Dual-mode Routing:
A Generic Framework for IP over ATM Integrated Routing

Bernard Sales and Piet Van Mieghem

Alcatel Alsthom
Corporate Research Center
AReNA
F. Wellesplein 1, B-2108 Antwerp, Belgium
salesb@btmaa.bel.alcatel.be *and* mieghemp@rc.bel.alcatel.be

Abstract

Design issues for an integrated routing architecture for IP and ATM are outlined. Two separate aspects of this integration are (1) a common routing architecture for IP and ATM (layer integration) and (2) integrating best-effort (BE) and QoS routing architecture (service integration). Whereas the first level of integration is highly recommended, we show that the second level of integration is not desirable because BE and QoS traffic have, in terms of routing, contradictory requirements. Four critiria are proposed, namely, route refreshing vs. route pinning, hop by hop vs. explicit routing, pre-computed routes vs. on-demand route computation and stable vs resource related metrics. A fifth alternative is whether or not to integrate in the routing architecture the capability to compute shortcut paths, that are bypassing layer 3 (L3) nodes and using only layer 2 (L2) devices. Using this framework, we conclude that BE traffic flows are well served by a combination of route refreshing, hop by hop routing pre-computed routes and static routing metrics while QoS routing is built on route pinning, explicit routing, on-demand route computation and resource related metrics. Finally, the ability to compute L2 shortcuts in an L2/L3 integrated routing architecture is an added value simplifying the overall network design and optimising the efficacy of the forwarding path.

1. INTRODUCTION

For multimedia applications ATM has the advantage over IP in that it has been designed with QoS in mind. However, the ATM model is not well-suited for a number of applications for which a datagram BE model is sufficient. Under pressure of the research community, Internet is moving to a multi-service network and thus can be perceived as a competing solution to ATM. Rather than to oppose them, we advocate that it would be better to design solutions enabling a convergence and take advantage of the respective complementary: the Internet design crucially relies on global connectivity ("IP over everything") and is providing BE services, while the ATM model is based on sophisticated resource management model and is offering the guarantees required by multimedia applications. The key point is to integrate IP and ATM to build a high-bandwidth scaleable Internet, where BE and QoS based services coexist.

The Internet community has pioneered the implementation and large scale deployment of routing protocols that are, unfortunately, not tailored for the support of QoS sensitive services. QoS capable routing protocols are currently investigated in standardisation bodies like ATM Forum and IETF. The most elaborated routing protocol supporting QoS is the PNNI specification of the ATM Forum [1].

As far as IP over ATM is concerned, a variety of ideas have been proposed at the ATMF (MPOA [2], PAR [3], I-PNNI [1]) and at the IETF (Classical IP over ATM [4] [5], NHRP [6], OSPF Address Resolution Advertisements *(ARA)* [7], MARS [8], Tag switching [9], IP switching [10], ARIS [11], IPSOFACTO [12], CSR [13],...). Most of them are today considered as competitors. The differences among these solutions can be characterised according to : 1) how the ATM connectivity among IP routers is managed (ATM VC management), 2) how the support of data flow is optimised in terms of L2/L3 (shortcut routing) and 3) how the routing procedures defined at IP and ATM level are collaborating (layered routing vs. integrated routing).

Considering the myriad of proposals, we are faced with the fact that routing procedures for IP and ATM providing BE and QoS services are badly integrated. In this paper, we investigate how to design an integrated routing architecture filling the requirements imposed by a very large scale Internet based on IP and ATM. To conduct this analysis, a classical approach is to start from the existing proposals for IP over ATM, and confront them to select the architecture matching best the targeted objectives. We feel that, because of the complexity of integrating IP and ATM, a better approach is to go one step back in the design process and identify the basic design alternatives which must be considered when engineering a routing protocol.

2. DEFINITIONS

2.1 Integrated IP and ATM

When integrating two environments where the protocols of one environment (e.g. IP at layer 3) uses the services provided by the other protocol suite (e.g. ATM at layer 2), three aspects should be carefully envisaged:
(1) integration at the addressing and at the routing level;
(2) mapping the BE and the QoS data flows (defined below) onto appropriate forwarding paradigms provided in both environments (i.e. CO or CL);
(3) mapping signalling, data flow management and traffic parameters between both environments.

