers for specific senders. Therefore, an intermediate node on the tree may contain forwarding entries of the form (S,G) (with RPT-bit set) and (*,G) : though the incoming interface for both entries may be the same, their outgoing interfaces will differ. The decision whether

to use a (S,G) entry or (*,G) entry is taken on a per-packet basis, using the source address of the packet. But, such a decision cannot be made when forwarding data (cells) in a switched path, since the data packet is not forwarded through the switch processor. Hence, under Ipsofacto, instead of using a common point-to-multipoint switched path, a separate switched path is used for each multicast flow.

Under Ipsofacto, the first data packet for a multicast flow is handled similarly for both PIM-DM and PIM-SM. The outgoing interfaces, on which to forward the packet further, is selected from the multicast forwarding entry, and a switched path is added to the VC tables. A corresponding entry (<input port, input VC>, list of <output port, output VC>) is added to the flow-table. Unlike PIM-DM, where each multicast forwarding entry corresponds to a single entry in the flow table, each multicast forwarding entry (e.g. (*,G)) in PIM-SM may be associated with multiple flows (switched paths) and therefore, multiple entries in the flow-table. This one-to-many association between multicast forwarding entries and switched paths is explicitly recorded in the flow-table. When a Prune or a Join message affects a multicast forwarding entry, the associated entries in the flow-table will be modified accordingly, e.g. if an outgoing interface is pruned, then for each switched path associated with that forwarding entry, the corresponding <outgoing port, VC> will be deleted from the switched path. Analogously, for a Join, an unused VC will be selected on the outgoing port (interface) and added to the switched paths associated with the affected forwarding entry.

There are two special interactions that need to be considered for PIM-SM. The first occurs when an intermediate router is in the midst of switching from the shared tree to a source specific tree for a particular (S,G) flow. When the first data packet arrives on the source-specific tree, PIM-SM triggers a Prune message to be sent on the shared tree. Under Ipsofacto, this implies that the existing flow-specific switched path associated with the shared tree will no longer be used: the switched path (and the flow table entry) may directly be reclaimed when the Prune message is sent, or allowed to time out. Note that the arrival of the data packet on the source-specific tree will lead to a new switched path being setup.

The second interaction occurs at the Rendezvous Point (RP). The RP may receive data packets from a sender either encapsulated as unicast (Register) packets or as native multicast (which typically happens when the RP has joined the source-specific tree). In the former case, the RP cannot create switched paths for specific flows: it must decapsulate the packet first. Further, the flow table entry will now contain a list of <outgoing port, VC> pairs; there is no incoming port, VC. For packets belonging to the same flow, the RP must use the same VC on an outgoing port for successive packets. This VC is recorded in the flow table. In the latter case, i.e. when the RP receives a native multicast packet, it creates a switched path as mentioned earlier.

7. Switching unicast flows

Mapping the first data packet of a unicast flow to a switched path is similar to that for a multicast flow. IP routing determines the outgoing port, Ipsofacto selects an unused VC on that port and adds a switched path {<input port, input VC>, <output port, output VC>}. Unlike the case for multicast, where PIM control messages are implicitly used to refresh the switched path, switched paths for unicast flows require a separate refresh protocol.

The downstream endpoint of a switched path, periodically monitors traffic on the <input port, VC>, checks if the received packet header matches the expected header, and if so, sends a REFRESH message to the upstream node. Intermediate nodes need only forward the REFRESH message upstream; it is not necessary to locally monitor traffic activity. When the downstream endpoint of a switched path decides not to send a REFRESH message upstream, it may optionally send a RECLAIM instead to speed up the process of path teardown.

Other alternatives to refresh the switched paths are also possible, such as using IFMP or FANP for refreshing the switched unicast paths.

7.1 Optimizations for TCP flows

For TCP flows, data transfer is preceded by a three-way handshake. Under Ipsofacto, the switched path is setup when the first packet (SYN) is forwarded; thus, an end-to-end switched path is available prior to data transfer. The teardown of the switched path (i.e marking the switched path VC as unused at each hop) can be hastened by sending the FIN and the ack to the FIN packets through the switch controller instead of the switched path, i.e these packets are treated as "control" packets. The switched path can be reclaimed at a node after the FIN from each endpoint, and the corresponding acks have been forwarded through the controller. This optimization works only at nodes where the forwarding path for the TCP flow is symmetric, i.e. the incoming port for one direction of the flow is same as the outgoing port for the reverse direction, and vice-versa.

7.2 FLIRT (Fixed Length Ip Routing Tag)

The first packet of a flow requires an IP routing table lookup at each hop within the Ipsofacto cloud. This longest-prefix match can be replaced by an exact match on fixed number of bits by attaching a fixed-length tag with each packet. The tag-switching architecture [[TAG](#)] uses a tag distribution protocol [[TDP](#)] to distribute a consistent set of tags. However, when using ATM switches as Tag-switch-routers (TSR), the tag also defines the VC on which the packet is forwarded under the Tag Switching architecture.