In this paper, we will concentrate our study on the integration of the routing aspects.

2.2 Flows

We define a *flow* as a set of correlated information elements (packets, cells, ...) transferred from a source to one or more destinations. A flow can be defined recursively, in the sense that a flow can be an aggregation of a number of little granularity flows. A flow is characterised by a set of service requirements to be met by the network while transporting the flow. A flow is supported over a path. A flow for which the service requirements include firm guarantees is defined as *a QoS flow* otherwise we term it a *BE flow*.

3. DESIGN CHOICES OR CRITERIA

3.1 Route refreshing vs. route pinning

Route refreshing indicates that the route is dynamically updated, periodically or/and after any environment change that could affect the route (failures, topology update, congestion,...). This technique tends to maintain optimal routes (according to the applied metrics) in all circumstances and is very powerful since it provides for automatic reconfiguration of routing paths triggered by environment changes. However, route refreshing is vulnerable to stability problems when routing metrics are coupled to traffic behaviour (e.g. bandwidth) that rapidly requires updating.

Route pinning ‘freezes’ the route for a specified duration of time, unless some exceptional condition (node failure, link failure, network management action, ...) occurs. Route pinning is, by definition, stable due to well controlled routes. But, it might generate sub-optimal routing decisions resulting in an enhanced blocking probability. Specifically, a ‘pinned’ route which was optimal at set-up can become sub-optimal during to the evolution of network resource utilisation. To remedy sub-optimality, a priority mechanism may be implemented in conjunction with route pinning. Priority management can guarantee that setting up a flow with a higher priority also has a higher probability to succeed, compared to a flow with a lower priority. In case a high-priority flow set-up is blocked because the necessary resources along the path are allocated to lower-priority flows, then these lower priority flows may be released and re-routed on alternate paths. The choice between route refreshing and route pinning is also influenced by a possible generation of *routing loops* [14] [15]:

- *permanent loops* caused by malfunctioning of routing algorithms
- *transient loops* due to non-synchronized topology databases

An architecture with route refreshing must only deal with the first type of loop: potential loops formed as a result of routing base inconsistencies will only persist during the convergence phase of the algorithm. Those loops will disappear when the routing bases get synchronised. In contrast, when route pinning is applied, the routing algorithms must compute loop-free paths [16] otherwise, there is a risk that routes with loops will be maintained until the routes are unpinned.

3.2 Hop by hop vs. explicit routing

In *hop by hop routing*, the nodes along the path determine the route and successively specify the next node on the route. Each intermediate node must apply the same routing algorithm and in a cooperative and distributed way, the route is built and maintained.

With *explicit routing*, a particular node (typically, the source node) selects the appropriate route and specifies the address of every node on the path. Hence, the routing decision is centralised. Explicit routing furnishes a powerful mechanism to enforce policy decision. In addition, explicit routing is robust against routing loops, as the complete route is determined by a single network node based on one topology database. In contrast, hop by hop routing is prone to both permanent and transient loops.

3.3 Pre-computed routes vs. on-demand route computation

Pre-computed routes are computed and maintained in advance by the network, independently of actual flow routing requests. *On-demand route computation* implies that the routing process is activated only when a data flow is established.

Pre-computing routes offers a better performance in terms of flow set up delay since the node just fetches the correct entry from the routing table, without the burden of a costly route computation. Nevertheless, this method has drawbacks since the performance of pre-computed routes is bounded by the reliability and accuracy of information stored in the routing base at the time the route is effectively used. Pre-computations need to compute all possible routing requests. The efficiency of pre-computing routes is the trade-off between the number of combinations that need to be computed and the probability of consulting a specific entry. Guillen *et al.* [15] have proposed only to pre-compute and maintain frequently used routes (equivalent to the *cashing* principle).