Under Ipsofacto, the VC selected for forwarding a packet is decoupled from the tag: the VC selection is implicitly by usage, as explained

earlier. Regardless of the VC selected on the outgoing link, the upstream switch may optionally insert a FLIRT before the IP packet

within a AAL5 frame. (Absence of FLIRT is represented by zero-ing out the bits of the fixed length field). The FLIRT on an incoming packet is then used to decide the outgoing FLIRT and the outgoing port without doing a IP table lookup. Before forwarding the packet, the FLIRT is replaced with the new value.

If the FLIRT is absent (as may be the case when the first packet is lost), normal IP route lookup is performed.

In future, the FLIRT may be extended to hold additional fields, e.g. id of the ingress point to the Ipsofacto cloud, flow type (host pair, port pair, IP-in-IP), QoS parameters.

7.3 Optimizing Mobile-IP flows

In scenarios where a switch processor acts as the home agent for a mobile host[Mobile-IP], it may optionally speed up the process of tearing down a switched path (leading to the previous location of the mobile host). For flows initiated from a correspondent host, the packet is addressed to the mobile host's home address, which is then intercepted by the home agent when the mobile host is not at home. The home agent then forwards the packet after encapsulating with the mobile host's current foreign address. Under Ipsofacto, this leads to two switched paths, one from the correspondent host to the home agent and the second, from the home agent to the mobile's current location. The second switched path is setup by using the outer source/destination address of the encapsulated packet as the flow-id (i.e there are no transport port numbers), and in effect, aggregates all flows to the mobile into a single switched path.

When the mobile host changes its foreign address, it must send a Registration Notification to its home agent; the home agent replies with Registration Reply to the mobile host's foreign address. If the registration is accepted, then the switched path leading to the mobile host's previous foreign address is no longer useful. In this situation, Ipsofacto may optionally send RECLAIM messages to the downstream nodes of the existing switched paths leading to the mobile host's previous location. The new switched path is automatically setup when the first packet addressed to the mobile host's new foreign address is forwarded.

For each incoming flow, in general, it is not possible to bridge to the incoming switched paths and outgoing (aggregated) switched path, since the home agent must encapsulate each packet with the mobile's foreign address. Specific optimizations are presently under study.

7.4 Aggregating multiple unicast flows into a single switched path

As the previous section points out, it is possible to use IP-in-IP mechanism to aggregate multiple flows with a common pair of entry and

exit points into the Ipsofacto cloud. This requires support from the routing protocol operating in the Ipsofacto cloud, for the entry point

to learn the exit point for a given flow; all flows with a common exit point can then be aggregated using IP-in-IP encapsulation.

It is also possible to use VP switching within the Ipsofacto cloud, instead of using IP-in-IP encapsulation. In this case, the entry router would assign a different VC# for a new flow, but assign a VP# already in-use for flows that share a common exit point. The VPI field is used to switch flows with a common entry and exit routers, within the cloud, while the VCI field is used to locally demultiplex individual flows at the exit router. This would require the exit router to "learn" and cache the flow-to-VC mapping: when a new flow is received at the exit router for which it has no flow-to-VC binding, it installs such a binding from the IP header and the incoming VC. For each VC that is bound to a flow, it must then periodically reassemble cells from that VC to a packet and refresh/replace the corresponding flow-to-VC binding. Alternatively, a specific VC may be reserved within each VP for the entry point to periodically update/refresh the exit point of current flow-to-VC bindings.

7.5 Merging encapsulated flows: multipoint-to-point aggregation

The previous sub-section briefly outlined how multiple flows with common entry and exit points may be aggregated to a single flow using IP-in-IP encapsulation. Multiple such flows, each from a different entry point, but with a common exit point out of the Ipsofacto cloud, may also be merged in the following way: at a switch N, which has an existing switched path for an IP-in-IP flow F to exit point B, may select the same outgoing VC as F for a new flow F1 to the same exit point B. First, this requires VC-merge capability at intermediate switches to avoid cell-interleaving. Second, when the REFRESH message from B is forwarded on the reverse path (as that of the switched path), a hop-count needs to be included in the message, which is incremented after each hop. This enables B to decide if the remaining TTL value on the new incoming flow is no less than the hop-count on the REFRESH message. If so, the incoming flow may be merged into the existing flow. Third, at each merge point, such as N, traffic activity on each incoming branch needs to be monitored periodically. When a REFRESH message is received from the downstream node, N forwards it along each incoming branch after incrementing the hop count if there was traffic activity on that branch within the last k periods.