3.4 Shortcut routing

In a network consisting of a mix of layer 2 (e.g. ATM) and layer 3 (e.g. IP) forwarding devices, minimising the use of layer 3 forwarding improves the throughput since layer 3 devices (e.g IP routers) perform - in constrast to layer 2 (ATM) switches - complex processing of packets including a table look up based on the longest prefix matching the destination address. Avoiding layer 3 in favour of layer 2 forwarding is coined *"shortcut routing"*.

To integrate the shortcut routing in the architecture, the topology information used by the routing protocol should include the capability to distinguish between routers, switches and integrated switch-routers (ISR). The route computation can take this layer 2-layer 3 additional information into account to find a 'faster' path.

3.5 Static metrics vs resource related metrics

Static metrics are updated as result of a network management action. Static metrics, by definition, do not model the actual utilisation of resources in the network and are sometimes referred to as *administrative weight*.

Resource related metrics, on the other hand, are regularly updated according to the actual resource utilisation in the network. This updating process is generally difficult if, due to heavily fluctuating traffic, the resource utilisation more rapidly changes than the time available to distribute (flood) this infomation over all topology databases of the network.

4. INTEGRATED ROUTING

We will focus on the following aspects:

- *layer integration*: a single routing architecture encompassing both IP and ATM versus two specific routing architectures, one for IP and another one for ATM (layered routing);
- *type of service integration*: a single general-purpose routing paradigm independent of type-of-service versus a dedicated routing paradigm for BE and another one for QoS flows.

This first level of integration is studied next while the second integration aspect is examined in section 4.2.

4.1 The layer integration perspective

Layered routing means that both the IP layer and the ATM layer use their own routing functions. As these routing functions belong to separate layers, they are, according to the layer definition, isolated from each other. More precisely, functions belonging to separate layers can only communicate via layer service interactions which usually hide, from the higher layer perspective, the internal structure of the lower layer. In addition, the lower layer is not aware of the existence of the higher layer. In practice, this implies that IP and ATM routing functions operate isolated from each other.

On the other hand, *integrated routing* means that the IP and ATM forwarding functions rely on a common routing platform and thus share the same routing architecture.

An integrated routing approach has a number of advantages:

- A single routing architecture for both IP and ATM dramatically simplifies the address resolution scheme: an address resolution protocol like ATMARP is no longer needed because both IP and ATM addresses are specified in the topological information base shared by IP and ATM. In fact, the support of two separate addressing mechanisms, one for IP and the other one for ATM, is no longer needed.
- An integrated routing architecture implies that shortcut routes are embedded in the routing base, thus, with integrated routing, shortcuts appear in more natural manner than in the layered counter part.
- Since the combined resources of IP and ATM are taken into account, integrated routing leads to better routing decisions with respect to resource allocation.

A single advantage of layered routing compared with integrated routing is that IP routing is independent of any other layer 2 protocol. However, this benefit is considered weak as compared to the promissing integrated routing approach. Hence, layered routing is merely regarded as a short term solution for operating IP over ATM, while integrated routing is the ultimate routing architecture for internetworking between IP and ATM.

4.2 The type of service integration perspective

4.2.1 Optimal strategy for BE and QoS

It seems reasonable, even for QoS flows to argue, based on queueing theoretical considerations [17] that the overall maximum throughput in a packet network can be obtained when each data unit of a flow is routed independently from the others (i.e. CL) over all paths fulfilling the QoS requirements for that flow. In terms of routing alternatives identified in section 3, this corresponds to combine on-demand routing, route refreshing, hop by hop computation and resource related metrics and is coined the *"optimal routing strategy"*. As the same strategy applies for BE as well as for QoS, the routing can be viewed as integrated. Unfortunately, we failed to find a rigorous proof of this claimed optimality in the literature so far. An intuitive argument is that any other combination of routing alternatives poses constraints on the packets of a flow - all packets of a same flow have to follow the same path through the network (i.e. CO) - and, hence, it has the probability to miss opportunities to serve more packets than with the optimal routing strategy.

Only recently, Harrison [18] in his BIGSTEP approach has proposed a concrete strategy for near optimal flow management in stochastic networks using what we have defined as the optimal routing strategy, dynamic input control and resource reservation. The policy is shown to be asymptotically optimal in the heavy traffic limit, where the Brownian motion approximation is well justified (see numerous references in his paper). In simple terms, Harrison's BIGSTEP policy consists of controlling the service discipline in the nodes of the network according to a time-discrete-review control policy that is strongly based on Brownian motion theory and capable to treat multi-service classes. Of course, Harrison's strategy assumes a centralised approach which is perhaps the most important, conceptual problem. Current data networks are based on a distributed approach which imply a delay on the information exchanged between the nodes to obtain that required centralised view. It is feared that the information exchange will take longer than the update time for a Harrison-like based review policy. In other words, the needed information about the network's state will be not available to compute the optimal routing strategy for each packet subject to QoS requirements. The other shortcomings in Harrison's method seem to have a minor consequence. In short, it is doubtful, in a distributed way to achieve the optimal routing strategy.

4.2.2 BE flows

Most routing architectures designed for the current, BE Internet are constructed on a combination of hop by hop routing, pre-computed routes and route refreshing. Let's briefly review the arguments

Explicit routing vs hop by hop

Explicit routing would require to store the complete path in a datagram or to set-up the connection (via signaling). The first alternative clearly creates impractically large overhead, while the second imposes that the routers maintain state info, which needs additional control info (and time) to install and release the state info. As a result, datagram routing uses hop by hop: this solution minimises the state information to be maintained in the network and is acceptable in terms of packet overhead.

On-demand vs pre-computed

Hop-by-hop routing requires a routing at every intermediate node of the path. To minimise the computational routing effort for a datagram, the best method corresponds to the setting up and maintaining of a semi-permanent full connectivity (i.e. pre-computed route) which should remain, in order to scale, as stable as possible. As pre-computing all routes to any destination necessitates a feasible complexity, only one single link metric seems desirable because a single metric results in a polynomial routing complexity whereas routing in a topology characterised by more than one link metric leads to a pseudo-polynomial complexity [19].

Route refreshing

Route refreshing is generally adopted because it enables an incremental change in the installed routes and it is more robust with respect to the formation of routing loops.

4.2.3 QoS flows

QoS routing uses resource related metrics

The routing decision is based on information reflecting the actual network utilisation. This routing information is disseminated on a periodic and/or event-driven basis. For scaleability reasons however, the routing information updates are distributed only when a significant change occurs (e.g. [1]).

Pre-computing all the routes is not feasible

Pre-computing routes for QoS routing means to continuously maintain routes for any pair of source/destination and this for each combination of QoS parameter values. If we have k QoS parameters, the total number of QoS value combination (i.e. the number of Classes of Service) is given by v^k (v is the maximum number of values that a discrete QoS parameter can adopt). Thus, for each source- destination pair, we have to compute at most v^k routes which leads us to conclude that pre-computing the routes for any combination of QoS parameter values is not feasible for large v or k.

Route refreshing is not applicable for QoS routing

Although route refreshment is desirable, it is unfortunately to difficult to exploit in practice because changing one QoS path implies the re-computation of all installed QoS paths which is not feasible due to the NP complete character of the QoS routing In addition, the routing information is more dynamic due to link metric coupling with resources so that the optimal route for a given flow would therefore frequently change. Applying route refreshing will lead to unstable flows oscillating among these best routes. To prevent oscillations, path freezing or route pinning (for some time) is mandatory.

Hop by hop routing is not convenient for QoS routing

Route pinning means that the route is maintained for a certain period of the time even though, during this period, the metrics values maintained by the routing protocol may vary. As a result, when route pinning is used, the routing protocol must provide "loop free" routes in any circumstances [16]. Otherwise, paths with potential loops will be maintained until the route is released. The fact that QoS routing is based on the route pinning mechanism implies that hop by hop routing will not be a sound solution for QoS routing because of its fragility with respect to the loop formation.