7.6 Routing table changes/ Loop detection and/or avoidance

Solutions for efficiently handling changes in IP routing tables require further study. One approach is to leave an existing switched path untouched in spite of a change in the corresponding IP route, if intermediate switch controllers on the path continue to receive REFRESH messages from downstream. Alternatively, when a routing table entry is updated, all affected switched paths may be forced to go through IP forwarding by marking the incoming and outgoing VC(s) of

the path as unused.

8. Conclusion

This draft presented a scheme for mapping IP flows to ATM switches, with special emphasis on IP multicast. Mechanisms for mapping unicast flows, refreshing unicast switched paths, handling flows leading to mobile hosts and fixed length tags to speed up routing were outlined.

9. Security Considerations

Security considerations are not addressed in this document.

10. Intellectual Property Considerations

NEC may seek patent or other intellectual property protection for some aspects discussed in this document.

11. Acknowledgments

The authors would like to acknowledge the following people for discussions on the contents and/or inputs to this draft: Dipankar Raychaudhuri, Kojiro Watanabe, Bala Rajagopalan, Kazuhiko Isoyama, Toshiya Aramaki, Ajay Bakre, Atsushi Iwata and Hiroshi Suzuki. The authors thank Hideyuki Sakaguchi, Joe Darabpour, Bun Mizuhara, and Youichi Ohteru for supplying us with switch software for prototyping Ipsufacto.

12. References

[DVMRP] T. Pusateri, "Distance Vector Multicast Routing Protocol", Internet Draft <[draft-ietf-idmr-dvmrp-v3-04](#)>, Juniper Networks, February 1997

[TAG] Y. Rekhter, B. Davie, D. Katz, E. Rosen, G. Swallow, "Cisco Systems' Tag Switching Architecture Overview", [RFC 2105](#), Cisco Systems, Inc., February 1997

[TDP] P. Doolan, B. Davie, D. Katz, Y. Rekhter, E. Rosen, "Tag Distribution Protocol", <[draft-doolan-tdp-spec-01.txt](#)>, Cisco Systems, Inc., May 1997

[FANP] K. Nagami, Y. Katsube, Y. Shobatake, A. Mogi, S. Matsuzawa, T. Jinmei, H. Esaki, "Toshiba's Flow Attribute Notification Protocol (FANP) Specification", [RFC 2129](#), Toshiba R&D Center, April 1997

[CSR] Y. Katsube, K. Nagami, H. Esaki, "Toshiba's Router Architecture Extensions for ATM : Overview" , [RFC 2098](#), Toshiba R&D Center, February 1997

[ARIS] Arun Viswanathan, Nancy Feldman, Rick Boivie, Rich Woundy, "ARIS: Aggregate Route-Based IP Switching", Internet-Draft,

<[draft-viswanathan-ariss-overview-00.txt](#)>, IBM Corp., March 1997

Acharya, Dighe, Ansari Expires: January 1998

[Page 11]

[IFMP] P. Newman, B. Edwards, R. Hinden, E. Hoffman, F. Ching Liaw, T. Lyon, G. Minshall, "Ipsilon Flow Management Protocol Specification of IPv4 Version 1.0", [RFC 1953](#), Ipsilon, Sprint, May 1996

[Ipsilon] P. Newman, G. Minshall and T. Lyon, "IP switching: ATM Under IP", Submitted to IEEE/ACM Transactions on Networking

[PIM-SM] D. Estrin et al. "Protocol Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification", [RFC 2117](#)

[PIM-DM] Steven Deering et. al. "Protocol Independent Multicast version2, Dense Mode Specification", <[draft-ietf-idmr-pim-dm-05.txt](#)>, IETF Draft, May 1997

[MPOA] ATM Forum, "Multiprotocol Over ATM version 1.0 (Baseline Text Version 16)", Andre N. Fredette, Editor.

[IPIP] C. Perkins, "IP Encapsulation within IP", [RFC 2003](#), October 1996

[Mobile-IP] C. Perkins, Editor, "IP Mobility Support", [RFC 2002](#), October 1996

[13. Authors' Addresses](#)

Arup Acharya
C&C Research Labs, NEC USA
4 Independence Way, Princeton
NJ 08540, USA
Phone: +1 609 951 2992
Email: arup@ccrl.nj.nec.com

Rajiv Dighe
C&C Research Labs, NEC USA
4 Independence Way, Princeton
NJ 08540, USA
Phone: +1 609 951 2982
Email: rsd@ccrl.nj.nec.com

Furquan Ansari
C&C Research Labs, NEC USA
4 Independence Way, Princeton
NJ 08540, USA
Phone: +1 609 951 2965
Email: furquan@ccrl.nj.nec.com