5. THE SOLUTION: DUAL-MODE ROUTING

The above discussion indicates that routing for datagrams (e.g. the best effort Internet traffic) and for QoS cannot be fully integrated because the route computation processes and the route maintenance are founded on different paradigms: best effort routing relies on a combination of pre-computed, hop-by-hop, route refreshing and stable routing metrics while QoS is based on on-demand, explicit routing, route pinning and resource related metrics.

However, we have argued that the layer integration of IP and ATM is attractive: IP and ATM should share the same routing architecture and different routing paradigms must be combined, one to support QoS flows and another to support BE flows.

We call this concept *dual-mode routing*, where the same routing stratum is used by IP and ATM but where two routing modes are supported, one for BE and the other for QoS. We believe that the established framework for routing is generic and can be used to evaluate existing and future routing protocols.

6. REFERENCES

[1] PNNI Specification, ATM Forum, 1996.

[2] Muti Protocol Over ATM, Version 1, AF-MPOA-0087.000, ATM Forum, July 97

[3] PNNI Augmented Routing, btd-pnni-par-01.01, ATM Forum, September 1997.

[4] Mark Laubach, “Classical IP and ARP over ATM,” RFC 1577

[5] Mark Laubach, Joel Halpern, “Classical IP and ARP over ATM,” <draft-ion-ipatm-classic2-03.txt>, October 1997

[6] James V. Luciani, Dave Katz, David Piscitello, Bruce Cole, “NBMA Next Hop Resolution Protocol (NHRP)” <draft-ietf-rolc-nhrp-12.txt>, September 1997

[7] Rob Coltun, Juha Heinanen, “The OSPF Address Resolution Advertisement Option”, draft-ietf-ospf-ara-00.txt, August 1997

[8] G. Armitage, "Support for Multicast over UNI 3.0/3.1 based ATM Networks.", RFC 2022

[9] Yakov Rekhter, Bruce Davie, Dave Katz, Eric Rosen, George Swallow, Dino Farinacci, “Tag Switching Architecture - Overview,” draft-rekhter-tagswitch-arch-01.txt, July 1997

[10] Peter Newman, Tom Lyon, Greg Minshall, “Flow Labelled IP: A Connectionless Approach to ATM,” proceedings of IEEE Infocom, March 1996

[11] Arun Viswanathan, Nancy Feldman, Rick Boivie, Rich Woundy, “ARIS: Aggregate Route-Based IP Switching,” draft-viswanathan-aris-overview-00.txt, March 1997.

[12] A. Acharya, R. Dighe, F. Ansari, "IPSOFACTO: IP Switching Over Fast ATM Cell Transport"

[13] Y. Katsube, K. Nagami, H. Esaki, “Cell Switch Router - Basic Concept and Migration Scenario,” Networld+Interop'96 Engineer Conference, July 1996

[14] J. Garcia-Luna, “A Unified Approach to Loop-Free Routing Using Distance Vectors or Link States,” proceedings of SIGCOMM 89, pp. 212-223, Austin, 1989

[15] A. Guillen, R. Najmabadi Kia, B. Sales, “An Architecture for Virtual Circuit/ QoS Routing,” Proc. of First IEEE International Conference on Network Protocol, pp.80-87, Los Alamitos, USA, 1993

[16] R. Najmabadi Kia, B. Sales, “Routing Architecture for the support of the Connection-mode Network Service,” Proceedings of the 3rd Joint European Networking Conference, Computer Networks and ISDN Systems 25(4-5), pp 405-410, North Holland, 1992

[17] Walrand, J., *An Introduction to Queueing Networks*, Prentice-Hall International, Inc., 1988

[18] Harrison, J. M., “The BIGSTEP Approach to Flow Management in Stochastic Processing Networks”, in *Stochastic Networks, Theory and Applications*, edited by F. P. Kelly, S. Zachary and I. Ziedens, Clarendon Press, Oxford., 1996

[19] H. De Neve and P. Van Mieghem, “TAMCRA: A Tunable Accuracy Multiple Constraints Routing Algorithm,” submitted to Sigcomm’98.